

國立交通大學

資訊科學與工程研究所

碩士論文

多媒體內容在手持行動裝置之
情境及內容協調與呈現



The Context Aware and Content Adaptation of
Multimedia Presentation on Handset Device

研究生：柯政邦

指導教授：陳登吉 教授

中華民國九十六年七月

多媒體內容在手持行動裝置之情境及內容協調與呈現

The Context Aware and Content Adaptation of Multimedia Presentation on Handset Device

研究生：柯政邦

Student：Cheng-Pang Ko

指導教授：陳登吉

Advisor：Dr. Deng-Jyi Chen



Submitted to Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in
Computer Science
July 2007
Hsinchu, Taiwan, Republic of China

中華民國九十六年七月

多媒體內容在手持行動裝置 之情境及內容協調與呈現

學生：柯政邦

指導教授：陳登吉 博士

國立交通大學資訊工程學系碩士班

摘要

隨著科技的迅速發展，手持行動裝置提供越來越多符合使用者需求的效能及服務，也因為如此，以前許多只能在個人電腦上才能使用的功能，現在的手持行動裝置都已完整地提供，人們變的不再過度依賴個人電腦，這也驅使了一股發展潮流，如果能使自己的多媒體產品移植到手持行動裝置上呈現，將會另一個發展的商機。但是一般針對個人電腦設計的多媒體產品，由於諸多限制，如果要轉移到手持行動裝置上呈現，必須重新設計呈現的方式和流程，這項工作將會花費相當多的人力和時間，造成業者的一大負擔。

降低個人電腦和手持行動裝置間轉換的時間和人力是本研究主要的目的，利用情境感知(Context Awareness)方式，分析對於手持行動裝置有影響的各項因素，讓使用者感覺舒適的呈現畫面為前提，在多媒體教材協調中，如何顧及教材內容的一貫性和連續性，不因為協調(Content Adaptation)而改變，是本研究一大重點。

延伸在《樣版式多媒體內容在手持行動裝置之情境及內容協調與呈現—以樣版式多媒體英文試題為例子》[3]的研究成果，以動態及靜態的混和方式讓空間和時間的使用最佳化。改進 XMG 播放器，使得播放器能呈現 PNG 格式的多媒體檔案，而且對於製作教材常用的動畫功能，也在本研究中新增入播放器中，使在轉換協調過程中，更有效、不失原意的呈現給使用者。

The Context Aware and Content Adaptation of Multimedia Presentation on Handset Device

Student: *Cheng-Pang Ko*

Advisor: *Dr. Deng-Jyi Chen*

**Department of Computer Science and Information Engineering
National Chiao Tung University**

Abstract

With the increasing development of the technology, more and more functions and services are provided by mobile devices based on user's needs. Therefore, many functions which only can be used on personal computer before, for now, they can be provided completely on mobile devices already. Nowadays, people do not rely on their personal computers so much as before, and this is the motivation to force the trendy of improvement. If the multimedia contents can be applied to mobile devices, there will be a business opportunity. However, normally the functions which are designed for personal computers have many restrictions, those functions have to be redesigned its presentations and processing to be applied on mobile devices. This work will be a great burden for the suppliers, because this consumes lots of human resources and time.

The main purpose of this research is to reduce transferring time and the human resources for the exchange between personal computers and mobile devices. Context Awareness is the method to analyze the elements which can affect the mobile devices. The main point for this research is to know how to maintain the consistence and the continuity of the teaching materials of multimedia materials on condition that the interfaces make users comfortable without being changed due to the Content Adaptation.

According to the result in the research [3], we improve XMG player allows PNG format multimedia files could be presented on the player. Furthermore, as to the animation functions which are usually used in some materials are added to the XMG player in this research. To do this, it will improve the transferring processing and will be presented to the end-users efficiently and completely.

致謝

首先要感謝的是我的指導教授陳登吉老師，對於我們耐心的教導，無論是課業、研究上，當遇到問題和走的方向有錯誤時，老師都會細心地指引導正，讓我們不會在為學的路上迷失方向，而最重要的是，老師對於我們的生活上，給予我們很大的幫助，關心我們的健康，讓我們能專心地做研究。

再來要感謝的是一起在實驗室為了研究而努力的同學們，在課業上相互的幫助，有問題的時候，能適時的給予我幫助，而在做研究上，遇到瓶頸時能給我很多有效的建議，讓我能夠完成這項研究。

最後我要感謝在背後默默支持我的父母家人們，從我第一天進入學校時，就給予我無限的支持，讓我沒有任何需要擔心的地方，也因為這樣才有今天的學歷及知識，我也會一直努力下去，不浪費他們對於我的期待與用心。



表目錄

表 1-1 全球及美國 PDA 數量預估（單位：千台）	1
表 2-1 各元件代表之意義	6
表 2-2 靜態轉換與動態轉換的差異性	9
表 3-1 未改編協調內容伺服器環境	17
表 3-2 已改編協調內容伺服器環境	17
表 4-1 資料庫欄位	27
表 5-1 R_w, R_h 運算規則	35
表 5-2 中文字型在桌上型電腦上顯示規格	37
表 7-1 與前系統相異與改良部分	68



圖目錄

圖 2- 1 不同情境所對應的設計方式	7
圖 2- 2 決策引擎(decision engine).....	8
圖 2- 3 樣版選擇演算法	10
圖 2- 4 文字斷行演算法	10
圖 2- 5 圖片縮放比例	11
圖 2- 6 選擇合適的呈現	11
圖 2- 7 CxImage 設計架構.....	13
圖 2- 8 XMG 描述檔範例	14
圖 3- 1 內容改編協調流程	16
圖 3- 2 教材教學流程圖	20
圖 3- 3 原始網頁與 PDA 顯示範圍比較	21
圖 3- 4 畫面切割完成後結構	22
圖 4- 1 系統架構圖	23
圖 4- 2 下載功能運作流程圖	24
圖 4- 3 主要協調部分之架構	25
圖 4- 4 上傳功能運作流程圖	26
圖 4- 5 內容資料庫架構圖	27
圖 4- 6 PDA 下載教材流程圖	28
圖 4- 7 描述檔架構	29
圖 5- 1 演員描述檔	30
圖 5- 2 互動式劇情	30
圖 5- 3 解譯描述檔流程	31
圖 5- 4 Vector 結構.....	32
圖 5- 5 單純等比例協調結果	33
圖 5- 6 文字放大顯示後的等比例協調	34
圖 5- 7 圖片高度寬度以不同比例縮放	34
圖 5- 8 以 R_w, R_h 轉換後的圖片	35
圖 5- 9 座標比例縮放情形	36
圖 5- 10 圖片協調結果	36
圖 5- 11 中文字型的計算	38
圖 5- 12 文字覆蓋畫面	39
圖 5- 13 文字的協調	41
圖 5- 14 背景色對文字顯示的影響	41
圖 5- 15 判定背景色協助文字協調	43
圖 5- 16 遷就高度的影像協調	44
圖 5- 17 影像的協調	45

圖 5- 18 多個影像協調步驟	46
圖 5- 19 變更 XMG 描述檔格式	47
圖 5- 20 動畫演員繪圖流程圖	48
圖 5- 21 協調兩個主要的步驟	49
圖 5- 22 劇情管理員	49
圖 5- 23 協調系統的分頁規則	50
圖 5- 24 發生分頁後的情形	51
圖 5- 25 原始的網頁內容	52
圖 5- 26 每一個頁面所需要的分頁數量	53
圖 5- 27 將頁面中物件的連結目標確實完成	54
圖 5- 28 協調每一個頁面	54
圖 5- 29 交通大學網頁協調後結果	55
圖 5- 30 Flash 簡略檔案結構	56
圖 6- 1 管理者登入畫面	57
圖 6- 2 選擇教材清單畫面	58
圖 6- 3 要求下載教材畫面	58
圖 6- 4 初步協調結果並檢閱	59
圖 6- 5 列出準備上傳的教材編號與預覽圖	59
圖 6- 6 上傳成功畫面	60
圖 6- 7 登入及啓始畫面	60
圖 6- 8 教材清單與選擇下載	61
圖 6- 9 下載過程與等播放過程	61
圖 6- 10 選取教材播放	62
圖 6- 11 循序播放教材	62
圖 6- 12 已製作完成教材範例一畫面	63
圖 6- 13 範例一協調後畫面	63
圖 6- 14 已製作完成教材範例二畫面	64
圖 6- 15 範例二協調後畫面	64
圖 6- 16 原始影像教材畫面	65
圖 6- 17 協調影像教材完成後畫面	65
圖 6- 18 原始教材連結情況	66
圖 6- 19 結構協調的範例	67
圖 7- 1 協調完教材架構圖	69

目錄

摘要.....	i
Abstract.....	ii
致謝.....	iii
表目錄.....	iv
圖目錄.....	v
目錄.....	vii
一、緒論.....	1
1.1 手持行動裝置應用之發展.....	1
1.2 研究動機與目標.....	2
1.3 研究方法與步驟.....	2
1.4 研究範圍.....	3
1.5 章節概要.....	4
二、相關研究.....	5
2.1 情境感知改編協調.....	5
2.1.1 情境感知(Context Aware).....	5
2.1.2 內容改編協調(Content Adaptation).....	9
2.2 樣版式多媒體內容協調.....	10
2.3 多媒體內容描述檔.....	12
2.3.1 EBS、EBP.....	12
2.3.2 HTML.....	12
2.4 多媒體相關工具介紹.....	12
2.4.1 CxImage.....	12
2.4.2 Devil.....	13
2.4.3 Lame.....	14
2.4.4 XMG Player.....	14
三、系統功能需求分析.....	16
3.1 內容改編協調伺服器功能需求.....	16
3.1.1 未改編協調內容伺服器.....	16
3.1.2 已改編協調內容伺服器.....	17
3.1.3 主要協調系統.....	17
3.2 協調管理者介面功能需求.....	18
3.3 行動裝置播放器功能需求.....	19
3.4 多媒體教材內容的完整性需求.....	19
3.5 多媒體內容協調後結構完整性需求.....	20
四、系統架構.....	23
4.1 系統架構圖.....	23

4.2	轉換協調系統架構說明	24
4.2.1	下載教材運作流程	24
4.2.2	主要協調部份之架構與流程	24
4.2.3	上傳協調完教材運作流程	26
4.3	線上內容資料庫系統	27
4.4	手持行動裝置呈現應用程式	27
五、	系統設計與實作	30
5.1	描述檔解譯器的實作	30
5.2	轉換協調系統的實作	32
5.2.1	單純的比例協調	32
5.2.2	圖片的協調	34
5.2.3	文字的協調	37
5.2.4	影像的協調	43
5.3	應用端播放器的實作	47
5.4	結構完整的協調	48
5.5	應用於其他種類之多媒體內容	56
六、	實作範例	57
6.1	管理者 web 操作介面	57
6.2	PDA 應用端播放器呈現畫面	60
6.3	協調前後的畫面比較	63
6.4	結構協調的範例	66
七、	結論	68
7.1	總結	68
7.2	未來展望	71
	參考文獻或資料	72

一、緒論

1.1 手持行動裝置應用之發展

自從第一代手持行動電話問世，主要的功能不外乎用來聯絡彼此，所能提供的功能相當有限，電話簿功能、文字簡短訊息的發送、鬧鈴提醒、簡單的萬年曆，還有幾種簡單的小型遊戲（例如貪食蛇、打磚塊等等）。隨著使用者的需求性，加上科技日新月異的輔助，以往使用者只能在大又笨重的筆記型電腦，甚至於一般桌上型個人電腦才擁有的功能，都已附加在手持行動裝置上，例如可以播放多媒體檔案 mp3、wmv...等等，功能較好的還可和隨插即用的記憶卡結合播放短片，透過小像素的鏡頭，可以使用照相等等這些功能，多元化的發展以符合使用者的需求。

手持行動裝置中功能較強大的理應算是 PDA(Personal Digital Assistant) 了，Microsoft 在 PDA 上開發了許多方便的應用程式，在一般桌上型個人電腦常使用的基本功能，PDA 都有提供服務，連編輯軟體 Office 都有做簡單的移植，很符合使用者的要求，但因為體型較大攜帶不方便，而且在電池的持久性較差，更重要的是在價錢方面較昂貴，一般大眾使用普及率並不高。但是隨著科技的發展將體積縮小，變的更輕巧，電池的持久性改良，因為技術的發展，價錢也相對地降低不少，更重要的是跟無線網路(Wireless network)和 GPRS(General Packet Radio Service)系統作結合，發展出了 PDA 手機。

表 1-1 全球及美國 PDA 數量預估（單位：千台）

PDA 銷售量	2000	2001	2003	2005	2007
全球 PDA 銷售量	12,175	16,375	28,490	43,525	61,370
全球 PDA 手機銷售量	230	675	3,845	10,062	18,710
美國 PDA 銷售量	5,540	6,640	10,770	15,680	21,110
美國 PDA 手機銷售量	N/A.	46	970	2,901	5,490
使用中的 PDA 數量	2000	2001	2003	2005	2007
全球 PDA	24,920	40,435	85,620	149,030	227,400
全球 PDA 手機	230	900	6,350	21,400	48,800
美國 PDA	12,345	18,510	35,165	56,550	80,645
美國 PDA 手機	N/A	46	1,350	5,950	14,250

如表 1-1[1][2]所示，因為 PDA 手機的實用性，許多手持行動裝置生產的大廠商，相繼推出了更方便、輕巧、省電、娛樂性更高的新機種，消費者的購買

量也逐年增多。對於軟體業者來說，這是一項龐大的商機，無論是針對新機種重新設計一系列的應用程式，與 PDA 廠商提供的硬體功能結合，增加 PDA 手機的實用性，或者對於現有的熱門應用軟體，移植到 PDA 手機上使用，都可以為該公司帶來不少的實質收入，所以越來越多軟體業者爭相進入這個領域，分食這塊大餅。

1.2 研究動機與目標

針對同一份多媒體資料，使它在不同的裝置上呈現，通常會將內容呈現方式重新設計，因為大部分的多媒體內容由一般桌上型電腦設計，內容的設計也是針對一般桌上型電腦，但是重新設計此一步驟必須花費廠商相當多的人力和時間，成為廠商的一大負擔。既然是同一份多媒體資料，如果能快速的轉換到不同的裝置上，將會節省很多人力跟時間。

藉由情境感知(Context Awareness)[4][5]機制，找出對於目標裝置呈現有影響的因素，如：通訊方式、行動裝置軟硬體的提供、使用者偏好...等等各種因素，提供給內容改編協調(Content Adaptation)[11][12]決策，來解決在不同的裝置環境下呈現同一份多媒體資訊的問題，減少多媒體內容設計者對於相同的一份資料，付出額外的時間和人力，而且能正確的轉換來源內容。

本研究延伸洪啟彰在《樣板式多媒體內容在手持行動裝置之情境及內容協調與呈現—以樣板式的多媒體英文試題為例子》[3]中的研究成果，利用靜態(static)及動態(dynamic)混和的多媒體格式轉換，在時間與空間的使用上，取得一個較佳的結果。但此研究對於其他多媒體資料：如多媒體教材，並無法做有效的轉換。而多媒體教材的設計，製作者常會將教材的教學內容，設計一定的流程，使閱讀者依序學習、漸入佳境，進而瞭解到教材製作者想要傳達給閱讀者的意涵，如果在轉換協調過程中，改變了教學的流程，此教材的原意極有可能會被改變，這也提高了多媒體教材在協調的難度。

將多媒體教材從一般桌上型電腦協調轉換至 PDA 上，讓使用者不因為 PDA 的畫面和硬體等等各種限制，造成閱讀的困難，在協調過程中能兼顧畫面的舒適性和協調完教材原意的保留，這是本研究最主要的目標。

1.3 研究方法與步驟

本研究是將舊有的研究成果加以延伸，加入新的功能且改進不足的部分，所以除了一般理論的探討外，還需先了解已有的運作原理及研究結果，評估所需的研究方向及實做，大略分為下列幾個步驟：

