

# 行政院國家科學委員會專題研究計畫 成果報告

## 內容感知與內容呈現模型之研究(3/3)

計畫類別：整合型計畫

計畫編號：NSC94-2524-S-029-001-

執行期間：94年05月01日至95年04月30日

執行單位：東海大學資訊工程與科學系

計畫主持人：朱正忠

計畫參與人員：林幸宜 許敏彥

報告類型：完整報告

處理方式：本計畫可公開查詢

中 華 民 國 95 年 7 月 31 日

行政院國家科學委員會補助專題研究計畫  成果報告  
 期中進度報告

智慧型數位學習平台前瞻技術之設計與實作－

內容感知與內容呈現模型之研究(3/3)

計畫類別： 個別型計畫  整合型計畫

計畫編號：NSC 94－2524－S－029－001－

執行期間：2005 年 5 月 1 日至 2006 年 4 月 30 日

計畫主持人：朱正忠

共同主持人：

計畫參與人員：林幸宜，許敏彥

成果報告類型(依經費核定清單規定繳交)： 精簡報告  完整報告

本成果報告包括以下應繳交之附件：

- 赴國外出差或研習心得報告一份
- 赴大陸地區出差或研習心得報告一份
- 出席國際學術會議心得報告及發表之論文各一份
- 國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、  
列管計畫及下列情形者外，得立即公開查詢

涉及專利或其他智慧財產權， 一年  二年後可公開查詢

執行單位：東海大學資訊工程與科學系

## 中、英文摘要：

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），以協助老師製作上課用的教學內容，其中包括靜態方面（Powerpoint、Word、PDF）與動態方面（Video、Audio）的教材，並藉由這些教材的整合以提供學習者多樣的呈現方式。然而，大部分 LCMS 之開發卻還是以桌上型的電腦環境為主，而非針對行動裝置。再者，每位學習者亦有屬於自己的學習特徵，因此同一份教材並不適合所有的學習者使用。有鑑於此，本計畫提出了一個內文感知中介軟體（Context-Sensitive Middleware，CSM），以轉換同一份教材內容於不同的行動裝置，將呈現方式作最佳化的處理，並且針對每位學習者的特徵，調適出適合的教材，讓學習更加有效率。

Nowadays, lots of Learning Content Management Systems (LCMSs) develop convenient authoring tools to help instructors fulfill their learning contents which may include static (powerpoint, word, pdf) and dynamic (video, audio) files, and then integrate these learning contents to provide learners with diversiform displaying. However, most of LCMS pay attention to develop systems based on desktop computer environments, rather than mobile devices. Moreover, each learner has his/her own learning characteristic, and not all learners can study the same learning source well. In view of it, this project develop a Context-Sensitive Middleware (CSM) to transform the same learning contents to different mobile devices, optimize their displaying format, adapt the learning source by each learner's learning characteristic and make the learning more efficient.

**關鍵詞：**內文感知、內容呈現、行動學習、動態最適化、學習內容管理系統、學習風格、學習活動、時間間隔序列樣版。

# 目錄

前言.....	P.2
研究目的.....	P.2
文獻探討.....	P.3
研究方法.....	P.4
結果與討論.....	P.19
參考文獻.....	P.21

## 1. 前言：

學習的本質，無非是有效的吸收知識。傳統上，學校是求知的必要場所，故學習必然受到時空的限制，導致學習品質的下降。而所謂的數位學習 (e-Learning)，意指學習者可藉由電腦與網際網路的輔助，故任何時間任何地點都可進行學習。

當我們將桌上型網路基礎之數位學習，轉換成鬆散式連結各種設備之行動數位學習的同時，必須遵循幾項新原則：數位學習要能符合使用者所需求的內容，考慮使用者何時才需要，以及使用者如何得到他所需要的資料。現實生活中，我們可以用多種方式來與電腦溝通。因此，數位學習的應用，應該被設計與實作成具有鍵盤、麥克風、光筆、電話按鍵、大型螢幕、小型螢幕、彩色螢幕、黑白螢幕等工具使用上的優點。然而，大部分的學習內容管理系統 (Learning content Management System, LCMS)[1]並無法處理與內文(context)相關的議題，誠如他們並非「內容感知」(context-sensitive)的學習內容管理系。

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。Context sensitivity 就是一種應用系統的一種功能，能夠偵測和分析來自不同的 context，它讓應用軟體能夠在不同的環境作不同的動作。針對不同的 device 格式，需有不同的互動模式。一個使用 WAP 手機的使用者想要和 e-Learning 的應用軟體作迅速互動是很困難的[2][3]，因此，device-sensitivity 可能就要知道使用者的需求，以便回應正當的資訊給使用者。另外一個例子就是 context-sensitive 要知道使用者在哪裡。由於無線設備發展迅速，學習素材必須可讓無線使用者有如使用一般的電腦。當然，資料內容的需求對於一般的電腦和無線環境下的電腦必須是不同的。

## 2. 研究目的：

在傳統，學習模式是以師資為學習活動的中心、以教室為學習活動的場所，然而現今因為資訊的快速增加、知識的快速汰舊，讓傳統學習方式面臨困境。為了提高學習的效率和效益，「數位學習」導入資訊技術和網路技術：利用多媒體互動學習特性、個人化學習歷程，網際網路的互通性，進而達到以學習者為學習活動的中心、在需要的時候學習、以及處處可以學習之境地。

### 2.1. 第一年研究目的：

為了讓學習更加便利，我們設計並且實作了一個 context-sensitive 的內容呈現模型。此模型不但能輕易的將 context-sensitive 加入學習內容管理系統 (LCMS) 中，更能夠完整的與像 SCORM[4]這樣之內容呈現國際標準結合。因此，任何以 SCORM 為基準所制訂的文件，皆能不需經過修改，直接運用於我們所提供的具 context-sensitive 之 LCMS 中。

### 2.2. 第二年研究目的：

由於現今的行動裝置 (PDA、Smartphone、Notebook 等)發達，於是人們開始透過行動裝置來學習，但由於行動裝置的多樣性 (Diversity)，因此，為了提升學習教材的可再用性 (Reusability)，本計畫第二年提出了一個內文感知中介軟體 (CSM) [5]來達到此目的，其功能在於轉換同一份教材內容於不同的行動裝置，然後透過動態最適化 (Dynamic Adaptation) 將呈現方式作最佳化的處理。

### 2.3. 第三年研究目的：

傳統上，學生都是透過老師所給與的教材來學習。在教導的同時，老師可以透過觀察以及與學生的互動，進而給予個別學生所適合的指導方式。然而，透過 E-Learning 的學習

方式，老師想要掌握每位學生的學習特性，卻是一件困難的事。這是由於透過網路的學習，並非一種面對面的教導、學習。有鑑於此，本計畫目的在於擷取學生的學習特徵，進而提供完整的學習教材。

### 3. 文獻探討：

#### 3.1 SCORM

SCORM (The Sharable Content Object Reference Model) 是由ADL (Advanced Distributed Learning) 所提出的一套標準，其發展之目的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學習元件 (Learning Object) [6][7][8][9]。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。在學習元件方面描述了CAM(Content Aggregation Model)及RTE(Run-Time Environment)，在SCORM 2004 又加入了SN (Sequencing and Navigation)。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[10]。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[11]，但目前應用上還不是很普遍。

#### 3.2 XML 與 XSL

XML 是由 W3C(World Wide Web Consortium)所制定而成的標準語言[12]。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) [12]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 (Linking Specification)。
- XSL Formatting Objects：這是一種 XML 的字彙，可被用來說明結構化的語意。