- (a). 相關文獻的收集與研究：
針對情境感知(Context Awareness)機制和內容改編協調(Content Adaptation)技術的相關文獻探討，PDA 上程式寫作書籍閱讀，了解多媒體檔案格式。
- (b). 了解舊有的系統與設計概念：
包括多媒體教材編輯軟體及其描述檔、英文試題平台、英文試題轉換協調平台、PDA 播放器程式。
- (c). 分析系統的需求：
了解現有系統設計概念和結果後，對於不同的多媒體檔案來源，找出要達到研究目標所需要新增、改進的部分。
- (d). 與已有的系統整合規劃
將原始多媒體教材資料庫、改編協調伺服器(Adaptation Server)，協調完成多媒體教材資料庫、PDA 下載及播放系統整合，規劃完整的協調和下載呈現流程。
- (e). 系統實作：
依照各系統間的規劃流程，選定所需的開發工具，依序實作各系統。
- (f). 實作結果分析：
評估協調完成後產生提供 PDA 呈現的多媒體教材，使用者閱讀狀況是否良好，教材的教學原意是否因為轉換協調而改變。

1.4 研究範圍

本研究相關名詞及參考範圍，描述如下：

- (a). 個人電腦上多媒體編輯工具—智勝國際科技編輯手[6]
使用文字、聲音、圖片、影像等多媒體素材的編輯工具。內容製作者不需要撰寫任何程式就可編輯出生動且劇情(Scenario)豐富的多媒體內容。
- (b). 情境感知改編協調(Context Aware Adaptation)
研究現有的情境感知協調系統[3]，針對非樣版化的多媒體內容，尋找相關文獻，進一步探討有效的轉換協調方式。
- (c). 手持行動裝置
PDA：Pocket PC 2003 為播放器主要開發平台[7][8]。
- (d). 多媒體檔案格式與轉換
多媒體檔案格式制訂探討，利用已有的多媒體工具[9][10]，進行多媒體檔案格式轉換、尺寸處理。

1.5 章節概要

本章先指出手持行動裝置因為科技的進展而更能符合使用者的需求，普及率增加後，在手持行動裝置上開發應用軟體價值相對提高，接著說明軟體業者在個人電腦和手持行動裝置之間轉換的高成本與高難度，最後再提出如果能快速且精確的將多媒體內容轉換至異質的裝置，在呈現上能讓使用者舒適的研究動機及目標，並且簡略地介紹本研究的研究方法及範圍。

第二章，詳細介紹和本研究相關的文獻內容，包括情境感知改編協調 (Context Aware Adaptation)，及多媒體內容格式的探討。

第三章，針對現有的情境感知協調系統(Content Adaptation Server)對於不同的來源（如多媒體教材）的情形下，系統的架構規劃及需求分析。

第四章，說明本研究主要的系統架構及其各部分的子架構。

第五章，詳細介紹系統中重要功能、設計流程、演算法的應用。

第六章，執行系統操作流程，利用實做畫面、圖表等等，展示實例。

第七章為總結，檢視研究成果及相對於前研究的改進部分，最後是未來展望。



二、 相關研究

利用情境感知(Context Awareness)機制，進行多媒體內容的改編協調[3][4][5][11][12][14]，在學術界已有許多相關受到討論的議題以及研究成果，透過對不同的環境、硬體設備，如果能藉由此機制自動感測相關所需的資訊，對於內容適當地改編，期望能更適合異質的環境。

此外本研究對於多媒體格式，必須根據決策系統產生的結果做相關的處理，所以相關的多媒體格式的制訂及多媒體處理工具也將作粗略的介紹。

2.1 情境感知改編協調

2.1.1 情境感知(Context Aware)

情境運算概念(Context Aware Computing)最早是由 Schilit 和 Theimer 在《Disseminating active map information to mobile hosts》[13]中所提出，由於當時手持行動裝置的研究相當熱門，在高動態的執行環境之下，與鄰近的裝置、服務系統發生作用及使用者周遭的環境，各項的資訊對於功能的影響是顯而易見的，進而討論本地資訊(location information)，或稱情境資訊(Context information)。

根據研究學者針對不同的研究主題，對於情境(Context)的定義都不盡相同，Schilit 和 Theimer (1994) 認為地點、週遭的人及物件之識別與改變情形為其定義。Brown (1997) 等人則以位置、使用者週遭的人及物件、時間、季節和溫度等為主。Ryan 等人 (1997) 提出使用者位置、環境、識別及時間。Dey and Abowd (1999) 參考上述各家定義，提出較為接受且較為正式的定義，在研究[15]中提出「Context是與使用者、應用程式之間互動有用的資訊，任何足以描述人、事、時、地、物之實體皆屬之」。

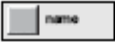





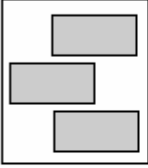
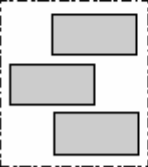

而在研究[3]中，單就多媒體內容改編協調功能中，提出下列幾類情境資訊：

- (a). Identity
- (b). Spatial information
- (c). Temporal information
- (d). Environment information
- (e). Social situation
- (f). Activity
- (g). Resources
- (h). Schedules and agendas

因為這些情境資訊的影響，無論相同時間或不同時間，不同的使用者如果在不同

地點，他們同時使用同一份資料所收到的品質都會有所不同。以下以研究[5]中所舉的例子來說明：

表 2-1 各元件代表之意義

	<p>An atomic item; the icon describes the media type:  video,  audio,  text and  image.</p>
	<p>A button for user control of media timing.</p>
	<p>Composite media items are represented by thick rectangles enclosing the component items.</p>
	<p>Virtual components are represented by dashed rectangles enclosing the context dependent items.</p>
	<p>Synchronization relationships label the arrows connecting the involved media items.</p>

系統業者如果提供氣象預報之訊息服務，其中可能包括文字、圖片、簡短的视频來說明目前的天氣概況和預測的情形，但面對不同的環境來說，並不一切都適用，藉此來說明情境的影響。以下圖 2- 1 有五種情境：

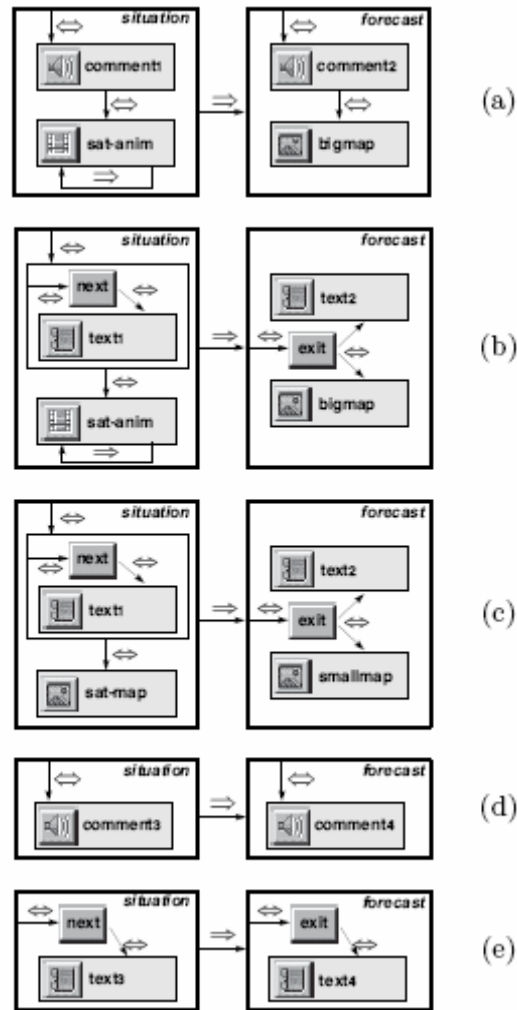


圖 2-1 不同情境所對應的設計方式

(a). 一般桌上型電腦在家中使用

在家中使用裝置運算速度和網路頻寬都很充裕的情形下，可使用文字、聲音、大型圖片，甚至於使用短片，盡可能地把目前的天氣型態清楚地告知使用者。

(b). 一般桌上型電腦在無聲音狀態

雖然裝置的運算速度快，但是在無聲的狀況下，如：在圖書館中或者音效輸出裝置有問題，會利用大型圖片和詳細的說明文字來告知使用者。

(c). PDA 和小頻寬的無線網路

因為 PDA 的運算速度和小頻寬的無線網路限制下，對於檔案較大的影片、聲音，甚至於大型圖片傳輸都相當耗時，所以如果能用經過處理後的小圖示搭配文字說明來呈現，會是較佳的選擇。

(d). (e). 也是類似的狀況。

由這個例子可知道，情境資訊(Context information)對於多媒體內容的設

計及呈現方式有相當大的影響。因為使用者可能經常在不同的環境之間移動，如果能掌握情境資訊並且加以利用，稱為情境感知(Context Aware)，如此一來便能讓使用者在不同的環境下，還能讓多媒體內容的品質保持一定的水準。一般的協調系統要經過決策引擎(decision engine)[11]，過程如圖 2- 2 所示：

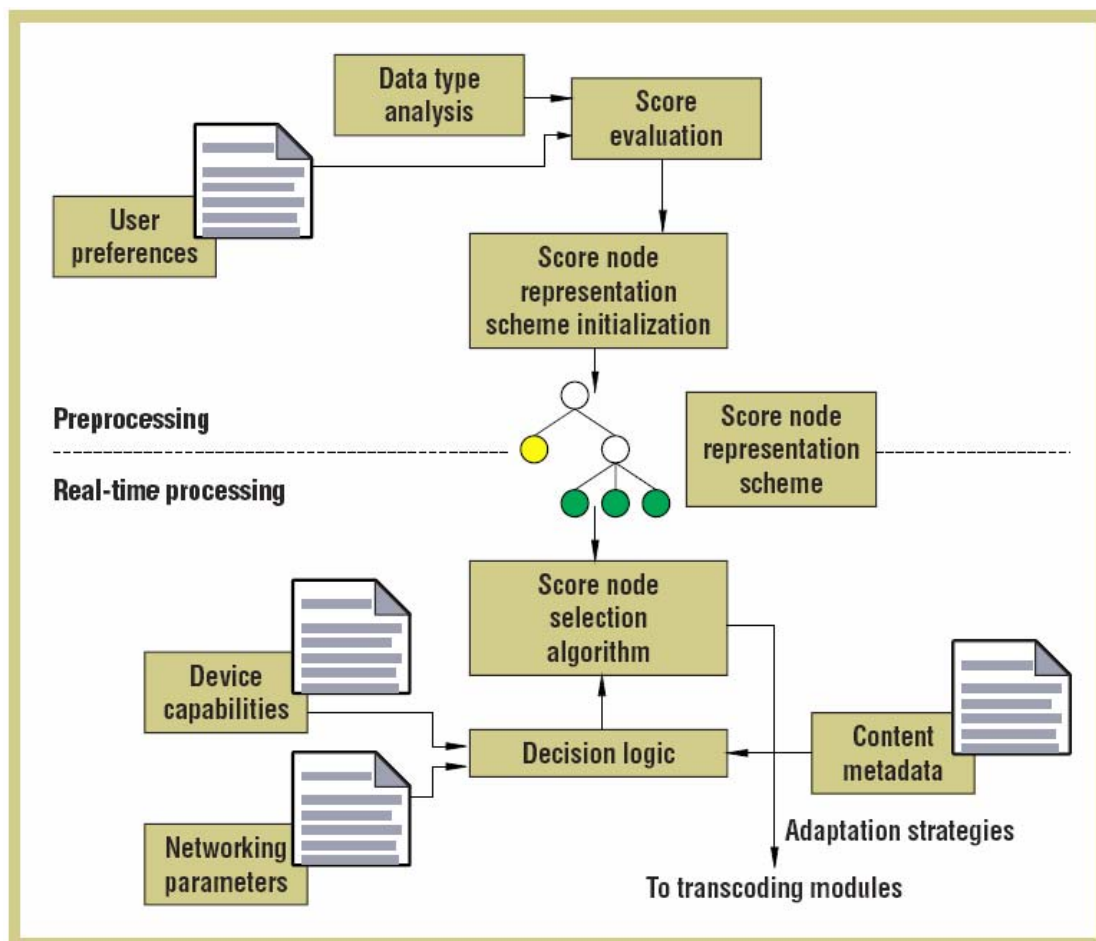


圖 2- 2 決策引擎(decision engine)

決策引擎分為兩大部分：

1. 預先處理部分(Preprocessing)

這是在使用者發出要求已經運作的流程，主要是產生一量化的資料集合，透過對於每項資訊定義的數值總計，分數評估(score evaluation)和表徵(representation)則為其統計結果，將送到主要決策中心作為決策時的參考資料。

2. 即時處理流程(Real-time processing)

決策引擎的主要演算中心：分數節點選定(score node selection)、決策邏輯(decision logic)、節點優先順序(node order)，定義各節點的分數，決策邏輯(decision logic)根據各種特性選定一個較適合的分數節點(score node)，而再根據使用者可能的偏好向下選擇一個最終節點(leaf node)最為最後輸出的結果。

2.1.2 內容改編協調(Content Adaptation)

多媒體內容的改編協調原因，大部分都是因為目前使用的載具(Device)和當初設計多媒體內容時的狀態不相同，所以必須適當地將多媒體內容作降級的動作，有些會以 QoS(Quality of Service)[16]為依據，另外還有會以能支援的程度範圍當作考慮，一般而言都會搭配情境感知(Context Awareness)機制來做較有效的轉換協調。

而轉換的方式可略分為下列兩種型態[3][5][17]：

- 靜態的轉換型態(Static adaptation)
在多媒體內容設計完成後，便將之轉成各種版本，存放在協調完內容伺服器中，等到要使用時，再根據所需的樣式下載使用。
- 動態的轉換型態(Dynamic adaptation)
不同於靜態的設計，多媒體內容製作完成後，並不做任何轉換的動作，只有等到使用者提出要求後，才動態產生所需要且適合的多媒體內容。

這兩種轉換型態的差別如表 2- 2：

表 2-2 靜態轉換與動態轉換的差異性

	優點	缺點
靜態轉換型態	<ul style="list-style-type: none"> ● 使用者要求前就已經完成協調的動作，所以可立即發佈給目標裝置。 ● 由於是事前就已協調完成，所以在呈現的品質可達固定水準。 	<ul style="list-style-type: none"> ● 由於必須準備許多版本，所以在檔案伺服器需要許多空間存放。 ● 如現有的版本都不符合目標裝置提供的條件，則無法提供呈現。
動態轉換型態	<ul style="list-style-type: none"> ● 檔案伺服器只需存放一份資料，不需額外的空間。 ● 動態根據需求做協調，所以較能滿足多種裝置的要求。 	<ul style="list-style-type: none"> ● 使用者要求時才針對目標裝置協調，所以作用時間較長，花費額外的時間。 ● 動態產生的協調畫面，呈現的品質無法固定。

總結來說，靜態主要是花費空間、減少時間，而動態的情形則是相反，本研究則是延續先前之研究成果[3]，綜合動態與靜態的優缺點，使用動態與靜態混和方式，使空間與時間的花費中取得平衡點。

2.2 樣版式多媒體內容協調

樣版式多媒體內容協調[3]，針對樣版式多媒體英文試題協調至 PDA 手持行動裝置。樣版式的多媒體英文試題設計[18]主要是以三個元素：Stimulus、Question、Answer 製作而成，協調方式是以多種設計完成的樣版，經過螢幕大小、文字斷行數量、圖片縮小失真程度等等，逐一淘汰不合適的樣版內容，並推薦一最適合的呈現方式，簡單的選擇方式如圖 2- 3[3]。