### 3.3 MVC 模版 (MVC Pattern)

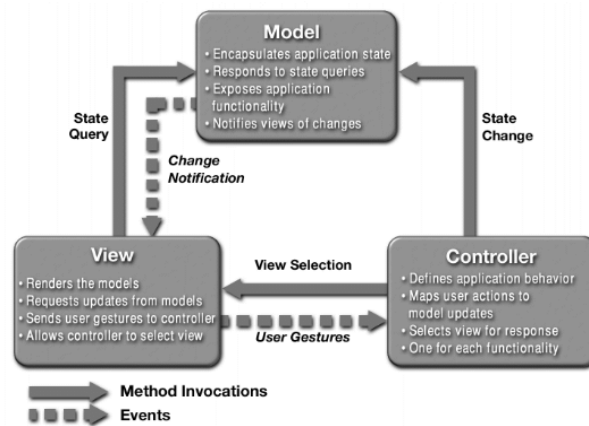


圖 1：MVC 模版架構圖

MVC (Model View Controller) [13] 是一種常被使用到的模版 (Pattern)，如圖 1 所示。MVC 主要是作為使用者介面設計 (UI Design) 的框架 (Framework)。依照 MVC 架構，可把功能相關的物件分類出來並集中管理，以降低物件之間複雜的關係。MVC 包括三個角色：

- Model：定義應用程式的資料和商業規則集合。通常被稱作應用程式的商業邏輯。
- View：定義應用程式的使用者界面。
- Controller：定義應用程式與使用者輸入及模型進行互動的方式。它被稱作應用程式邏輯。

## 4. 研究方法：

大部分的學習內容管理系統，均針對桌上型 PC 的使用者而設計的，這對於目前處在普遍計算環境下的學習者而言，在學習過程中以及系統的使用上，明顯地不足，因此，學習內容管理系統的架構需要被重新設計與實作，並且具有「內文感知」以及「呈現多樣化」的能力，以符合實際的需求。

### 4.1. 第一年計畫研究方法：

#### 4.1.1. 內文感知學習內容管理系統之系統架構

基本上，一個完整的「內文感知學習內容管理系統」必須具有以下幾項子系統與其相關功能：

- Authoring Tool：一個讓 Content Developer 能容易開發教材的編輯工具，因為教材元素包括了文字、影像、音樂、視訊，所以 Authoring Tool 必須要能針對這些元素作處理。
- Content Management System (CMS)：負責管理教材的內容與紀錄其關係，包括 Content Packaging、SCO、SCA (Sharable Content Asset)、Asset 等，以及描述這些教材與其結構關係的 XML 檔案。
- Content Storage：實際儲存學習教材的媒體。
- Content-Sensitive Middleware (CSM)：依據 Client 端的各種 Context，為元素作適當的 Adaptation，並編製成最佳化的版面佈置，再轉成 HTML 或 WML 回應給 Client。

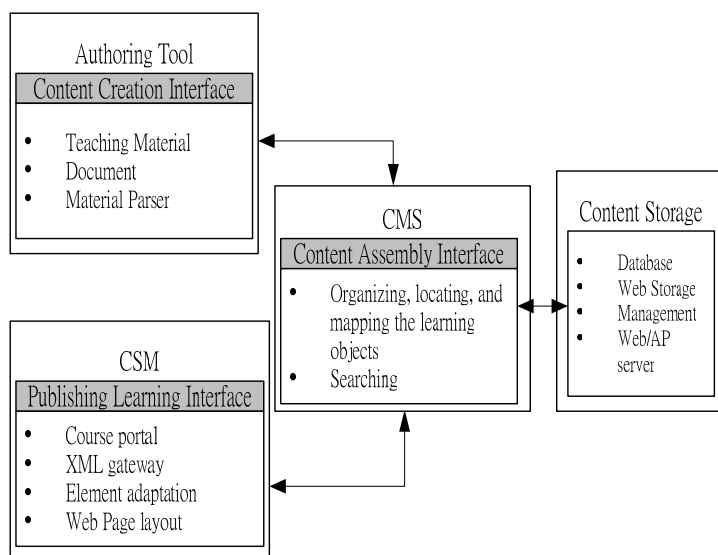


圖 2：內文感知學習內容管理系統架構與子系統交談概念圖

而各子系統之間的交談關係，則如圖 2 所示。在 Authoring Tool 與 CSM 各自有一個 Interface，分別針對不同的 Client User。Authoring Tool 裡的 Content Creation Interface 是給 Content Developer 使用，讓 Content Developer 可以編輯教材；而 CSM 裡的 Publishing Learning Interface 是給 Learner 使用，讓 Learner 可以對 Server 進行存取。這兩者皆透過 Content Assembly Interface 來與 CMS 作溝通，並利用 CMS 來對資料庫進行存取。

#### 4.1.2. 內文感知教材編輯

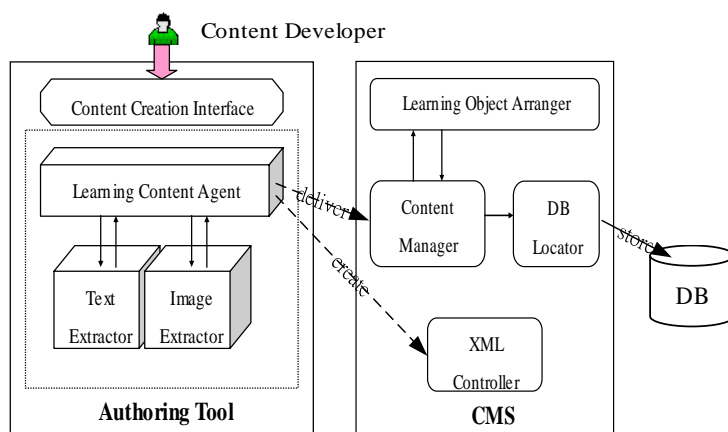


圖 3：CSLCMS 解析靜態文件概念圖

如圖 3，當 Content Developer 透過 *Content Creation Interface* 將手中的教材如 PDF、DOC、PPT 等靜態文件上傳給 Authoring Tool 後，Authoring Tool 會呼叫 *Learning Content Agent* 來解析 (parse) 靜態文件的架構與組成元素。一般來說，靜態文件的組成元素為文字與影像，因此 Learning Content Agent 會使用 *Text Extractor* 與 *Image Extractor* 將這些元素擷取出來，然後依據 SCORM Content Aggregation Model 內所定義的元素層級，呼叫 *XML Controller* 建立一份 XML 文件 (Logical Files) 來描述這些元素的架構與關係。我們將一份文件定義成 Content Packaging (即 Lesson 或 Course)；文件裡的每一頁定義成一個 SCO 或 SCA；每一頁中的文字與影像則定義為最基本的 Asset。



在 XML 建立完成後，Authoring Tool 裡的 Learning Content Agent 會把這些 Asset 交給 Content Manager 來處理。Content Manager 主要有兩個工作，一個是呼叫 Learning Object Arranger 來幫這些元素作分類並定義它們彼此之間的關係，譬如我們有時必須將某段 Text 與某張 Image 看成一組 Unit，這種 Unit 其實是為了「Document Context」而考量。所謂的 Document Context 是說在文件中，某些元素間的關聯性與相依性相當高；遇到這種情形，若系統決定將其中一項元素呈現給 Learner，則另一項元素也必須存在於同一份 Web Page，而且位置必須相近或相鄰，例如圖 4。最後才呼叫 DB Locator 來將這些元素一一存進資料庫中。