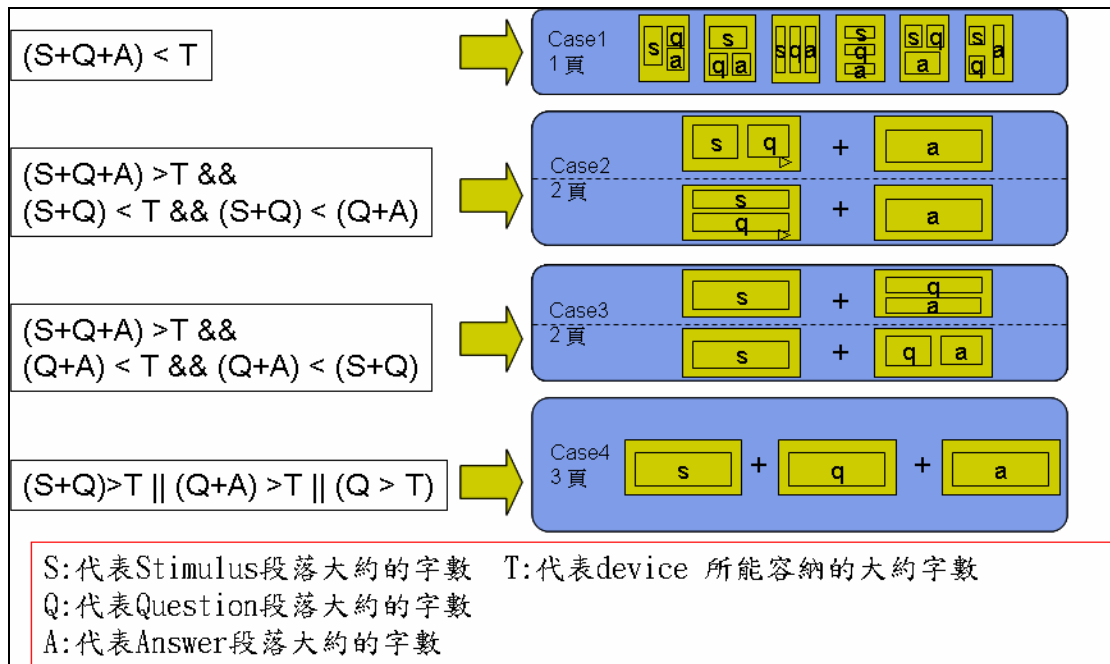


圖 2-3 樣版選擇演算法

等到選擇某一類型(Case)集合後，檢查文字斷行效果，當斷行數過多，會造成閱讀者的困難，淘汰斷行數過多的樣版，演算法如圖 2- 4[3]：

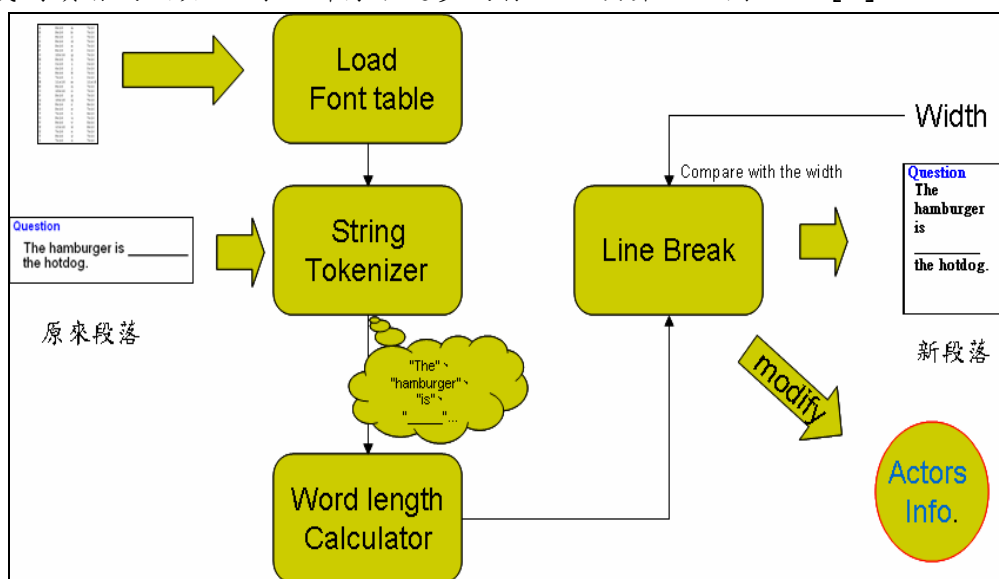


圖 2-4 文字斷行演算法

圖片在協調過程中，勢必經過縮小的步驟，設計的樣版長度 and 寬度比例不一定適合每一張圖片，所以檢視圖片失真的程度，選出程度較低的樣版，演算法如圖 2- 5、圖 2- 6[3]。

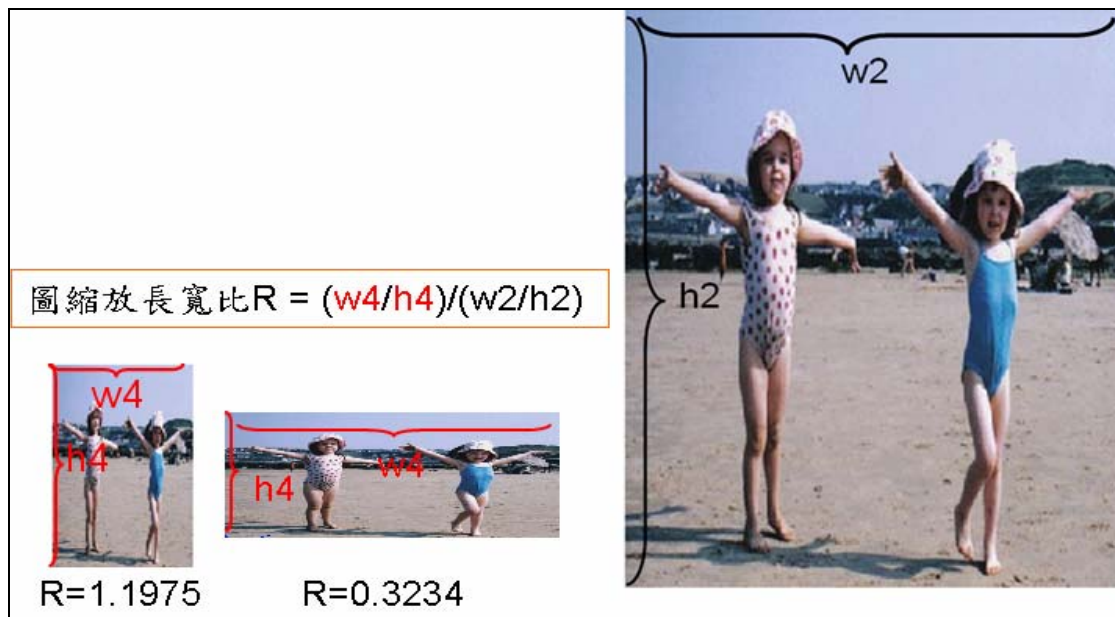


圖 2- 5 圖片縮放比例

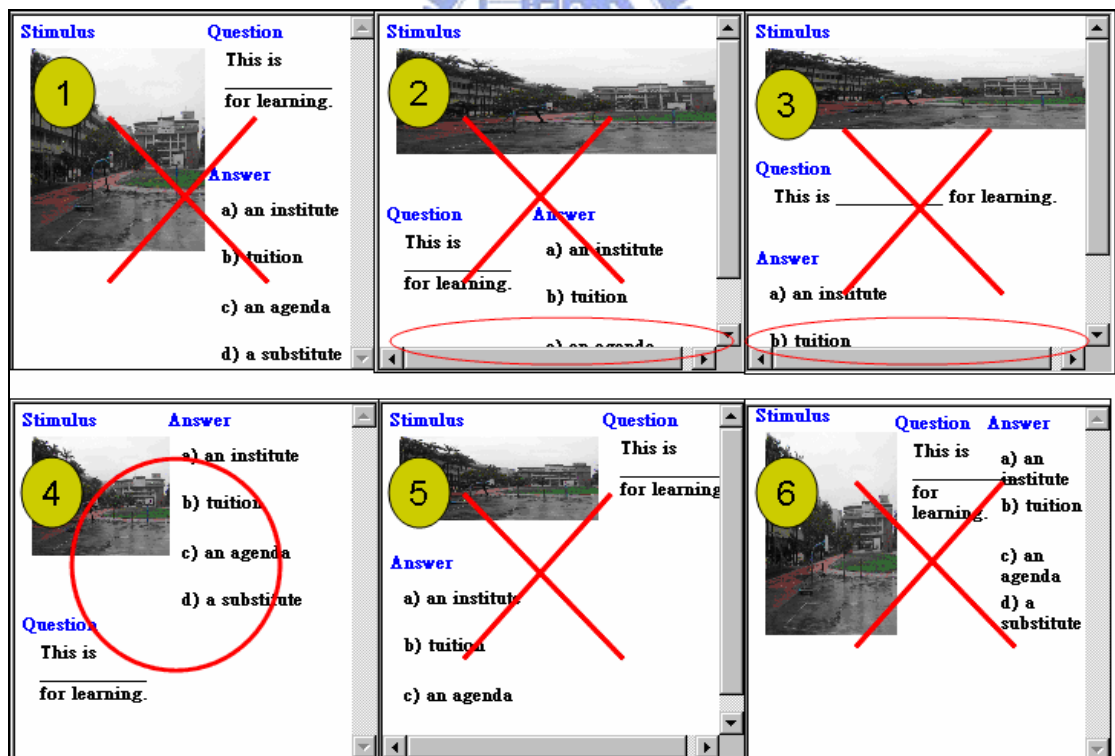


圖 2- 6 選擇合適的呈現

2.3 多媒體內容描述檔

2.3.1 EBS、EBP

EBS 和 EBP 描述檔格式為本實驗室技術轉移給廠商開發多媒體編輯軟體—編輯手[6]中用來描述演員資訊，動畫效果、呈現方式等等各種資訊，將描述檔與多媒體檔案分離，播放器只需先讀入描述檔中的演員資訊和劇情，等到需要演員出場時在到目標位置將檔案呈現給觀看者。編輯手也以此描述檔為基準，轉換至不同的格式，達到不同的呈現方式，如：html、flash、MP4 等等。

2.3.2 HTML

HTML(Hypertext Markup Language)為 Tim Berners-Lee 在 1990 時設計的語言，和 Macintosh 的 Claris XTND 系統極為相似，不同的是可以在不同的平台運作。但當初設計的標籤(TAG)太過於簡略，所以所能提供的功能有限，不過 HTML 的設計概念很簡單，能快速上手，藉由網路的快速發展，綜合使用者的意見及經驗增加了、<embed>、...等等好用的標籤。但是也由於這個原因，使 HTML 的發展情況變的相當複雜，各式各樣的版本充斥市面，微軟和 Netscape 制訂兩種不相干的版本，造成使用者的困擾。

有鑑於這樣的情形，微軟、Netscape、W3C 著手整合的工作，將大部分的規格制訂一致，雖然還有某些部分，如：<layer>和 CSS(Cascading Style Sheets)的定位還不明，在網路上兩方仍爭執不休，但相對於前面幾個版本，HTML4 相對於穩定而踏實許多。

2.4 多媒體相關工具介紹

本研究用來處理多媒體檔案的相關工具，會在此節做簡略的介紹其原理架構及應用方向。

2.4.1 CxImage

CxImage[9][19]是由一位義大利人 Davide Pizzolato 開發的多媒體圖形處理工具，作者以開放原始碼(open source)的方式放在網路上提供使用者使用，也由於這樣的方式，功能也在使用者的回報和建議下日趨強大，可對 BMP、GIF、ICO、TGA、PCX、WBMP、WMF、JPEG、PNG、MIN、TIFF、JBIG、PNM、PPM、PGM、RAS、JPEG-2000 等等多媒體格式作相關的處理。CxImage 簡略設計架構如圖 2- 7：

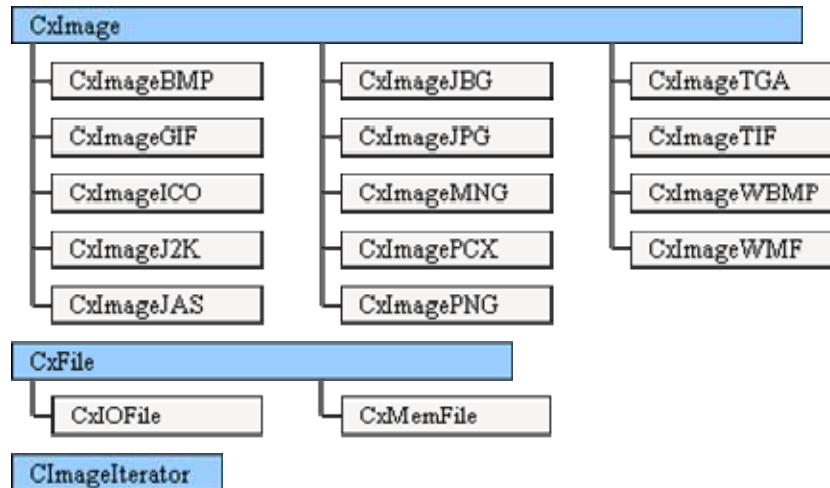


圖 2-7 CxImage 設計架構

CxImage 的功能相當強大，而本研究只使用下列幾種功能：

- (a). 載入圖檔
- (b). 旋轉圖檔
- (c). 讀取圖檔特定座標像素(pixel)數值
- (d). 儲存圖檔

2.4.2 DevIL

DevIL[10](Developer's Image Library)也是以開放原始碼(open source)的形式提供給程式開發者使用於處理影像方面的工具，所能提供載入的檔案格式有 bmp、cut、dds、doom、gif、ico、jpg、lbm、mdl、mng、pal、pbm、pcd、pcx、pgm、pic、png、ppm、psd、psp、raw、sgi、tga、tif、hdr，可儲存的格式有 bmp、dds、h、jpg、pal、pbm、pcx、hdr、pgm、png、ppm、raw、sgi、tga、tif。

針對 OPENGL、Windows GDI、DirectX and Allegro，DevIL 提供對應的 API，讓使用者方便使用，也可以利用 Djgpp、MSVC++、Linux gcc、Delphi、Visual Basic、Power Basic and Dev-C++ 等等編譯器，重新編譯 DevIL 已提供使用者更有效開發程式的工具。

本研究使用此工具主要的應用功能：

- (a). 載入圖檔
- (b). 改變圖檔尺寸
- (c). 改變圖檔格式
- (d). 儲存圖檔

2.4.3 Lame

MP3(MPEG : Moving Picture Experts Group Audio Layer-3) , 1993 年由德國夫朗和費研究所和法國湯姆生公司合作發展, 剛開始編碼是採用固定編碼率(CBR), 一般常見的是以 128KBps, 如果需要更好的音質可提高至 320 KBps, 但空間的使用相對增加。為了解決這個問題, 開始有研究改用可變編碼率(VBR)的壓縮方式, 在需要高品質時採高比特率(Bitrate), 不需要時則反之, 早期是由 Xing 公司所提出的壓縮方式, 但是其壓縮的效果很差, 和 CBR 壓縮品質相去甚遠。

因為 Xing 編碼器品質不符合需求, 各方紛紛提出自己的 VBR 壓縮演算法, 其中公認最好的為 Lame 提出的, 以 VBR 為基礎, 改進為 ABR(Average Bitrate) 平均比特率, Lame 針對 CBR 的空間使用問題和 VBR 品質不符合需求而提出獨特的編碼演算法, 以每 50 幀(30 幀約為 1 秒)為一段, 對於人類不敏感和沒有影響的頻率使用相對於低的流量, 高頻和大動態的表現則使用高容量, 可以說是 VBR 和 CBR 的折衷選擇。

2.4.4 XMG Player

XMG Player[3]原本是設計用在手持行動裝置上呈現研究結果的播放工具, 播放器會先將多媒體內容的描述檔讀入並且解譯, 描述檔是以 <Story>、</Story>、<ActorInfo>、</ ActorInfo > 等等標籤制訂而成, 在描述檔的開頭會先將每個多媒體角色的檔案名稱、字型大小、顏色、初始位置、移動路徑等等告知播放器, 等到輪到該角色演出時, 播放器會根據所提供的資訊, 將角色呈現出, 畫面的呈現方式是使用微軟提供的繪圖 API 將畫面當作畫布, 當需要更改時, 便將畫面重新更新, 圖 2- 8 為簡易的描述檔範例。

```
1 <Story>
2 <ActorInfo>
3   <actor type="IMAGE" name="Undefined_0_Actor" file="03c620.jpg" frames="1" path="0,0"/>
4   <actor type="IMAGE" name="Undefined_1_Actor" file="03c775.jpg" frames="1" path="0,0"/>
5   <actor type="IMAGE" name="Undefined_2_Actor" file="03c343.jpg" frames="1" path="0,0"/>
6 </ActorInfo>
7
8 <Scenerio>
9   <event name="Undefined_0_Actor" vscrollbar="false" hscrollbar="false">
10    <animate name="Undefined_0_Actor" action="SHOW"/>
11    <animate name="Undefined_1_Actor" action="SHOW"/>
12    <animate name="Undefined_2_Actor" action="SHOW"/>
13  </event>
14
15  <event name="Undefined_1_Actor" vscrollbar="false" hscrollbar="false">
16    <animate name="Undefined_0_Actor" action="SHOW"/>
17    <animate name="Undefined_0_Actor" action="SHOW"/>
18  </event>
19 </Scenerio>
20 </Story>
```