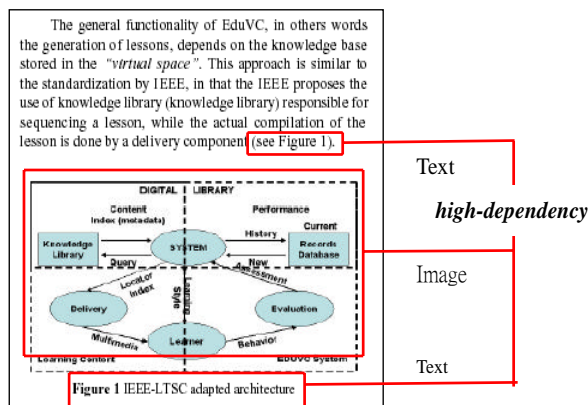


圖 4：元素關係相依示意圖

在實際的做法上，文字元素我們在編製 XML 檔案時，已經直接存進 XML 裡，至於影像元素，我們在 XML 檔案內描述它的屬性（如檔案大小等）與紀錄實際存放於系統中的路徑。而設計 Learning Object Arranger 的目的，是考量到未來系統會持續擴充，加入 Audio 與 Video 元素；同樣地，文字、影像、聲音、視訊等元素間會彼此存在著許多相依關係，因此 Learning Object Arranger 也必須為這些元素進行分類與定義其相關的 Unit。

#### 4.1.3. 版面佈置的辨別



圖 5：文件版面佈置類型

通常在靜態文件的版面佈置上分成三種，分別為「一欄」、「兩欄」、與「三欄」如圖 5，因此我們可以利用各欄之間的空白區域來判斷欄數。在此，我們先定義一些變數如表 1、圖 6，然後再用一個虛擬碼來說明欄數判斷的規則。

表 1：判別欄數所用之變數

名稱	意義
$S$	空白區段或此區段內大部分為空白 (Space)
$C_1$	將文件縱向切為兩等份，中心部分的區段
$C_2$	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
$C_3$	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
$C_{start}$	某區段的起點
$C_{end}$	某區段的終點
$D_{start}$	文件的起點
$D_{end}$	文件的終點
$Thre$	空白區段的門檻值 (Threshold)
$T_1$	文件為一欄格式
$T_2$	文件為兩欄格式
$T_3$	文件為三欄格式
$Type$	文件格式的型態

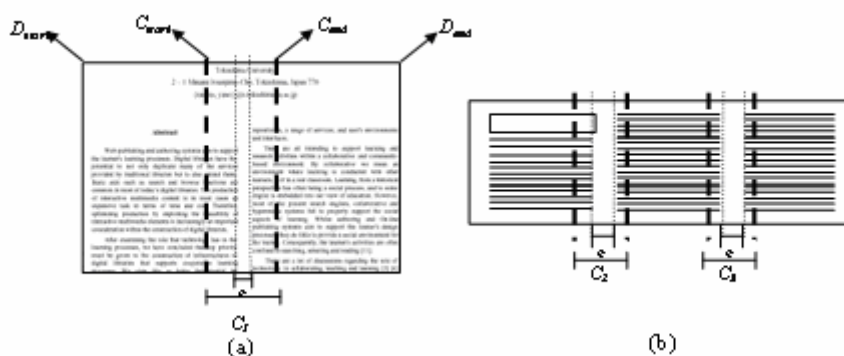


圖 6：變數判斷示意圖 (a)兩欄格式 (b)三欄格式

經由上述變數的定義，我們可以得到  $C = \{ C_{end} - C_{start} \mid C_1, C_2, C_3 \}$ ,  $S \in C$ ,  $C \in [D_{start}, D_{end}]$ ,  $Type = \{ T_1, T_2, T_3 \}$ ，然後用上述的演算法來辨識文件的格式。

#### 4.1.4. 文字的處理

在文字元素的處理上，我們將文字元素擷取出來後，還要進一部作正規化的動作。文字的意義包含兩大類：

- 本文部分：文章的實際內容，這類的文字佔文件的大部分面積。
- 其他部分：包括標題、註解、或其他資訊如論文裡的作者資料，通常這部分的文字會以粗體、置中、或特殊字型來排版。

因此當 Text Extractor 把文字交給 Learning Content Agent 後，Learning Content Agent 必須為這些文字定義其特徵。在實作方法中，我們先將靜態文件轉換成 HTML 格式，然後利用 font tag 裡的 size 屬性，如 `<font size = "2" ></font>`，進行第一步的判斷。然後統計這份文件中 size 出現最多次的數值當作門檻值，等於或小於此數值的文字元素系統認為「本文部分」；反之，若大於此數值的文字元素，且其具有粗體、置中等特性者，則為「其



簡稱 CSM)主要的功能是将一份 LCMS 的學習教材，針對不同行動裝置的設備內文 (Device Context)，提供學習者不同的呈現方式。

在設計上，本文採用 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，我們遵循 SCORM 1.3.1[6]的國際標準，將學習內容分成 Asset、SCO、SCA 以及 Content Packaging，並利用 XML 來對這些元素加以描述以及鏈結他們之間的關係，CSM 之間的子系統模組歸納如下：

- *CSM Interface*：透過 *CSM Interface* 與 *LCMS Interface* 之溝通，以獲得 *LCMS* 所提供的行動學習教材資源。
- *CSM GUI* 及 *Context Detector*：學習者則透過 *CSM GUI* 以獲取行動學習教材，其中，藉由 *Context Detector* 來偵測不同行動裝置之設備內文。
- *Bare Course Storage*：從 *LCMS* 所獲取的教材，將存放於此，以一份行動學習教材為例，主要的教材資源有三種：教材影像 (Slide IMG)、學習內容 (Learning Content) 及多媒體 (Multimedia)。
- *XML Profile Generator*：為了能讓同一份教材於不同的裝置上呈現，XML 文件所具有之通用特性，能達成此一目標。*XML Profile Generator* 分成三個子模組：*Slide IMG Generator*、*Learning Content Generator* 及 *Multimedia Generator*，將於第 5.1 節詳述之。
- *View Adaptor*：透過前述之 *XML Profile Generator*，可產出三份 XML 文件：教材影像文件 (Slide IMG Profile)、學習內容文件 (Learning Content Profile) 及多媒體文件 (Multimedia Profile)，而 *View Adaptor* 主要的作用是，將這三份 XML 文件與 *Context Detector* 所獲取之設備內文作結合，以達最適化。*View Adaptor* 亦分成三個子模組：*Slide IMG Adaptor*、*Learning Content Adaptor* 及 *Multimedia Adaptor*，將於第五小節詳述之。
- *Synchronization Processor*：*Learning Content Adaptor* 針對靜態文件作處理，如投影片之大綱內容；*Multimedia Adaptor* 則針對動態文件來作處理，如老師上課的影音視訊檔案等。藉由 *Synchronization Processor* 將兩者結合，以達到同步播放的功能。
- *XML Parser*：*Synchronization Processor* 整合出來的一份 XML 文件，與 *View Adaptor* 之 *Slide IMG Adaptor* 所產出的『最適化教材影像』，透過 *XML Parser* 將所有的 XML 文件打包成一份。
- *Context-Sensitive XSL* 及 *XSL Proxy*：由 *XML Parser* 所得到的產物，便是一份客制化的行動學習教材，因為是針對不同行動裝置之設備內文所得，故在這裡稱作是 *Context-Sensitive XSL*。此外，*XML Parser* 每次解析出來的 *Context-Sensitive XSL* 將被集中存放於 *XSL Proxy*，方便學習者下次瀏覽時可直接由 *XSL Proxy* 取得。