圖 2- 8 XMG 描述檔範例

本研究的目標是達成多媒體內容的協調與格式的轉換，首先對於已有的內容協調理論做詳細地學習，所以從文獻中提出的論點和範例清楚瞭解對於裝置運作時有影響的情境資訊(Context information)，與在不同的情況下，協調的決策中心所選擇的重點依據。在設計系統架構時，循著前系統在靜態與動態混和協調的設計理念下，找出對於不同來源的有效協調方法，也因為來源的不同，所以需要更多多媒體工具輔助本研究做相關的處理，詳細了解這些多媒體工具的設計理念與架構後，更可以在最適當的地方做最有效發揮。



三、系統功能需求分析

本系統的目標是將多媒體教材從 PC 版本協調轉換至 PDA 所能呈現的畫面，需提供資料庫系統讓教材編輯者上傳已編輯完成之教材，透過轉換協調系統的運作，將未改編協調教材轉換至已改編協調教材，並放置於另一資料庫系統，在協調過程中，協調管理者透過管理者介面監控協調品質並適時提供必要資訊，而 PDA 使用者透過無線網路的連線，下載所需的教材並播放。

從上面的運作流程可知，為了讓使用者快速協調未經改編協調的多媒體內容，而且不需要多餘繁瑣的操作就能藉著行動裝置直接協調後的多媒體內容，因此本系統大略分為四個主要功能架構：

- (a). 內容改編協調伺服器(Content Adaptation Server)功能需求
- (b). 協調管理者介面功能需求
- (c). 行動裝置播放器功能需求
- (d). 多媒體教材內容的完整性需求

以下會針對各部分作詳細解說：

3.1 內容改編協調伺服器功能需求

其中分為三部分：未改編協調內容伺服器(Pre-Adaptation Content Server)、已改編協調內容伺服器(Post-Adaptation Content Server)、主要協調系統(Main Adaptation System)。主要運作流程如圖 3- 1。

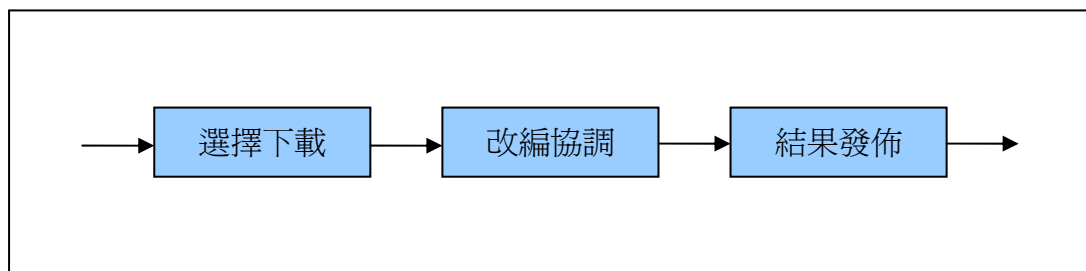


圖 3- 1 內容改編協調流程

3.1.1 未改編協調內容伺服器

主要存放多媒體編輯者已製作完成多媒體內容的內容伺服器，需求如下：

- (a). 多媒體內容上傳功能
需提供多媒體內容製作者將已編輯完成的多媒體內容上傳的功能。
- (b). 瀏覽、管理功能
內容管理者可透過 web 介面瀏覽，維護及管理存放之多媒體內容。

(c). 多媒體內容下載功能

透過帳號與密碼的控管，提供選擇目標內容的下載功能。

表 3-1 未改編協調內容伺服器環境

未改編協內容伺服器	
作業系統	FreeBSD5.0/apache2.0
資料庫	MySQL
程式開發語言	PHP, JAVA, HTML

3.1.2 已改編協調內容伺服器

存放已協調完成之多媒體內容，所需提供的需求如下：

(a). 協調完多媒體內容上傳功能

將協調完成的多媒體內容，透過所提供的介面或功能上傳至已協調完成多媒體內容伺服器。

(b). 瀏覽、管理功能

與未協調完成內容伺服器相同，提供內容管理者可透過 web 介面瀏覽，維護及管理存放之多媒體內容。

(c). 手持行動裝置下載功能

當手持行動裝置提出需求，經過認證功能，傳送所需要的多媒體內容給使用者。

表 3-2 已改編協調內容伺服器環境

已改編協內容伺服器	
作業系統	FreeBSD5.0/apache2.0
資料庫	MySQL
程式開發語言	PHP, JAVA

3.1.3 主要協調系統

為系統的核心部分，主要功能為將來源多媒體內容，協調轉換至符合其他手持裝置的另一版本，其需求如下：

(a). 下載功能

由於內容協調系統 (Adaptation System) 和多媒體內容伺服器 (Content Server) 屬於不同的系統，彼此溝通需經過傳輸，本研究選擇使用 http 協定，利用 php 和 MySQL 所提供的功能，下載使用者所選定的檔案，

(b). 解譯描述檔

當收到所選擇的多媒體內容，經過解壓縮完後，系統會尋找檔案中加密過的 EBP 描述檔案，使用技術合作廠商[6]提供的解密工具將檔案解密後，解譯描述檔中的演員(文字、圖片、聲音、影像)設定和劇情設定(換場、互動資訊)，並改存放自訂資料結構以便使用。

(c). 輸入目標裝置環境、使用習慣等等變異係數

系統提供一般化的預先設定，但為了因應不同的使用者，必須允許使用者根據自己的喜好變動，在產生最後的結果前，使用者可使用協調管理者介面提供的工具，選擇不同的呈現結果，內容協調系統會將新的設定套用到最後產生的版本。

(d). 多媒體檔案格式轉換

協調過程中，對於圖形、影像必須做額外的處理，由於本實驗是使用動態轉換(Dynamic adaptation)和靜態轉換(Static adaptation)混和方式，所以圖形和影像是在此便處理完成，根據協調系統決策結果，將圖形和影像根據所需要的大小做縮放。

(e). 描述檔案改寫

由於轉換過程中，多媒體物件(文字、圖形、影像、聲音…等等)的檔案名稱、顯示位置、甚至於演出流程都被改變，所以必須更改描述檔，配合新的演出劇情和對應到正確的檔案名稱，而輸出的格式也分為輸出成網頁(html、php)和提供給行動裝置撥放器(XMG Player)讀取的 XMG 描述檔。

(f). 壓縮發佈功能

將已協調完成的多媒體內容壓縮完成發佈至伺服器上，並同步新增資料庫內容清單，以防止資料庫欄位與實際內容不符合。

3.2 協調管理者介面功能需求

協調管理者透過此介面掌控協調過程的變異係數和協調品質的維持，其所需的需求如下：

(a). 瀏覽資料庫、指定下載功能

透過網頁瀏覽的方式連線至為改編內容伺服器的資料庫，選擇所需要協調的多媒體內容，並且交給下載模組下載至協調系統。

(b). 輸入變異係數

基本上協調過程中為自動根據所需資訊做協調的工作，而協調管理者可以在預設條件協調完後，預覽協調推薦的樣式，再進入產生最後結果前，管理者可改變預設條件，例如：本身的使用習慣、字型大小、物件的擺放方式，做更彈性的轉換方式。

3.3 行動裝置播放器功能需求

行動裝置上播放器主要是用來驗證和檢視實驗結果的工具，不過為了讓整個協調轉換的流程更順暢，對於播放器還是有幾個功能需求，其列出如下：

(a). 下載功能

使用者登入已協調完內容伺服器後，畫面會列出所有已存在的多媒體內容，使用者可自由選擇需要的檔案，如果所選擇的多媒體內容已刪除或者還沒進行協調轉換而需要動態轉換時，則和協調轉換系統做溝通，等到轉換完成後再進行下載，由於協調轉換系統上傳至內容伺服器前，會將檔案打包成一個壓縮檔案，以防傳輸過程中有遺漏的缺失，所以當行動裝置播放器收到檔案後，將先進行解壓縮的工作，使用者不需做額外點選動作。下載過程中為了讓使用者瞭解下載進度，而且行動裝置是利用無線網路連接，故設計一下載狀況進度條，讓使用者掌握資訊，並確定網路連線沒有發生問題。

(b). 播放功能

在下載檔案且解壓縮完後，自動提示播放檔案，播放功能流程為先讀入多媒體內容的描述檔案，然後在依照劇情的設定一一演出，而且演員不同型態的多媒體檔案格式，所以需要有以下功能：

- 1) 讀取多媒體內容描述檔案並且解譯
- 2) 顯示文字功能
- 3) 顯示圖檔功能
- 4) 播放聲音檔案
- 5) 播放影像檔案
- 6) 演員根據路徑移動功能
- 7) 根據劇情可和使用者互動功能



3.4 多媒體教材內容的完整性需求

教材製作者在設計教材時，會將一般大眾化的教學習慣，再加上個人的教學風格，設計一獨特的教學流程，利用此教學流程和教學工具配合，使學習者能快速、準確地瞭解教材製作者想要傳達的意義，就好像電影的劇情和演員的配合演出一樣，在此可將教學流程(Scenario)當成電影的劇情，多媒體教學工具(文字、圖片、聲音、影像...等等)當成演員。

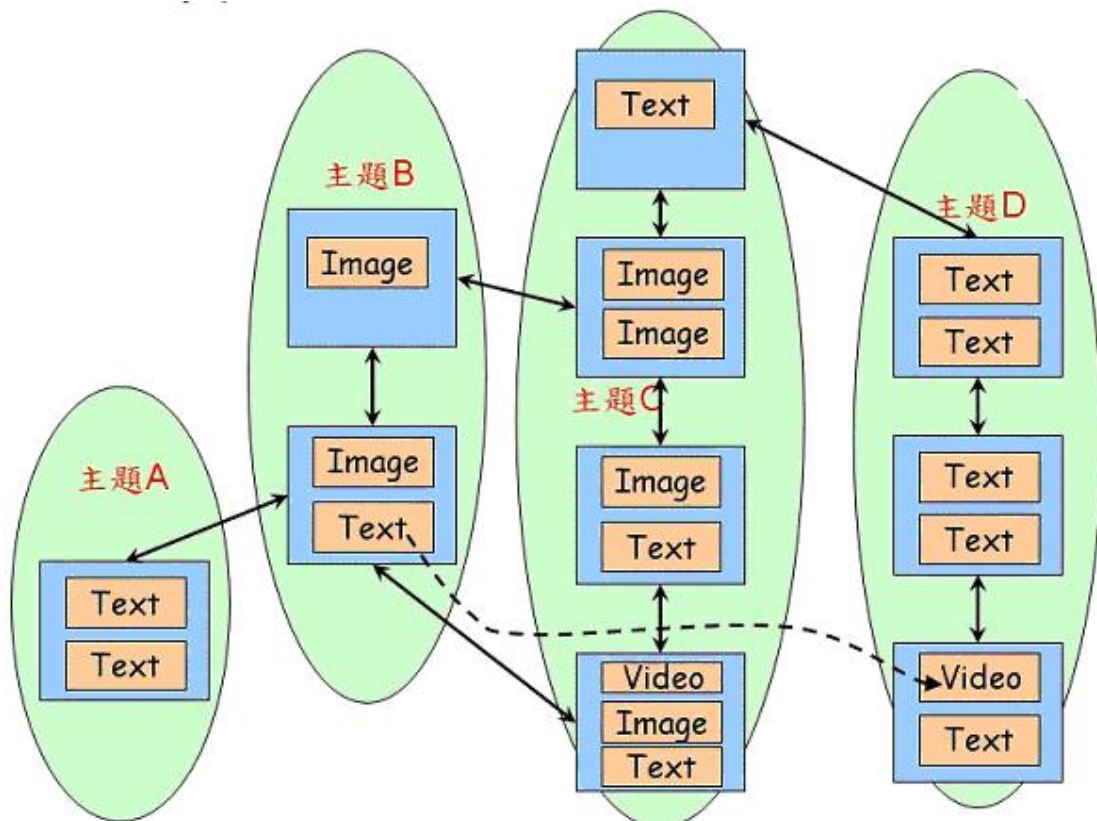


圖 3-2 教材教學流程圖

如圖 3- 2 所示，教材設計時會有幾個小單元，在小單元中每個頁面的關連性相當高，依序閱讀便可清楚瞭解，此外有可能在設計課程時，小單元之間也可能有所相關。以英語的學習為例子，課程設計可能有發音、字彙、來源典故、分組練習...等等小單元，學習和字彙時發音時，有可能會跳躍到來源典故單元去說明這個英文單字為什麼會這樣發音，或者這個字是因為什麼原因才被創造出來，是因為當初誤用還是其他的原因，在學習某幾個字發音之後，可能跳躍到分組練習小單元，讓課程學習者能相互的練習，加強記憶效果。

由英文學習的例子我們可以知道，教材製作者製作的教學流程有一定的用意，如果在改編協調過程中，任意改變教學流程，就好像觀看電影時，演員將演出的順序改變，造成觀看者對於劇情的混淆或錯誤的認知，教材學習者可能在學習完整份教材後，對於剛剛的教學前後摸不著頭緒，沒有辦法領略教材製作者想要傳達的意義，故在改編協調過程中，如何保留教材原意部分，為本實驗最重要的研究主題之一。

3.5 多媒體內容協調後結構完整性需求

大部分的多媒體內容設計時，都以桌上型電腦為出發點，所以在 PDA 畫面

呈現的範圍受限制是可以預期的，一般的作法是使用拖曳軸的方式做呈現，但由於畫面太小，使用者必須反覆的拉動拖曳軸，這樣的操作方式對於使用者來說相當的麻煩。如果我們使用畫面分割的手法處理，測試效果是否會比較適合，圖 3- 3 為交通大學的入口網站，當初設計時是以 800*600 的顯示規格，對於 PDA2 的顯示畫面 240*320 來說，圖片中斜線部分就是 PDA 可顯示的範圍，利用面積的計算，PDA 需要 6.25 個(800x600 / 240x320)畫面才能將整個網頁呈現出來，但是實際上在協調的處理時，對於圖片和文字會做相對應的縮小動作所以 6.25 個畫面可視為最大值 (worst case)。



圖 3- 3 原始網頁與 PDA 顯示範圍比較

圖 3- 3 中顯示，從交大的入口頁面連線至另一個網頁，而當協調系統對於

這個頁面做適當轉換與協調後可能將網頁切割成幾部分顯示，如圖 3- 4 所表示的情形，右方的三個頁面為協調系統因為 PDA 畫面限制而所做的切割動作，所以整個內容可以完整地呈現給使用者，但是這樣的協調方式衍生出另一個問題，原本指向到這個頁面的連結變成有三個目標可以選擇，所以在處理連結這部分必須做精確的處理，以確保協調完成後，多媒體內容結構的完整性。



圖 3- 4 畫面切割完成後結構

四、系統架構

4.1 系統架構圖

系統總架構大略如圖 4-1 所示：

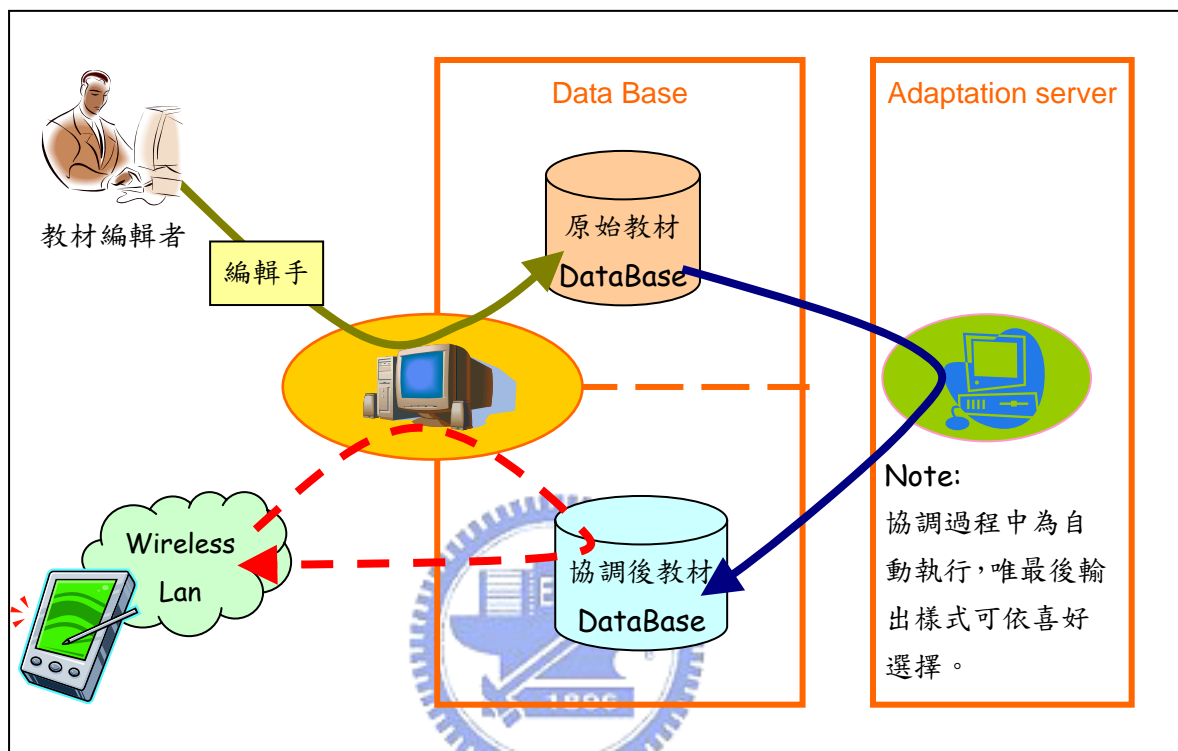


圖 4-1 系統架構圖

系統運作三個主要流程：

- 教材編輯者使用編輯手[6]製作教材完成後，上傳至原始教材資料庫。
- 轉換協調伺服器從原始教材下載 PC 版本的教材，針對 PDA 版本做相關的轉換協調動作，完成後上傳至協調後資料庫。
- 手持行動裝置(PDA)使用者經由無線網路連線至線上資料庫，選擇需要的教材檔案，下載完後解壓縮播放使用。

由這三個主要流程可知道，整個系統可大略分為四個部分，其中教材製作部分，由於是使用已開發完成的多媒體編輯軟體編輯手，所以本研究暫不討論：

- 教材製作
- 轉換協調系統
- 線上內容資料庫系統
- 手持行動裝置呈現應用程式

各子系統詳細的架構接下來的章節會做詳細的介紹。

4.2 轉換協調系統架構說明

4.2.1 下載教材運作流程

轉換協調伺服器(Adaptation Server)先連線至教材內容伺服器(Content Server)下載需要協調之教材。

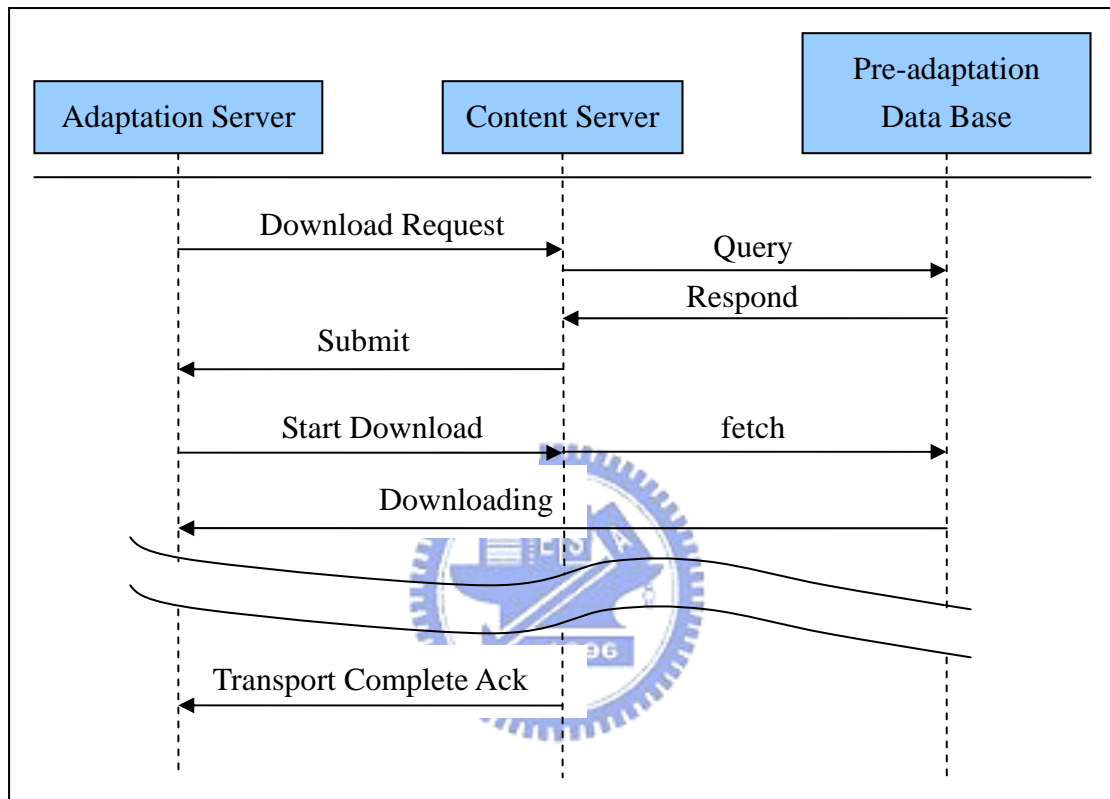


圖 4-2 下載功能運作流程圖

下載功能的運作流程圖如圖 4- 2 所示，轉換協調伺服器發送下載檔案需求給內容伺服器，內容伺服器收到需求後會先至存放教材資料庫系統查詢所需要的檔案是否存在，確認存在後回傳允許下載訊息給轉換協調伺服器，轉換協調伺服器得到確認回覆後發送開始下載的訊息給內容伺服器，內容伺服器找到下載的目標檔案並發送給轉換協調伺服器，直到檔案完全下載完成後，內容伺服器送傳輸完成訊息給轉換協調伺服器，整個下載動作才算完成。

4.2.2 主要協調部份之架構與流程

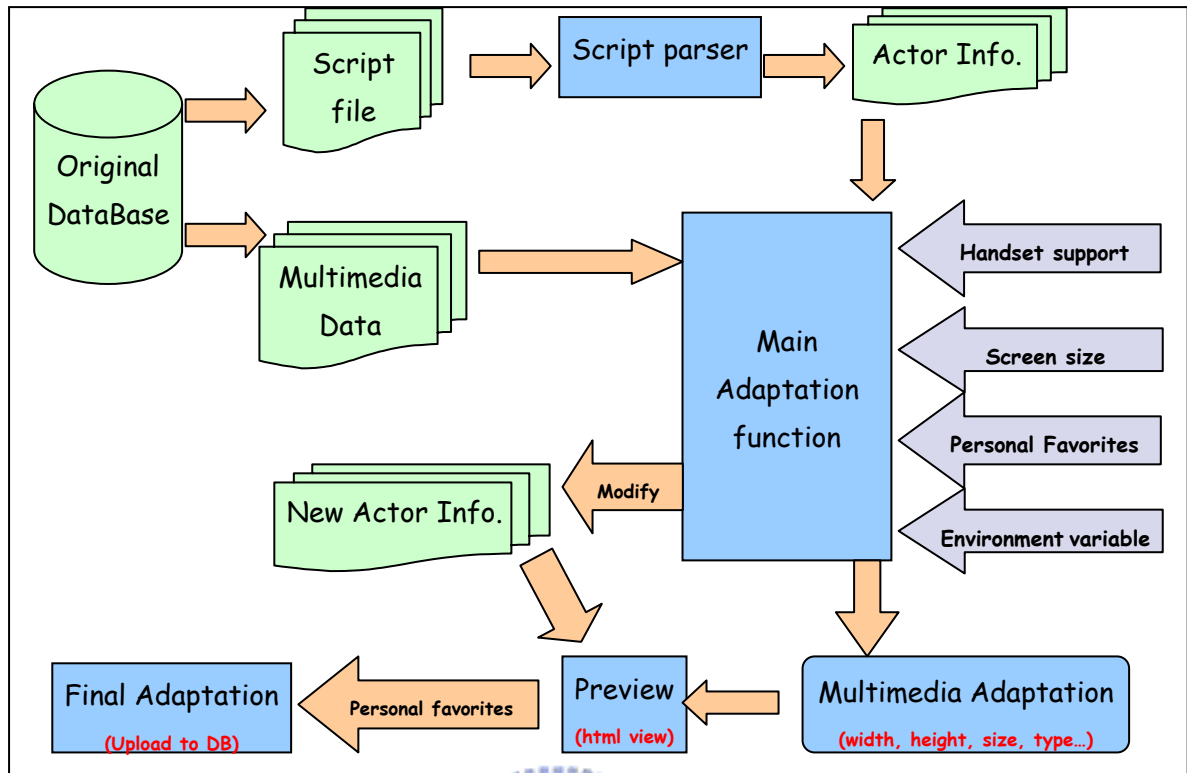


圖 4-3 主要協調部分之架構

轉換協調系統(Adaptation System)主要的架構如圖 4-3 所示，其中再分割成三個更小的部分，分別詳細地說明各部分運作方式：

(a). 描述檔解譯模組 (Script parser module)

將由編輯手編輯完成後，教材檔案中的教材描述檔，根據其原始定義的規則，將每一個演員的定義(位置、顏色、大小...等等)分別儲存到自訂的資料結構中，而描述檔中最重要的劇情部分也紀錄到自訂的劇情管理員，之後將這些資訊提供給主要協調中心(Main adaptation function)以供決策。

(b). 轉換協調模組 (Main adaptation function)

這是整個系統核心部分，它會先接受由描述檔解譯模組 (Script parser module) 送來的演員資訊和劇情管理員，再將由協調管理者送來的變異係數，其中如果某些變數管理者沒有設定，例如字型大小，可以採預設變數。根據這兩大資訊，決策中心會分析 (Analysis)、決策 (Make policy)、暫時轉譯 (Temporal translation)，之後送交給輸出模組 (Output module) 給使用者預覽協調的結果，此時系統便等待回傳使用者選擇結果，最後根據管理者選擇的方式改寫新的教材描述檔，並且變更使用者資訊和劇情管理員，過程中還必須進行圖片、影像、聲音的

轉換，等一切都處理、改寫完成，再次送到輸出模組輸出成最後完成的版本。

(c). 輸出模組(Output module)

主要接受轉換協調模組(Main adaptation function)送來的所有資訊，將這些資訊輸出到指定的樣式，協調中的第一次送來的初始版本會輸出成網頁(html/php)版本，搭配微調功能選項一起呈現給協調管理者，等待管理者選定之後，回傳給轉換協調模組，等下一次收到輸出要求時，便將所有資訊輸出至提供給 XMG Player 讀取的 XMG 描述檔案，並且將檔案壓縮完成。

4.2.3 上傳協調完教材運作流程

轉換協調系統將已經協調完成的教材內容壓縮打包後，上傳至內容伺服器(Content Server)中，存放至已轉換協調資料庫中。

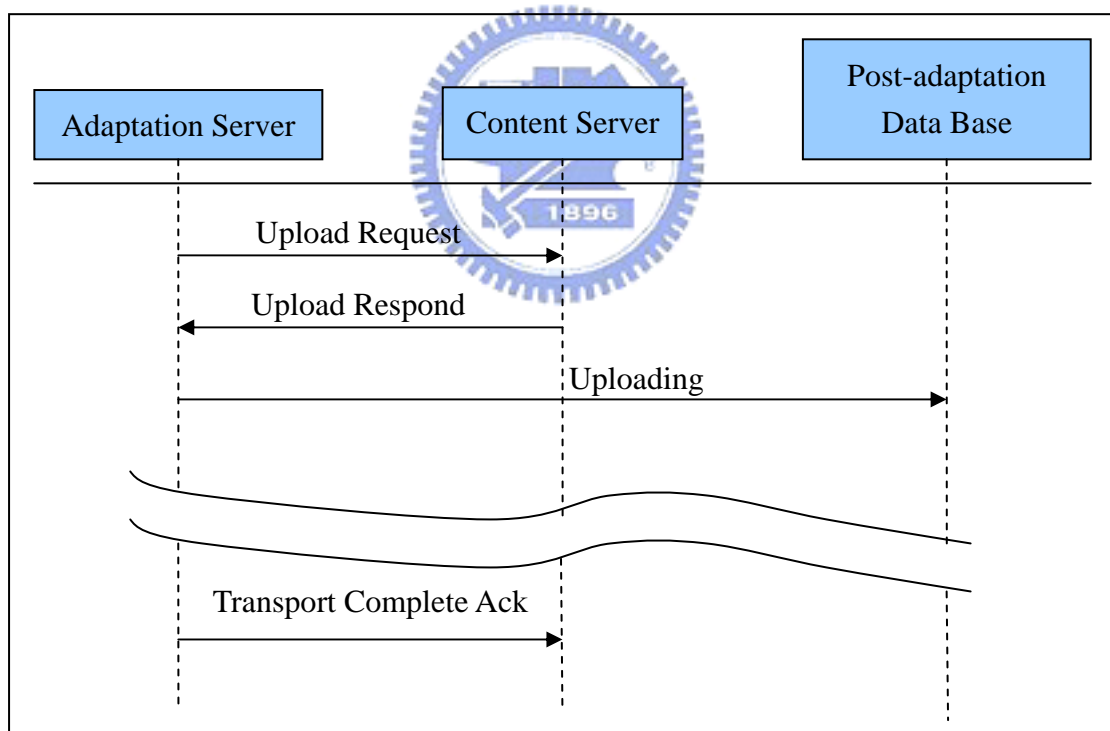


圖 4-4 上傳功能運作流程圖

如圖 4-4 所示，轉換協調伺服器會先送上傳要求至內容伺服器，收到回內容伺服器回應之後，開始上傳目標檔案，等到檔案已完全傳輸完畢，在發送訊息通之內容伺服器已經完成再結束之間的溝通。

4.3 線上內容資料庫系統

線上的資料庫分為兩個部分：未協調內容(Pre-adaptation)和已協調內容(Post-adaptation)，由內容伺服器統一管理，表 4- 1 是資料庫的欄位設定：

表 4-1 資料庫欄位

	欄位	型態	屬性	Null	預設值	附加
<input type="checkbox"/>	no	int(11)		否		auto_increment
<input type="checkbox"/>	filename	varchar(100)		否		
<input type="checkbox"/>	date	date		否	2007-01-01	

內容資料庫管理者可直接執行 SQL 語法來新增、刪除、查詢資料庫內容，也可以使用現下較常被應用的免費軟體 phpMyAdmin[21]，透過它所提供的視窗介面，能方便管理者檢閱資料庫內容，而不需要逐一輸入繁瑣的 SQL 指令，圖 4- 5 為內容資料庫的架構圖。