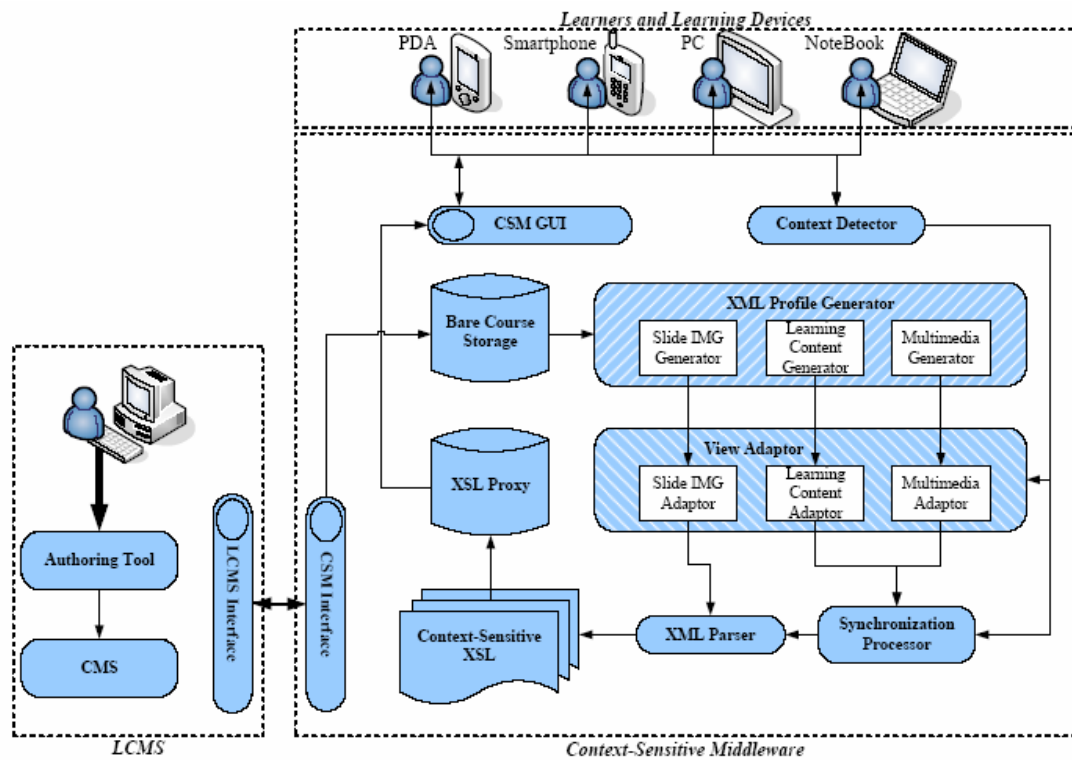


圖 9：行動學習教材呈現系統架構圖

#### 4.2.1. XML Profile Generator

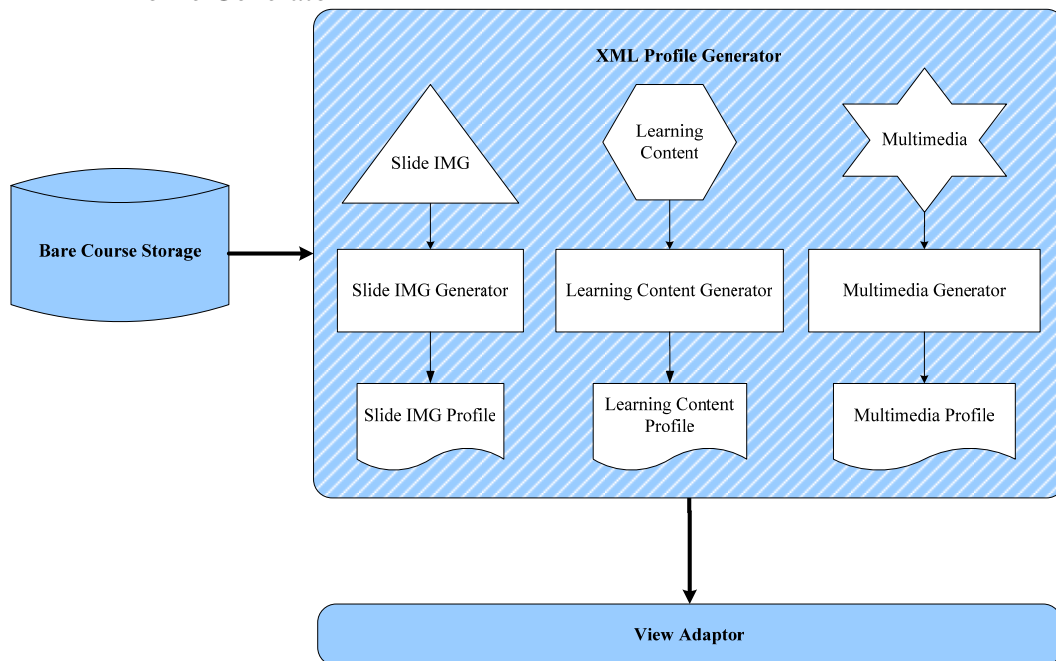


圖 10：CSM 產生 XML 文件概念圖

為了能讓學習者透過各自的行動裝置來獲得學習教材，首先，透過 *XML Profile Generator* 將所有的教材資源轉換成 XML 文件，如圖 10 所示。

以 *LCMS* 提供的『多媒體輔助投影片』之學習教材為例，該教材主要的功能在於，學習者能夠一邊瀏覽投影片，一邊觀賞老師上課的情形，此目的在於幫助學習者快速地掌握課程重點。由於行動裝置的環境限制，例如：小面積之螢幕大小 (Screen Size) 限制，故原始的投影片教材，無法完整地呈現於行動裝置上。為了解決此問題，*CSM* 將該教材分成三種教材資源：教材影像 (Slide IMG)、學習內容 (Learning Content) 及多媒體 (Multimedia)。為了保持教材的原貌，其作法是將每一張投影片 (Slide) 轉換成影像，也就是所謂的教材影像，最後呈現於行動裝置，則是一份『最適化教材影像』，這裡所指的最適化，意指其影

像大小剛好符合行動裝置的螢幕大小。

透過『最適化教材影像』，便能夠得知一份教材的全貌，因為每一張教材影像，即是每一張投影片的縮小版。不過，卻產生另一個問題：學習者無法了解該教材的內容。為了解決此問題，藉由 CSM 來擷取投影片之大綱內容，也就是所謂的學習內容，其好處在於投影片的重點，透過學習內容便足以表示。

由上述可知，靜態文件的問題已獲得初步解決。由於該教材具有多媒體輔助之功能，故還需針對動態文件的部分來作處理，也就是所謂的多媒體。最後，透過 *Synchronization Processor* 及 *XML Parser* 將這三種教材資源整合在一起，即是一份完整的『多媒體輔助投影片』教材。

#### 4.2.2. View Adaptor

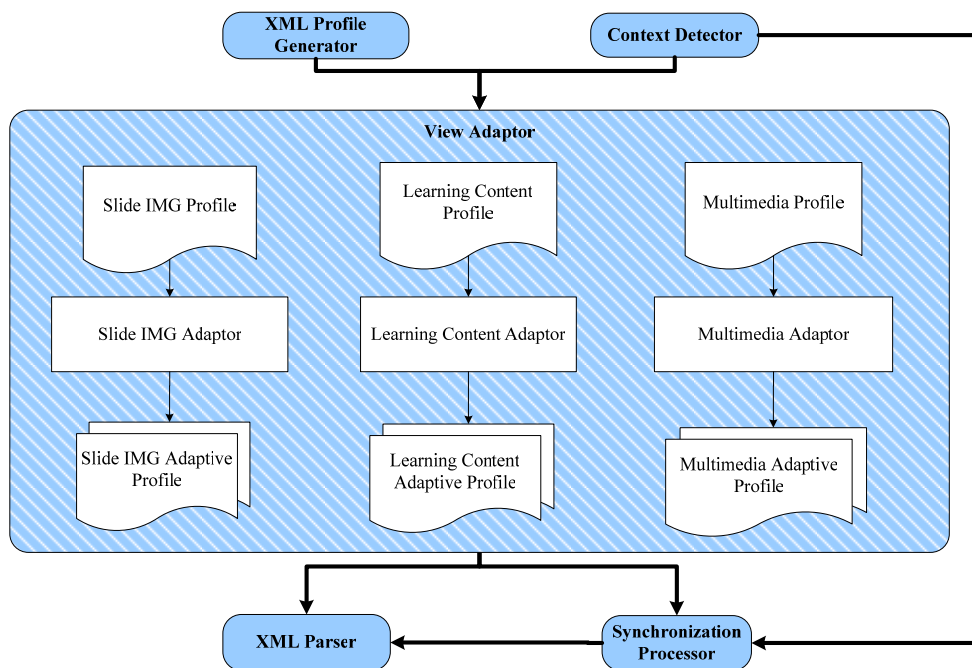


圖 11：CSM 最適化 XML 文件概念圖

在前一節，說明了如何透過 *XML Profile Generator*，產出三種教材資源的 XML 文件，分別為教材影像文件、學習內容文件及多媒體文件。接下來，則描述這三份 XML 文件，藉由 *Context Detector* 與 *View Adaptor* 來轉換出最適化的文件 (Adaptive Profile)，最後再整合出一份客制化的學習教材，以達到客制化的呈現，如圖 11 所示。以下 4.3 節至 4.5 節將更深入探討之。



#### 4.2.3. 設備內文之偵測

表 2：設備內文之各種限制

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.	是否支援聲音檔播放
12.	是否支援影像檔播放

要達到客制化的要求，首先必須考量設備內文 (Device Context) 的各項限制。針對一份『多媒體輔助投影片』之學習教材，整理出影響性較大的限制，如表 2 所示[14]。

為了能順利處理表的所有限制，透過 R. Mohan、J. R. Smith 與 C. S. Li 所提出的基本方法 [15]，並考慮到目前的新機種可用資源較多，為了能讓這些新機種發揮最佳效能，因此也擴充了客制化的機制。

#### 4.2.4. 教材影像與學習內容之連結

藉由教材影像的呈現，學習者能夠掌握一份教材的全貌，卻無法了解其細節，透過教材影像與學習內容之連結，便能夠解決此問題。當學習者點選其中一張教材影像，則對應到該影像所包含的內容，當然學習者也能夠直接切換到『學習內容模式』。其作法就是將學習內容之標題 (CourseTitle) 作一個『雙向超鏈結』至教材影像，如圖 12 所示。

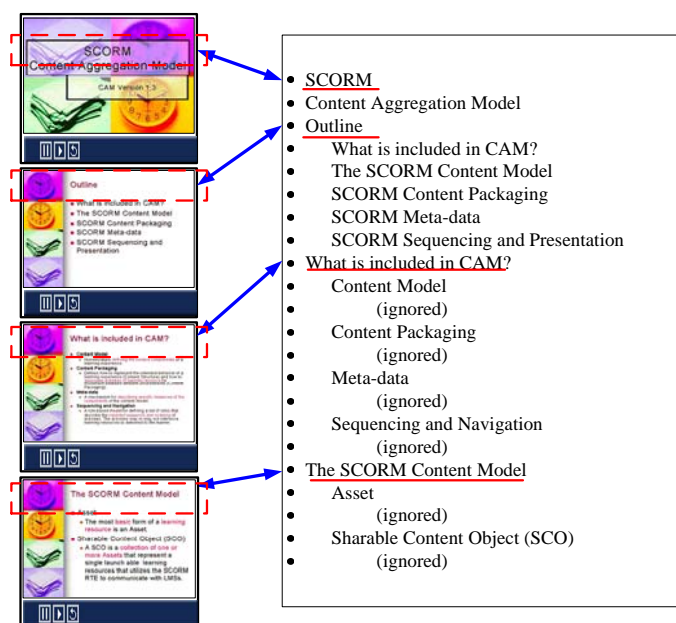


圖 12：教材影像 vs. 學習內容連結示意圖

#### 4.2.5. 教材之客制化

針對表所歸納的各種限制，設計出專門描述設備內文的XSL，存放於*Context Detector*中，當學習者作線上學習的同時，*Context Detector*偵測出該行動裝置的設備內文，以挑選出最適合的XSL，與*XML Profile Generator*產出的XML文件作結合。最後再根據行動裝置所支援的瀏覽器種類，例如：HTML、WML或XHTML等，解析成一份客制化的行動學習教材，在這裡稱為一份*Context-Sensitive XSL*，存放於*XSL Proxy*中，若學習者再次使用同一行動裝置來學習，則直接透過*XSL Proxy*取出。

### 4.3. 第三年計畫研究方法：

學習者的特徵包含了兩個部分。一個是學習風格(Learning Style)，而另一個是學習的行為。為了定義每一個學習者的學習風格，我們使用 Felder 及 Silverman 所提出的學習風格模型(learning style model) [16]。根據每位學習者的學習風格，我們給與個別調適後的學習內容以及資源。除了學習風格外，我們也記錄了可以反應學習者的學習方法及態度等的學習行為，如：學習時間、休息時間、問題形式等等。而在本計畫裡，我們將這些行為稱為學習行為(Learning Activity)。我們收集、分析這些學習資料，並且使用這些資料來建構完整的學習特徵資訊。

#### 4.3.1. 學習風格(Learning Style)

學習風格模組根據以下四種維度來分類個人偏好：(1)主動及被動維度；(2)感覺及主動維度；(3)視覺及言語維度；(4)連續及整體維度。每一個維度代表不同的個性、觀念及學習偏愛。當學習者學習同一份教材時，被分類成不同的種類的學習者會有不同的反應。因此，根據每個學習者的形式提供適合的教材顯得相當重要。我們考慮了偏好、特徵、個性，來達到讓教材更加的有吸引力，並且讓學習者更有效率的學習。學習風格索引 (ILS, Index of Learning Styles) 是由44種問題所組成的，用來讓 Felder-Silverman 模型的四個維度來評估偏好。學習者在學習前，必須先回答這44個問題。在回答完這些問題後，便可以將學習者進行分類。

##### 4.3.1.1. 學習活動(Learning Activity)

每一個學習者都有屬於他們自己獨特的喜好以及能力，這也導致他們有不同的學習路徑、行為、方法等。我們稱這些行為、方法叫做學習活動(Learning Activity)。系統記錄使用者所有的活動，譬如學習者閱讀一個章節所花的時間、他有什麼樣的問題、閱讀相關資料時所花的時間、使用偏好的連結多少次等等。這些行為都由學習特徵提取器(Learner Characteristic Extractor)所提取出來，並且轉換為學習特徵文件。

這份文件描述關於「活動類型」、「活動內容」、「活動發生次數」以及「活動發生時間」。一般來說，從頭到尾學習一份教材時，學習活動指出學習者的「學習內容」、「相關資料」、「問題」以及「答案」。因此，我們定義了四個種類的學習活動：

- 問題活動(Question Activity): 此活動內容描述問題的內容。
- 問題活動(Answer Activity): 此活動內容描述答案資源、答案作者、答案的部分內容以及答案的來源。
- 連結活動(Link Activity): 此活動內容描述種類(關鍵字、相關章節)、名字、以及連結的位址。



- 休息活動(Rest Activity): 此活動內容描述休息的種類。  
另外，我們也記錄以上四種活動所發生的次數及時間。

#### 4.3.1.2. 學習特徵擷取器(Learner Characteristic Extractor)

當學習者開始學習後，系統便開始偵測所發生的學習活動。學習特徵擷取器(Learner Characteristics Extractor)進行收集，並且建構屬於學習者的學習特徵文件。經由這些文件，我們可以輕易的比對出在哪個部分的教材會發生什麼樣的學習活動。