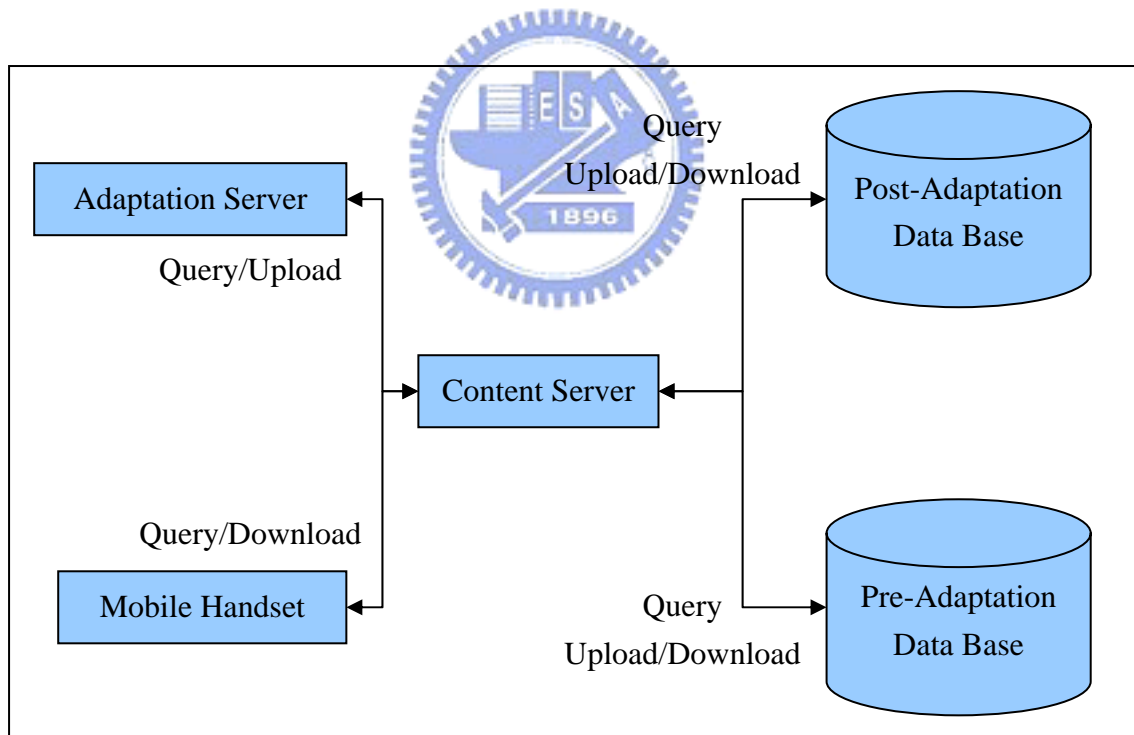


圖 4- 5 內容資料庫架構圖

4.4 手持行動裝置呈現應用程式

在本實驗中，手持行動裝置是使用 PDA 來呈現多媒體教材，第一步 PDA 會先連線到教材資料庫瀏覽清單並下載，之後再播放所選擇的檔案，圖 4- 6 為

PDA 選擇教材下載流程圖：

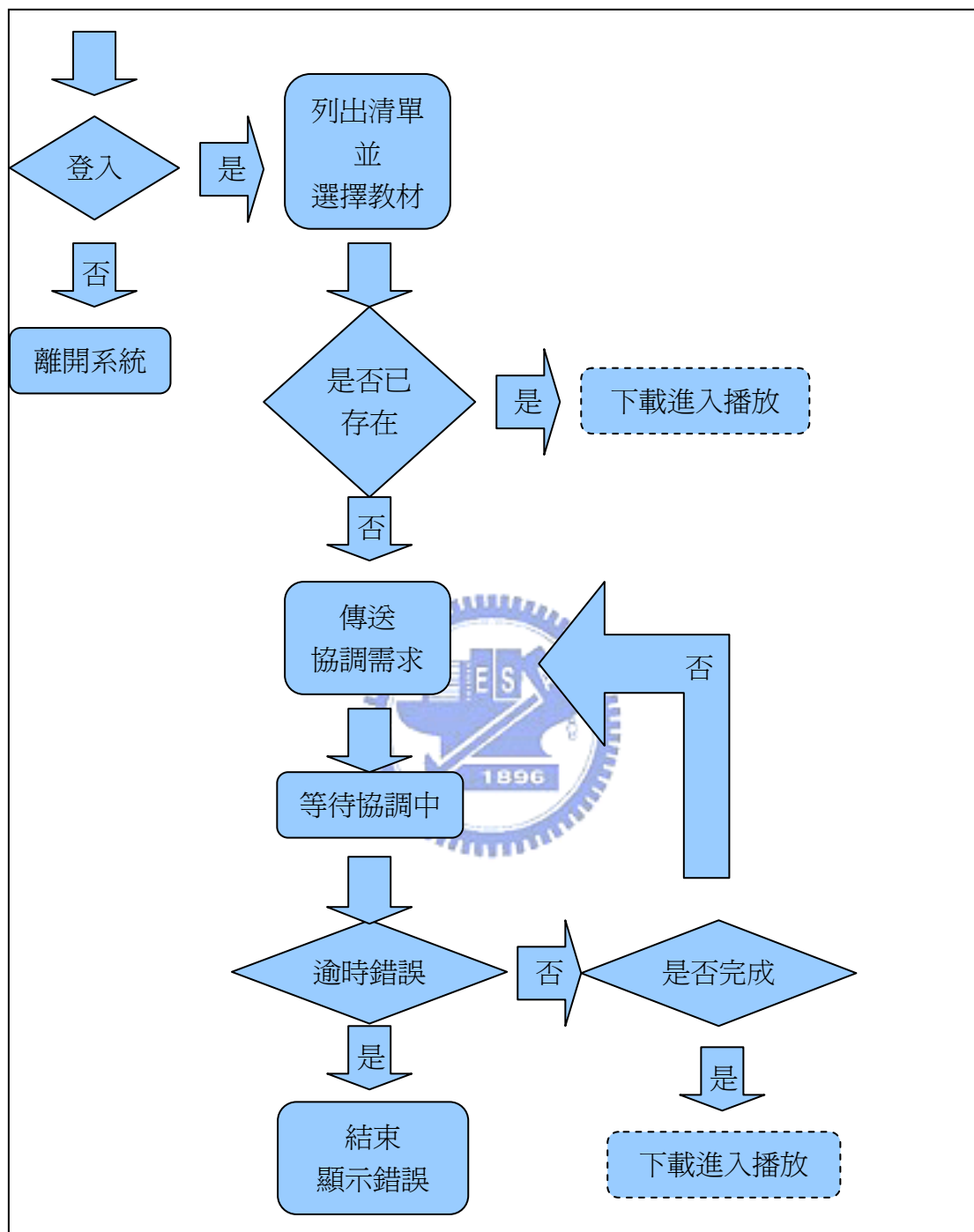


圖 4- 6 PDA 下載教材流程圖

PDA 使用者利用 PDA 連線到教材資料庫入口登入系統，如果登入成功後伺服器會送給 PDA 一份教材清單，使用者可任一點選一份下載，送出需求後，如果所選擇的教材已經協調完成，伺服器會立即傳送，如果此份教材的協調版本還不存在，PDA 會等待協調伺服器進行改編協調，其間會持續偵測是否逾時錯誤

或已經完成協調，等偵測已完成後便可進行播放。

進入播放流程後，首先將檔案中的 EBP 加密檔案解密，接著將解密後的檔案解譯(parsing)，XMG 播放器描述檔是利用類似於 XML 的標籤語法描述教材的演員資訊和劇情，一方面是設計較簡單，另一方面是解譯時除錯也相對於容易，圖 4- 7 是 XMG 描述檔的架構。

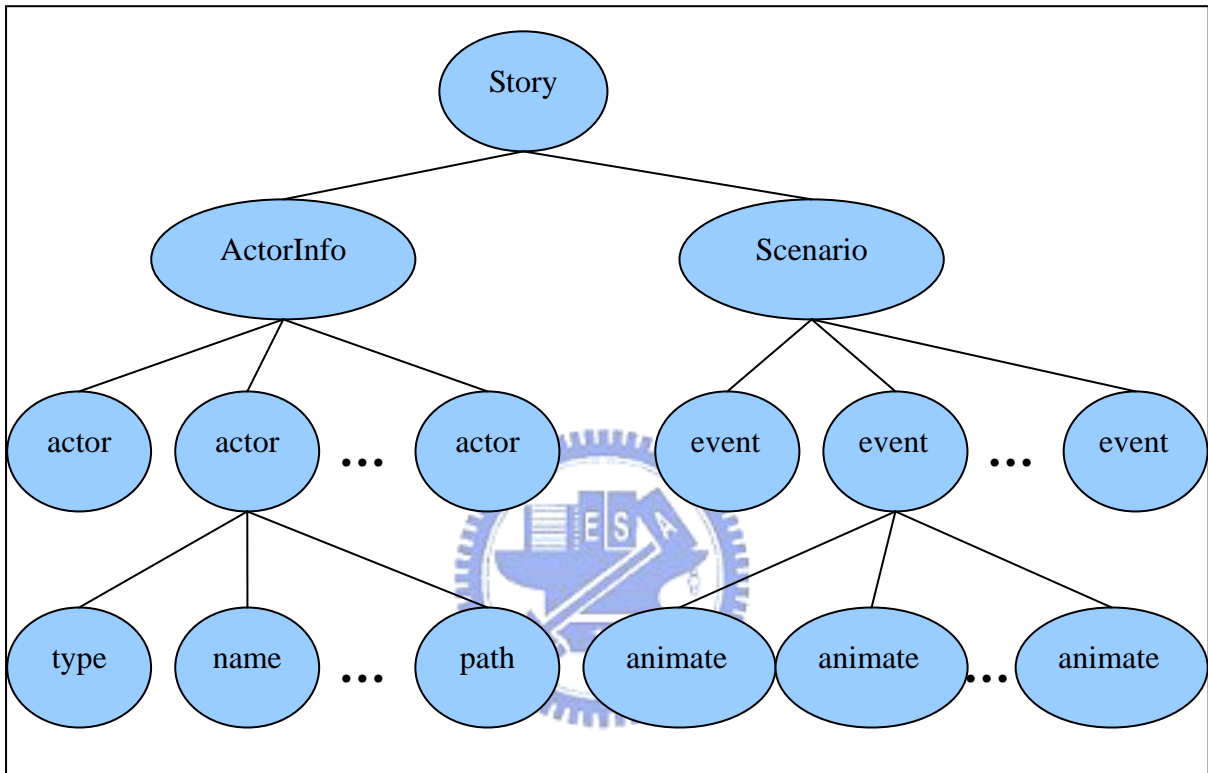


圖 4- 7 描述檔架構

<Story>將整份教材包裝起來，其中分為兩大部分，先將<ActorInfo>中的角色資訊讀入記憶體中，每一個<actor>中有<type>、<name>、<path>等等可利用的資訊，第二部分是劇情的部分，設計的概念是以事件為一個單位，由於是在 PDA 上做呈現，當使用者在螢幕上點某個演員，就視為一個事件，事件發生時看樹狀結構中哪些< animate>負責演出就將演出者呈現出來，其中需要聲音及影像的播放時，呼叫 PDA 本身提供的播放程式來負責，這就是大略的播放流程。

五、系統設計與實作

5.1 描述檔解譯器的實作

由前一章的說明，轉換協調系統(Adaptation Server)接收完檔案會先解譯其中的加密描述檔 EBP 檔案，圖 5- 1、圖 5- 2 是描述檔的演員設定和劇情設定部分：

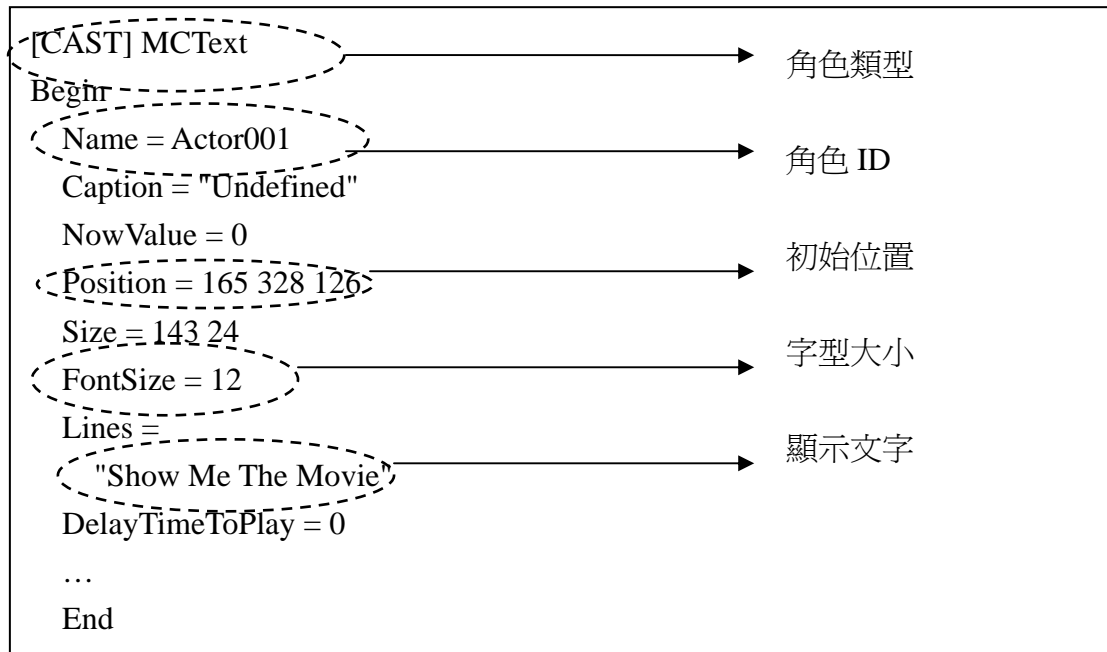


圖 5-1 演員描述檔

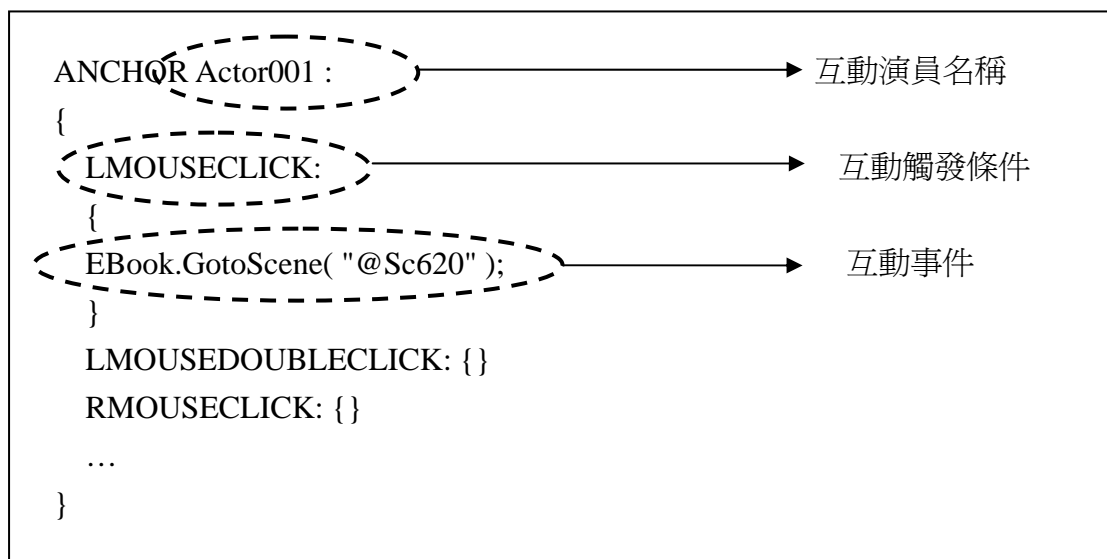


圖 5-2 互動式劇情

由描述檔的說明我們可以清楚的知道所有演員的各種屬性，以圖 5- 1 的文字演員為例子，“MCText” 是代表這個演員是文字演員，其他還有圖形、影像... 等等，name 是系統主動指定的系統 ID，以利系統方便使用，其他還有文字大小、初始位置，圖形有大小、圖層順序等等各種詳細的設定。互動式劇情部分以圖 5- 2 為例子，由於是在 PDA 上的呈現，所以大部分的互動是劇情以點擊畫面的事件為主，當使用者點擊此演員後，系統就會執行“EBook.GotoScene” 這個事件，描述檔的定義相當簡單但又功能性十足。

已經將描述檔解密完成，再來就是逐一解譯並轉換至自訂的資料結構以方便使用，在解譯過程中，我們使用最基本的組譯(compiler)技術，透過 Lexical 產生字串(token stream)，之後給 yacc 產生文法檢查器作文法的比對，產生最後的 parsing tree，解譯出來的 parsing tree 會和圖 4- 7 一樣，這樣一來便可以根據此 parsing tree 轉換至自訂的資料結構，如圖 5- 3：