在本計畫裡，我們使用LCMS所提供的投影片教材來舉例。學習特徵文件經由時間安排、章節序列、以及投影片的順序所組成。圖13是學習者X的學習特徵文件範例。首先，這份文件記錄學習者X在經過 ILS 44 個問題測試後學習風格。學習者X被歸類為 Active / Intuitive / Visual / Sequential 種類的學習者。第2、3行表示X是正在學習資料結構的第四章。第5、6行是紀錄開始閱讀第六章投影片的時間點。第6至16行是描述問題活動的發生。學習者X提出關於「First in, first out」的問題。另外，學習者X過去時常閱讀同一份答案兩次(答案活動)。內容的部分以及答案的作者亦描述於此份文件內。至於連結活動的部分，是記錄在17至26行的地方。學習者X曾經分別點選過「關鍵字連結」以及「相關章節連結」。

```

1 <LearnerProfile Type = "Act / Int / Vis / Seq">
2   <Course Name = "Data Structure">
3     <Chapter Name = "Queue" No = "4">
4       <Slide No = "6">
5         <DateTime>2004-06-25 03:28:43</DateTime>
6         <Question>
7           First in, first out
8           <DateTime>...</DateTime>
9           <Answer ref = "Forum" src = "http://.....">
10            <From>Expert 1</From>
11            <Content> First in, first out means...</Content>
12            <Click is = "True" Num = "2">
13              <DateTime>2004-06-25 03:50:12</DateTime>
14              <DateTime>2004-06-25 04:30:16</DateTime>
15            </Answer>
16          </Question>
17          <Link>
18            <Keyword Name = "Stack" ref = "http://..">
19              <Click is = "True" Num = "1" >
20                <DateTime>2004-06-25 04:20:16</DateTime>
21              </Keyword>
22            <ChapterRelate ref = "http://..">
23              <Click is = "True" Num = "1">
24                <DateTime>2004-06-25 04:44:28</DateTime>
25            </ChapterRate>
26          </Link>
27        </Slide>
28      </Chapter>
29    </Course>
30 </LearnerProfile>

```

圖 13. 學習者X的學習特徵文件範例

#### 4.3.2. 資料挖掘過程(Data Mining Process)

在章節4.3.1裡，我們展示了如何使用學習風格及學習活動來建構一個完整的學習特徵資料庫。藉由使用這些資料，資料挖掘過程分析並且進行挖掘來算出大部分學習者在同一份教材所發生的方法及行為。另外，我們可以找出有意義的學習方法、教材的相關資料，以及對於延伸、完成學習內容有幫助的資料。再者，我們可以使用資料挖掘的分析結果來將每位學者的學習方法、學習路徑進行分類。

#### 4.3.2.1. 資料前端處理器(Data Preprocess)

首先，資料前端處理器從資料庫裡收集同一份教材的所有學習特徵文件，並且消除不必要的資料，如活動內容的細節。再來，我們分類不同的學習特徵至不同集合。活動形式的集合可以用  $I = (Q, A, L, R)$  來表示。然後，每一個活動型式的集合可以用以下方式來表示：

- $Q = \{ q_1, q_2, q_3, \dots, q_m \}$ ， $q_j$  表示問題活動，其中  $1 \leq j \leq m$ ；
- $A = \{ a_1, a_2, a_3, \dots, a_m \}$ ， $a_j$  表示答案活動，其中  $1 \leq j \leq m$ ；
- $L = \{ l_1, l_2, l_3, \dots, l_k \}$ ， $l_j$  表示連結活動，其中  $1 \leq j \leq k$ ；
- $R = \{ r_1, r_2, r_3, \dots, r_p \}$ ， $r_j$  表示休息活動，其中  $1 \leq j \leq p$ 。

根據ILS的定義，總共有四種學習風格維度。我們用  $LS = (LS_1, LS_2, LS_3, LS_4)$  來表示學習風格，其中  $LS_1$  表示在  $i_{th}$  維度的學習風格。

再來，我們根據學習活動時間序列來轉換每位學習者的序列資料。舉例來說，學習者  $Y$  的學習活動序列用  $Y = \{(LS_1, LS_2, LS_3, LS_4), (y_1, t_1), (y_2, t_2), \dots, (y_n, t_n)\}$  來表示。其中， $y_j \in I$ ， $t_j$  代表  $y_j$  發生的時間，而  $1 \leq j \leq n$ ， $t_{j-1} \leq t_j$ ， $2 \leq j \leq n$ 。

Lid	Sequence
10	$\{(Act, Int, Vis, Seq), (s_1, 1), (s_2, 10), (q_2, 15), (a_2, 22), (s_3, 25)\}$
20	$\{(Act, Int, Vis, Seq), (s_1, 4), (q_1, 6), (a_1, 7), (s_2, 20), (s_3, 24)\}$
30	$\{(Rfl, Int, Vis, Glo), (s_1, 3), (q_1, 5), (a_1, 7), (s_2, 19), (s_3, 26), (q_3, 28), (a_3, 29)\}$
40	$\{(Rfl, Sen, Veb, Glo), (s_1, 2), (q_1, 3), (a_1, 3), (s_2, 14), (l_1, 15), (s_3, 19)\}$

表 3. 序列資料的表格。

經過轉換後，每一個學習教材都會有一個序列的表格。拿LCMS的學習教材投影片來當例子。序列資料的表格，如表3，其中Lid是學習者的ID，而  $j_s$  表示第  $j_m$  張的投影片。

#### 4.3.2.2. 序列樣版挖掘及結果(Sequential Patterns Mining and Mining Result)

學習特徵的資料在前一個階段已經轉換成依序列數字的資料了。每一個序列資料代表一個教材的學習活動序列。在本計畫理所提出得學習活動都是連續並且相關的。當學習者在學習的過程有問題時，那麼問題活動(Question Activity)便發生了。假如問題已經被發問過並且已經有人解答此問題時，答案活動(Answer Activity)便會發生。

在每一個活動裡考慮學習活動序列以及時間間隔是重要的。大部分學習者的學習間隔都與教材設計者有出入，因此，時間間隔的設計可以讓一份教材有依據而進行較適切的修改。然而，大部分序列樣版挖掘的演算法，包含我們先前所提的 Apriori[20]、PrefixSpan[21] 演算法，都沒有將時間間隔納入考量。而 I-Apriori 以及 I-PrefixSpan 演算法[19]，便是針對此部分作改進。這兩則演算法便是用來挖掘時間間隔序列樣版(Time-Interval Sequential Pattern)。為了找出時間間隔序列樣版，本計畫的方法是基於 I-Apriori 以及 I-PrefixSpan 演算法。序列樣版的挖掘結果由學習活動序列樣版來表示。學習內容可以藉由挖掘結果來進行延伸。此外，序列樣版也可以當作有相同學習風格分類的學習模組。

#### 4.3.3. 學習內容描述(Learning Content Description)

在我們的資料庫中，有許多的學習教材是依據 SCORM 的標準所設計的。每一份教材都有一份描述的檔案，包含作者、投影片數量、表格的是數量、圖片的數量，以及與教材相關的資訊。每份學習教材都是由不同的作者所設計的。因此，每份教材都會有各自的特徵。為了提供調適後的教材給不同的學習者，教材的內容便要依據符合學習者的描述檔案來組織。舉例來說，喜歡利用圖片學習的學習者，我們就要提供多一點的圖片，勝過喜歡用文字學習的學習者；敏感的學習者，我們就要給予每張投影片較長的學習間隔[17][18]。