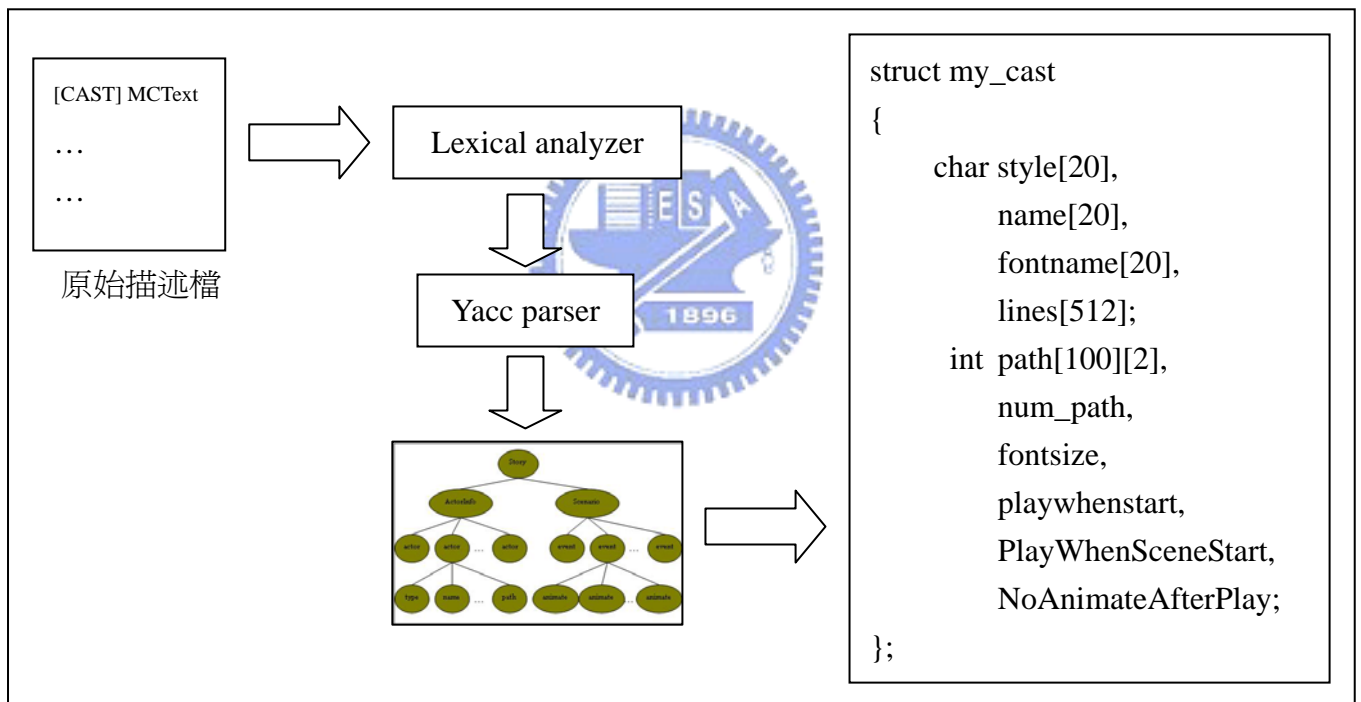


圖 5- 3 解譯描述檔流程

在此本研究使用 Vector 這個資料型態來串接所有 my_cast 自訂資料型態，會選擇使用的原因是因為 C/C++ 在 Vector 上提供了許多方便的 API：可以同時使用循序搜尋(Sequence Search)和直接搜尋(Direct Search)、直接使用 my_cast.Size() 便可得到所有演員的總數等等工具，最重要的是 Vector 的使用可以動態宣告空間，對於系統的容錯性較高，這就是選擇使用的原因，串接起來的結構會像圖 5- 4 一樣。

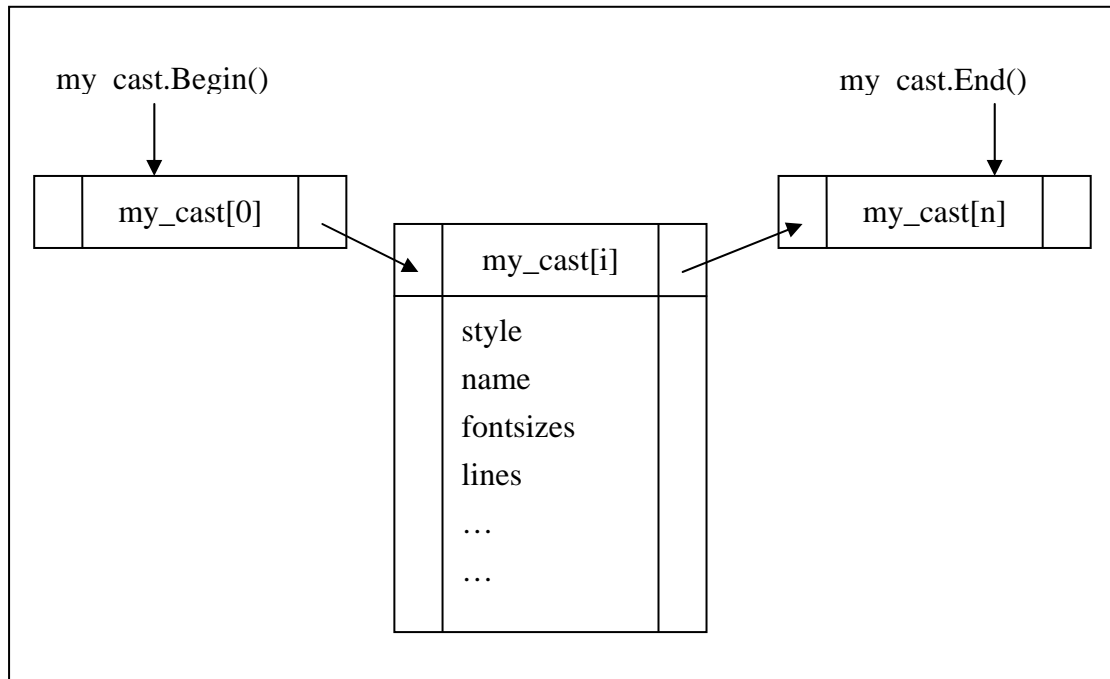


圖 5- 4 Vector 結構

5.2 轉換協調系統的實作

本章會先討論單純的比例協調的好壞處，為何捨棄使用單純的比例協調，之後在根據文字、圖片、影像分開協調的演算法和考慮的原因，最後再將再將三種類型的教材元素合組成一個畫面。

5.2.1 單純的比例協調

單純的比例協調的作法是根據原始教材設計的大小和目標裝置的大小依一定的比例把所有的角色縮小，這種作法可以快速的處理任何版本的教材，不需要額外的演算法處理，現在一般市面上很多網站的處理都是採用這樣的方法，相當的普遍，不過本研究不採用等比例縮放的方法，在這個小節提出兩個原因說明。

手持行動裝置以 PDA 為例，顯示器的規格是以高度大於寬度的方式，而一般教材在製作時，為了講師教學方便，都會製作成支援個人桌上型電腦或筆記型電腦，寬度大於高度的螢幕類型，所以在顯示能力上就有很大的差異性，如果單純利用等比例縮小的方式，勢必只能遷就於 PDA 的寬度，所有物件的屬性都乘上一個定值，這個定值 $R = (\text{PDA 寬度} / \text{PC 寬度})$ ，這樣子的結果雖然所有物件都不會超出邊界，但是顯示的畫面就會像圖 5- 5 一樣，PDA 顯示畫面只佔了上半部，且在文字部分因為等比例的效果，已經模糊不清，造成使用者無法閱讀，這份教材就失去教學的功能，而 PDA 顯示畫面已經相當有限，所以空間相對地

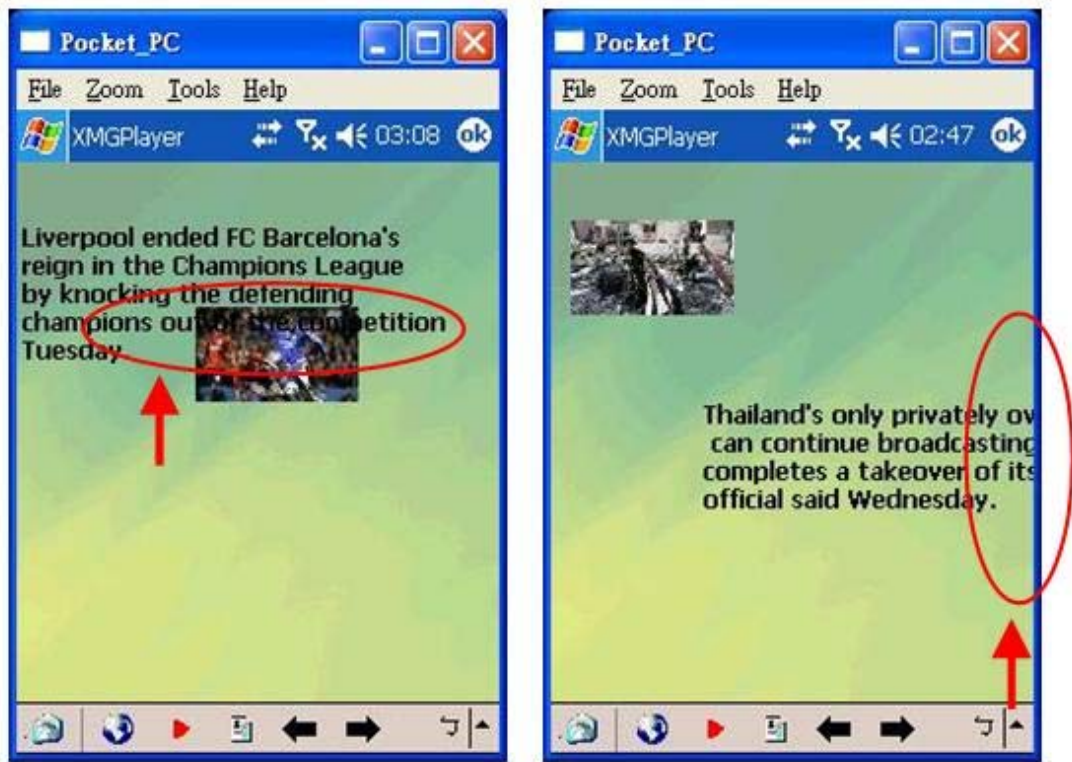
珍貴，等比例縮放的結果，造成下半部斜線部分完全沒有使用，這樣子的結果顯然不是我們想要的。



圖 5-5 單純等比例協調結果

如果對於文字的部分放大顯示，而其他物件仍然採等比例的方式，是否會產生讓使用者有良好地閱讀，且又可以不超出畫面的顯示範圍呢？圖 5-6 就是這樣的結果，原本每個物件都是採同一個比例縮小，而文字放大了一些，但是對於原本其他圖片的位置座標來看，所佔的面積相對來的大許多，所以會發生圖 5-6 左邊的文字覆蓋到圖片的問題，而放大的文字對於顯示範圍也是一大問題，因為文字的原始座標再加上放大過的面積，會發生圖 5-6 右邊文字超出畫面的問題，對於這樣的結果也不是我們想要的。

由這兩個簡單的例子可以知道，單純的等比例協調並不能解決所有的情況，需搭配其他方式才能將畫面協調至較為適合的程度，本研究先針對製作多媒體教材時較常用的三個元素：文字、圖片、影像分開做協調的動作，完成後將之合成一個畫面，探討此作法之可行性。



文字覆蓋圖片

文字超出範圍

圖 5-6 文字放大顯示後的等比例協調

5.2.2 圖片的協調

圖片在協調時最需要注意的事就是高度和寬度的比例，如果在縮放的過程中，長度和寬度乘上不同的比例，圖片就會發生變形失真的情形就如圖 5-7 所示，不易辨識圖片的內容。如果要使用同一比例，該如何選擇？上一節提到 PDA 的顯示畫面是以高度大於寬度的方式，若能改良等比例協調方法，應用到圖片的協調。

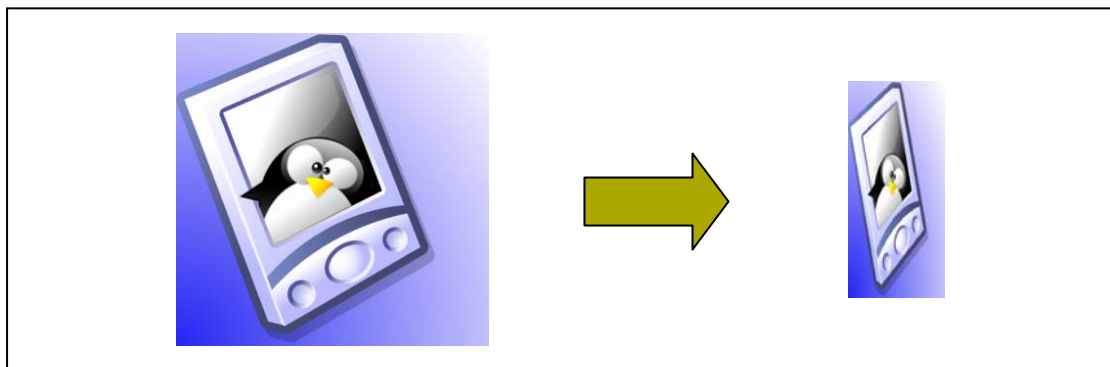


圖 5-7 圖片高度寬度以不同比例縮放

在縮放比例 R 的選擇有兩種方式， R_w (以寬度為基準)和 R_h (以高度為基準)，運算的方式如表 5- 1 所示，以 PC 畫面 $640*480$ ，PDA 畫面 $240*320$ 為例，轉換出來的結果會如圖 5- 8 顯示的樣子，以單一種比例的縮小會保持原有圖片的樣式，如果以不同比例做轉換就最下方的圖一樣拉長而失真，這樣的方式選擇不採用，故基本上選擇單一比例的縮放方式。可以選擇的方式有兩種，雖然的 R_h 結果圖片比 R_w 大，但是相對於 PDA 的畫面寬度，便有超出邊界的可能性，所以在圖片縮放的比例我們選擇使用 R_w 。

表 5- 1 R_w, R_h 運算規則

寬度比 $R_w = w_2 / w_1$	(w_1, h_1 原始教材寬度、高度)
高度比 $R_h = h_2 / h_1$	(w_2, h_2 PDA 顯示畫面高度、寬度)
協調後圖片長寬：	
寬 $I_w' = I_w * R_w$	
高 $I_h' = I_h * R_w$ (以寬度比為基準)	
協調後圖片座標：	
X 軸 $I_x' = I_x * R_w$	
Y 軸 $I_y' = I_y * R_h$ (以高度比為基準)	

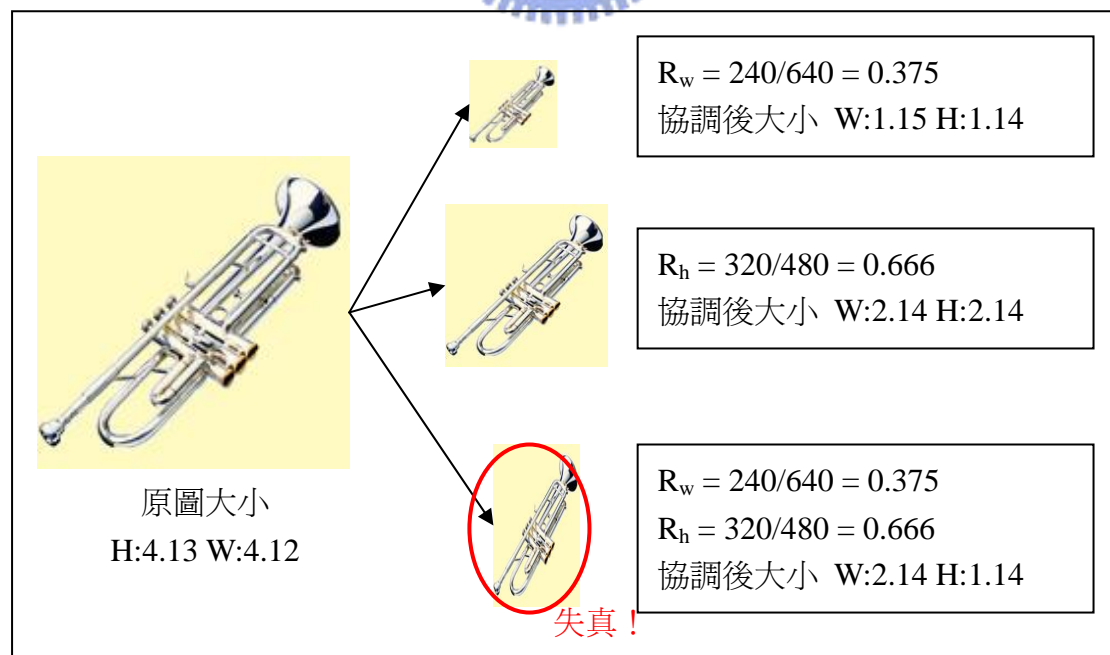


圖 5- 8 以 R_w, R_h 轉換後的圖片

放置圖片的座標也因為 PDA 顯示器大小的關係，也必須做縮放的動作，同樣的是考慮 R_w 與 R_h 的比例，圖 5- 9 左圖是以 R_w 的比例縮放，所有物件都集中在上方，與之前所討論的情形相同，而圖 5- 9 右圖是以 R_h 的比例縮放，相對於 X 軸的變動量較大，所以會超出畫面的邊界。這樣的結果顯示，單一一種的比例協調都不能滿足我們的需求，故在座標的比例我們採用混和使用方式，對 X 軸使用 R_w 的比例協調，對 Y 軸使用 R_h 的比例協調，結果如圖 5- 10 所示，這樣圖片自然會往下移動，不浪費顯示的空間，也不會有超出畫面範圍的問題。