我們拿LCMS所提供的教材資源當範例，學習描述檔案由章節以及投影片的順序來建構。經過資料挖覺得過程後，序列樣版的結果便延伸至描述檔案裡。圖14是「Data Structure」的描述檔案範例。第2行至第5行是教材的基本描述，譬如檔案資源、檔案名稱以及作者的名字。第6行是描述相關資料(第四章節「Queue」)。第8~12行說明有20張投影片、5張圖片以及6個表格。紅色方塊區域是我們所加入的資料挖掘結果。我們可以發現問題活動(Question Activity)通常發生在第6章投影片。大部分的問題是關於「first in, first out.」。這部分是最適合專家來進行調適教材的部分。大部分的學習者通常點選關鍵字「Stack」的連結。因此，當學習這份教材時，假如學習者詢問相同問題時，便可以立刻提供答案。另外，當學習者學習到第6章投影片時，便可以建議使用者學習「Stack」的資料。

```

1 <LearningContentDescription>
2   <Source>http://...</Source>
3   <Type>Computer Science</Type>
4   <Instructor>Dr. Smith</Instructor>
5   <Course Name = "Data Structure">
6     <Chapter Name = "Queue" No = "4">
7       <smil>...</smil>
8       <NumberInfo>
9         <slid>20</slid>
10        <fig> 5</fig>
11        <tabl>6</tabl>
12      </NumberInfo>
13      <Slide No = "6">
14        <smil>...</smil>
15        Queue is.....

16        <extend>
17          <Question>First in, first out</Question>
18          <Answer ref = "Forum" src = "http://.....">
19            <From>Expert 1</From>
20            <Content> First in, first out means...</Content>
21          </Answer>
22          <Link>
23            <Keyword Name = "Stack" ref = "http://.." >
24            <Click TotalNum = "60" IsImportant = "True">
25          </Link>
26        </extend>

28      </Slide>
29    </Chapter>
30  </Course>
31 </LearningContentDescription>

```

圖 14. 「Data Structure」課程的描述檔案範例。

#### 4.3.4. 學習內容組織器(Learning Content Organizer)

學習內容組織器根據學習者的特徵資料以及學習內容描述檔案的挖掘結果來組織調適的學習教材。當學習者需要一份教材時，學習內容組織器會收集教材的描述檔案，並且比對學習者的風格。學習內容組織器會找出最適合的教材並且重組學習內容。舉例來說，針對視覺型的學習者給予圖片形式的教材，針對主觀型的學習者給予觀念型教材。另外，學習內容的建議可以經由學習活動的序列樣版來提供。這些建議的增加是依照描述檔案的挖掘結果。我們拿LCMS所提供的投影片教材來舉例。每一張投影片的比例、時間間隔可以由相同學習風格的序列樣版來進行修訂。經由分析、比較以及修改後，學習內容可以更加的適合每位學習者的學習特徵。附帶一提，學習者也可以更有效率的學習了。

## 5. 結果與討論：

### 5.1. 第一年計畫結果與討論：

當學習者透過學習設備與系統連線時，系統首先會偵測與分析出學習設備的規格，包括了 MIME 型態、螢幕資訊、記憶體大小等，並以此作為內容呈現的排版依據，以下圖 15 所示。

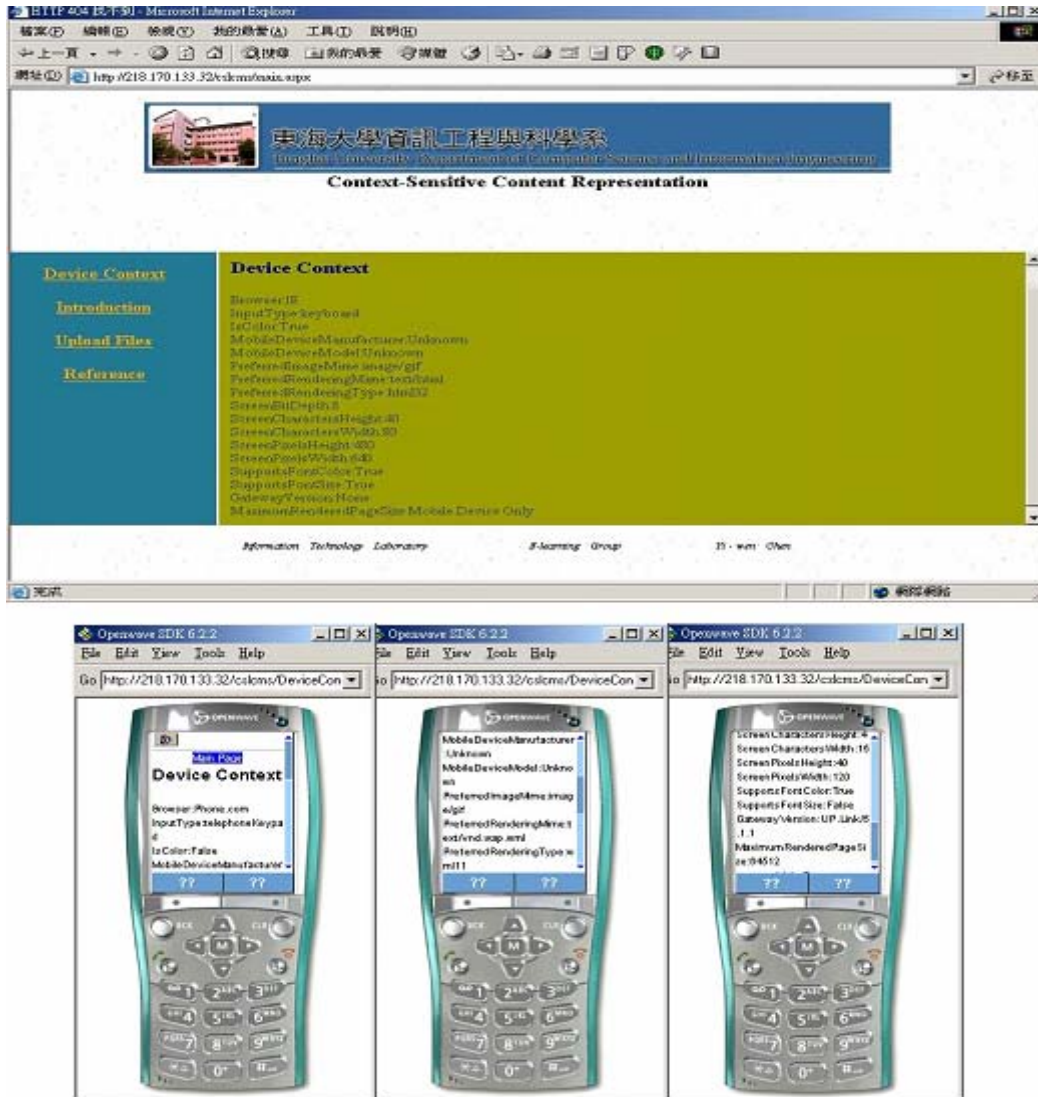


圖 15：實驗結果一

當使用者透過不同的學習設備連上網站後，系統能根據不同的 Context 提供不同的 View 給用戶端，譬如圖 16 是使用者連結「Introduction」選項的範例，當學習者使用 IE 來閱讀文件時，系統會給予較豐富的內容與樣式；而當學習者使用手機或 PDA 來瀏覽時，則給予其所能接受的排版內容。



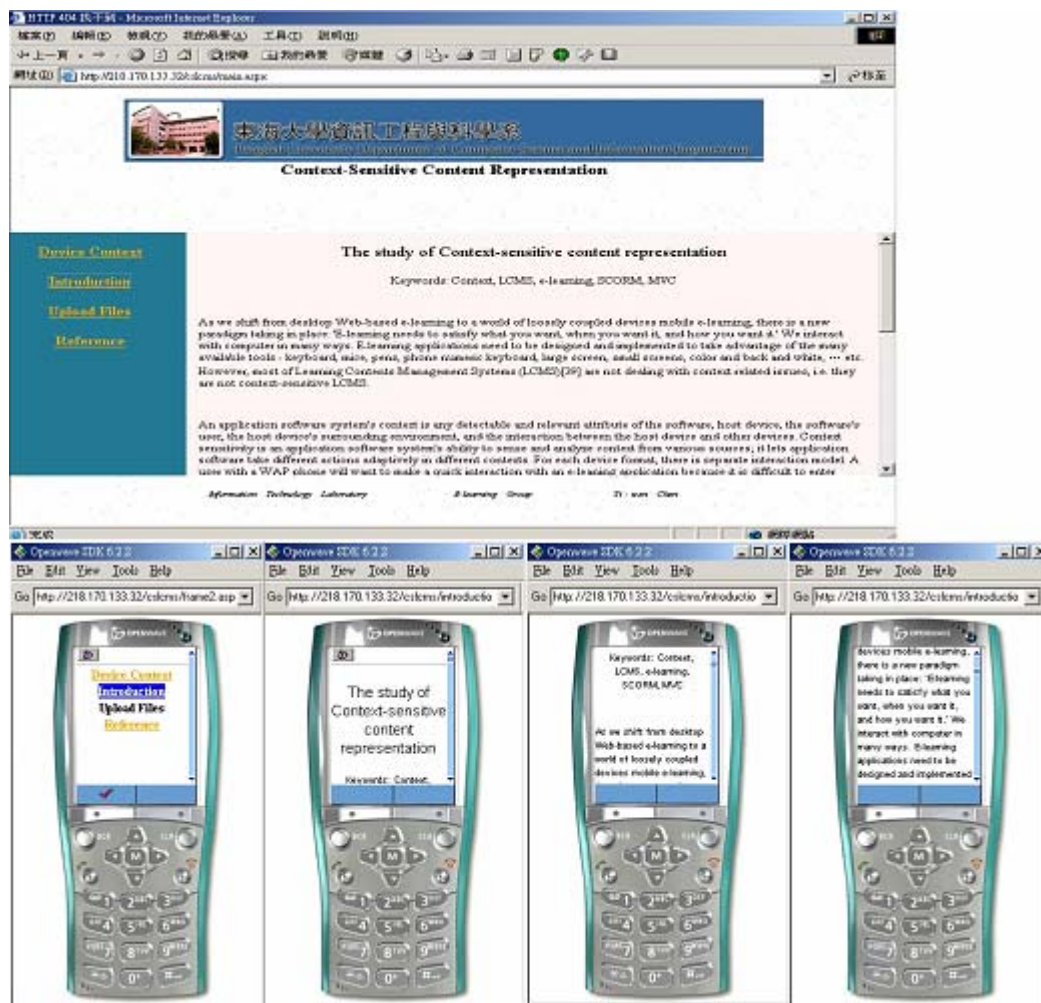


圖 16：實驗結果二

### 5.2. 第二年計畫結果與討論：

目前的學習內容管理系統，雖然能夠整合靜態文件與動態文件，不過學習平台卻依然是以個人電腦為主，而非針對小型的行動裝置。在本文所提出的 *CSM*，先利用 *XML Profile Generator* 來產生三種教材資源的 XML 文件，而後 *View Adaptor* 再透過 *Context Detector* 所偵測到的設備內文，轉換出最適化的 XML 文件，最後打包成一份客制化的行動學習教材，成功的提升教學資源的可再用性。在未來，除了針對『內容呈現』的議題作更深入的研究外，還將提到『互動式學習』(Interactive Learning) 與『離線學習』(Offline Learning)，以解決數位學習領域所會面臨到的各種問題。

### 5.3. 第三年計畫結果與討論：

本計畫擷取指導者與學生的學習特徵，進而提供完整的學習教材。我們記錄每位學習者的活動，並利用資料探勘(Data mining)的技術來挖掘每個時段學生的學習活動循序樣式(Learning Activity Sequential Patterns)。在挖掘出學生的個人學習特徵後，指導者便可以利用我們所挖掘的資料來製作教材內容。在未來，我們將會考慮更多有關於學習行為與能力，譬如：PVRT、學習能力測試等等。此外，利用資訊擷取技術(Information Retrieval)亦可以用於比較學習者特徵以及學習內容，讓學生可以更加有效率的學習。

## 6. 參考文獻：

1. LCMS, Available to <http://www.elearnspace.org/LCMS.htm>
2. Joseph Williams, "The Hard Road Ahead for WAP", IT Professional, Vol. 2, No. 5, September/October 2000.
3. The WAP Forum. <Http://www.wapforum.org/>
4. Shareable Content Object Reference Model Initiative (SCORM), <http://xml.coverpages.org/scorm.html>
5. 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.
6. Advanced Distributed Learning (ADL), "SCORM Content Aggregation Model Version 1.3.1," Available at [ADLNet.org](http://ADLNet.org), July, 2004.
7. 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.
8. Jin-Tan David Yang and Chun-Yen Tsai, "An Implementation of SCORM-compliant Learning Content Management System – Content Repository Management System," *Proceedings of the 3<sup>rd</sup> IEEE International Conference on Advanced Learning Technologies (ICALT' 03)*, 2003.
9. 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 17<sup>th</sup> International Conference on Advanced Information Networking and Applications (AINA' 03)*, 2003.
10. 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.
11. 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.
12. W3C, Available to <http://www.w3c.org>
13. T. Reenskaug, "Working With Objects: The OOram Software Engineering Method," Manning Publications, 0-13-452930-8, Per World and Odd Arild Lehne, 1996.
14. 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.
15. 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.
16. R.M. Felder and L.K. Silverman, "Learning and Teaching Styles in Engineering Education," *Engineering Education*, Vol. 78, No. 7, 1988, pp. 674-681.
17. R.M. Felder, and L.K. Silverman, "Learning and teaching styles in engineering education,"



*Engineering Education*, Vol. 78, No. 7, 1988, pp. 674-681.

18. R.M. Felder, "Reaching the second tier: Learning and teaching styles in college science education," *Journal of College Science Teaching*, Vol. 23, No. 5, 1993, pp. 286-290.
19. Y.L. Chen, M.C. Chiang and M.T. Ko, "Discovering time-interval sequential patterns in sequence databases," *Expert Systems with Applications*, Vol. 25, No. 3, 2003, pp. 343-354.
20. R. Agrawal, and R. Srikant, "Mining sequential patterns," *the Proceedings of 1995 International Conference on Data Engineering*, 1995, pp. 3-14.

### 本計劃所發表的論文

1. **C.W. Chu**, Y.W. Chen, and J.N. Chen, "Context-sensitive Content Representation for Static Document," *the Proceedings of APSEC 2004*, Nov. 30 – Dec. 3, 2004, pp. 719-725.(EI)
2. C. W. Chu, H. X. Lin, J. N. Chen, X. Y. Lin, "Context-Sensitive Content Representation for Mobile Learning," *the Lecture Notes in Computer Science (LNCS)*, Jul. 31 – Aug. 3, 2005. pp. 349-354.(SCI)
3. X.Y. Lin, **C.W. Chu**, J.N. Chen, "Learning Characteristic Extraction and Content Suggestion," *the Proceedings of Sixteen Workshop on Object-Oriented Technology and Applications*, Sep. 15, 2005, pp. 19-25.