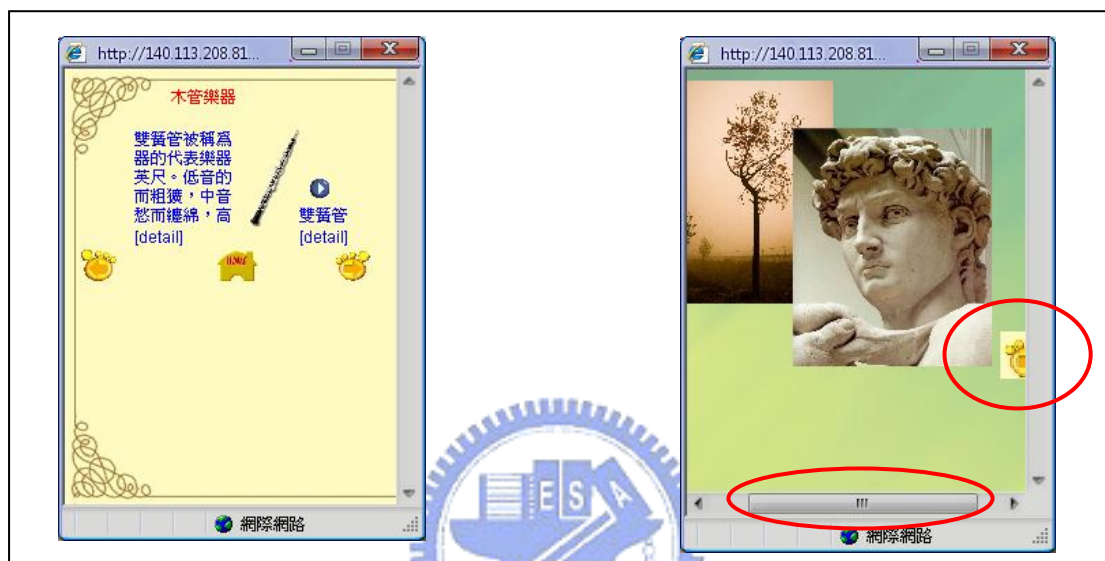


圖 5- 9 座標比例縮放情形



圖 5- 10 圖片協調結果

5.2.3 文字的協調

文字的協調首要的目標就是必須讓使用者能清楚的閱讀，但是不能超出範圍和覆蓋到其他物件，第一步是計算 PDA 顯示能力和文字的像素值(pixel)，英文字型的所佔的面積在本研究室先前的研究結果[3]已做詳細的討論，所以在本研究延伸討論中文字部分，表 5- 2 是中文字在桌上型電腦使用不同作業系統的顯示型態：

9pt	12px	Windows 標準小號尺寸
10pt	13px	一般建議用於內文段落的最小適讀字尺寸
11pt	15px	略近於 DOS 下倚天中文系統 16 點陣字型樣貌 (字型檔實為 16x15 px)
12pt	16px	相當於 DOS 下宏碁中文系統(DOS 3.21 中文版) 16 點陣字型樣貌 (字型檔為 16x16 px), 線條轉角處較上者多修飾。亦為瀏覽器裏未指定尺寸的一般文字之預設大小(font size="3" or "0")
13pt	17px	Windows 既不能以「分明線條」、又不能「平滑化修飾」(derasterization)來顯示的尷尬尺寸。其美觀性甚差, 避免使用。
14pt	19px	標題字常用尺寸。14pt 及 18px 為 Windows 對於(新)細明體能進行「平滑化修飾」的最小向量尺寸, 即以「類印刷品質」樣貌來顯示的最小字型尺寸。

表 5- 2 中文字型在桌上型電腦上顯示規格

以 12 號字型為例，在桌上型電腦上文字的高度和寬度為 16*16px，但是在 PDA 上顯示規格和一般電腦不確定不相同，所以我們做了以下的測試，以 PDA 顯示畫面 240*320px 為例，使用字型大小 font size=14 來顯示，填滿 PDA 畫面的高度與寬度，如圖 5- 11，因為中文字形的設定為同一種字型每一個字的大小相同，所以可做以下的計算：



圖 5-11 中文字型的計算

粗略的計算後，14 號字型、新細明體的中文字，字型寬度為 12px，字型高度約為 16px，利用這樣的方法得到各個不同大小文字的高度和寬度，不過這裡有一個困難點，中文字型的樣式多種，新細明體、標楷體...等等，如果再加上用來設計海報、平面廣告等等的多媒體藝術字型，例如華康一系列的藝術字型等等，如果要每一種字型都做相同的偵測動作，系統的複雜度提高許多，為了解決這個問題，本研究先統一將文字輸出成“新細明體”一種字型，一方面新細明體是許多文件和網站的預設顯示字型，較能讓使用者接受，另一方面對於降低系統的複雜度有顯著的功用。

當我們掌握了文字的尺寸，接下來便可以進行文字的協調動作，文字的座標部分和圖片相同是利用 R_w (以寬度為基準)和 R_h (以高度為基準)分別對 X 軸和 Y 軸做協調，結果的畫面應當會像圖 5-12 所示，因為無法預期設計的樣式，所以文字覆蓋到圖片的情形一定會發生。



圖 5-12 文字覆蓋畫面

因為文字的尺寸已經知道，所以系統可透過計算面積的方式，計算是否超出 PDA 顯示的邊界，與鄰近的图片是否發生重疊的情形，判定的方式如下方的虛擬碼：

```

bool checkboundary(int x, int y, int w, int h)
{
    if x+w > Width
        return 1; //超出 PDA 寬度邊界
    else if y+h > Height
        return 1; //超出 PDA 高度邊界
    else
        return 0; //無超出邊界
}

```

```

bool CheckOverlap(int x1, int y1, int w1, int h1, int x2, int y2, int w2, int h2)
{
    if overlap
    {
        return case 1: //圖片 1 由左方覆蓋圖片 2
        return case 2: //圖片 1 由上方覆蓋圖片 2
        return case 3: //圖片 2 由左方覆蓋圖片 1
        ...
    }
    else
        return 0; //無發生覆蓋
}

```

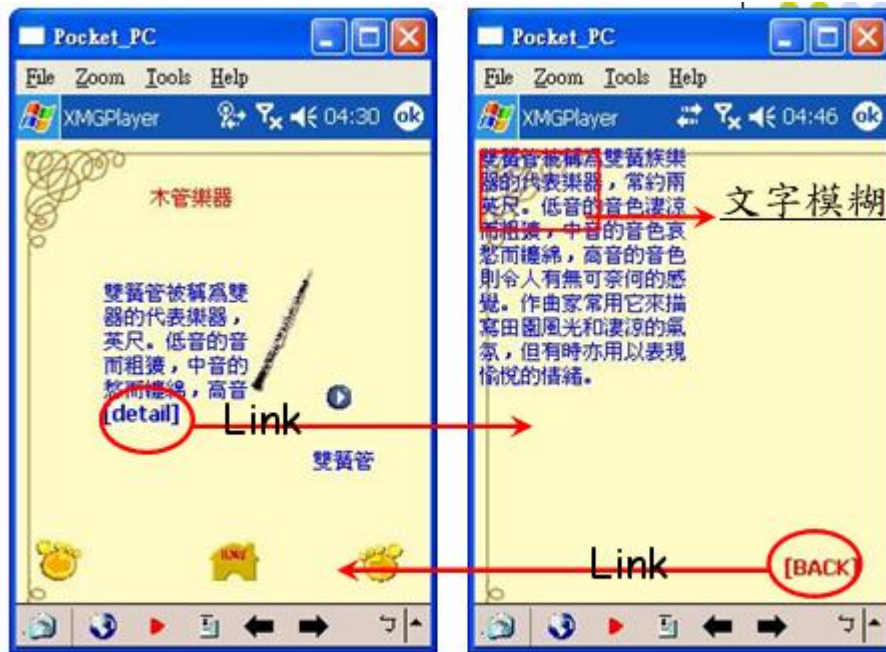
若判定重疊後開始進行協調，協調的方式是採用和網路新聞文字版的方式，僅顯示幾行當作標題，讓使用者先初步的閱讀，而文字重疊的部分濾掉不顯示，再加上“[detail]”當作新的連結，連結到新增加的頁面，完整的文字訊息擺放至新的頁面上，當使用者閱讀完初步的訊息想進一步瞭解內容，便可點選連結到新的頁面檢視，文字的協調過程虛擬碼如下：

```

void Ad_Text(int casts_id)
{
    if(CheckBoundary() || CheckOverlap())
    {
        //Step1. 顯示沒有覆蓋的文字
        //Step2. 新增新的頁面，放置完整的文字
        //Step3. 新增文字[detail]製作連結至新增頁面
    }
    else
        //按照原本設定正常輸出
}

```

文字協調完成結果如圖 5- 13 所示，如此一來使用者在觀看完第一頁的樂器圖片介紹後，可點選連結看完整訊息，等觀看完整段文字再回到前一頁繼續整份教材的流程。



協調後畫面

圖 5- 13 文字的協調

在新增頁面擺放完整文字訊息時還有一個需要探討重點，擺放的位置原本是以原點(0,0)開始放置文字，以期望能顯示最多的文字數量，但因為教材製作時的背景圖可能不是使用單純的背景色，有可能是圖片或花紋，而這些顏色對於文字的影響，有可能讓文字變的不易閱讀，如圖 5- 14 中左上角的文字，因為背景的花紋變的模糊。

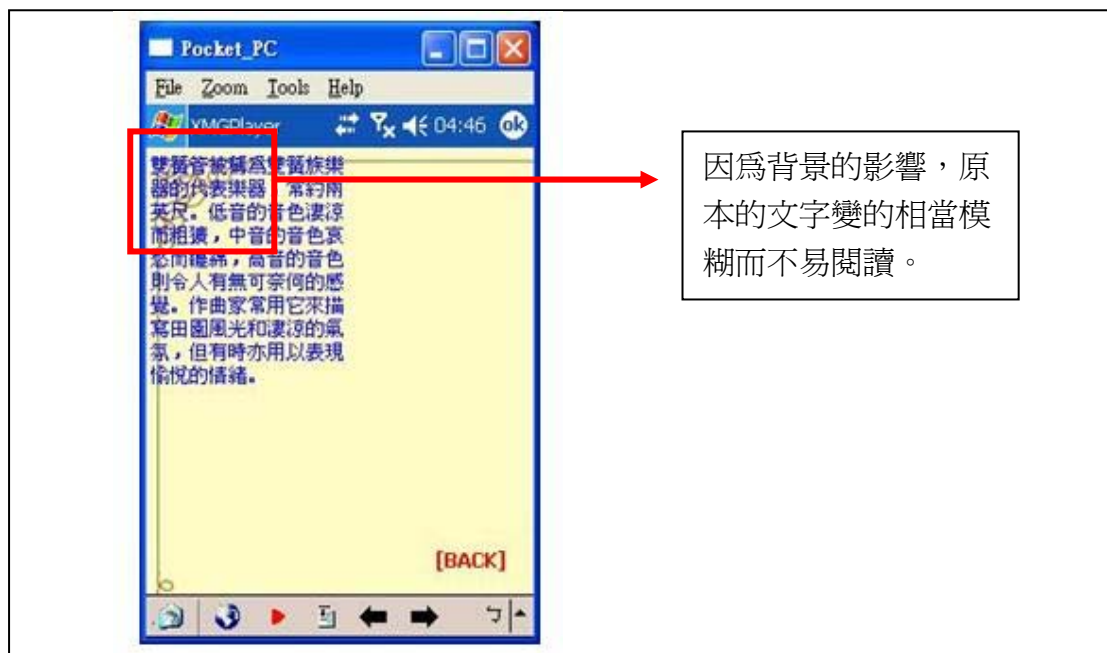


圖 5- 14 背景色對文字顯示的影響

因為背景色的影響，使得系統不知道應該將文字擺放在哪一個位置，為了解決這個問題，文字覆蓋判定後，在進行協調前先將背景圖送給副程式判定應該擺放在那個位置比較適當，以上面的背景色為例子，將每一個座標的 pixel 值讀出判斷顏色深淺，進而判定某個區塊的平均顏色深淺，如此一來系統知道從何處開始擺放文字，以下為加上判斷背景的文字協調虛擬碼，協調完成的結果就像圖 5- 15 所示：

```
void Ad_Text(int casts_id)
{
    if(CheckBoundary() || CheckOverLap())
    {
        //Step1. 顯示沒有覆蓋的文字
        //Step2. 判斷背景圖片 checkBgImage(image, x, y)
        //Step3. 新增新的頁面，根據 step2 的結果放置完整的文字
        //Step4. 新增文字[detail]製作連結至新增頁面
    }
    else
        //按照原本設定正常輸出
}

bool CheckBgImage(char* image, int x, int y)
{
    int total;
    int key;        //設定一個標準值
    for(int i=0;i<x;i++)
        for(int j=0;j<y;j++)
        {
            int R, G, B = image.GetPixelColor (image, i, j)
            //讀取(i,j)的 RGB
            if(RGB > key)    //如果 RGB 的顏色大於標準值
                total++;
        }
    if(total > 1/4*x*y)
        return 1;        //如果深色座標達 1/4 的面積，判定為深色區塊
    else
        return 0;
}
```

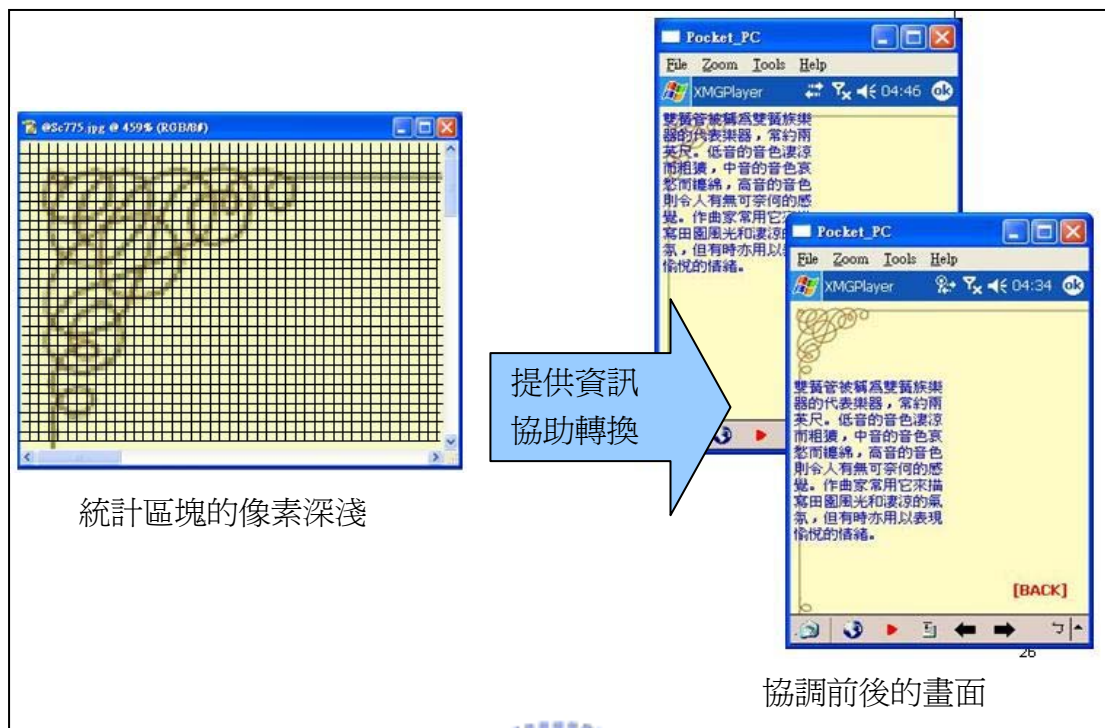



圖 5-15 判定背景色協助文字協調

如上面的步驟就完成文字覆蓋圖片的協調過程，其他有關文字的處理，例如文字超出顯示邊界、文字覆蓋到文字，也是以類似的方法處理，有一點小小不同的地方就是當兩個文字相互覆蓋時，Y 軸數值較小的稱為上方文字，另一個稱為下方文字，處理的方式視為上方的文字覆蓋下方的文字，只針對上方的文字做協調的動作，而下方的文字不因為這次的覆蓋而做協調，以減少整個畫面都是連結和新增太多額外的頁面，但如果下方的文字發生超出邊界或覆蓋到其他物件，仍然必須進行協調，不過與上方文字的協調屬於不同的事件。

5.2.4 影像的協調

教材製作者利用簡短的影片播放，來傳遞給使用者書本以外的教學資源是一個很普遍的設計方式，為了讓使用者能觀看清楚影像的內容，大部分都會將影像的尺寸盡量放大，甚至於全螢幕的播放，對於 PDA 的顯示能力來說，原本的畫面已經收受到限制，影像的大小還遠大於能顯示的範圍，所以對於影像尺寸必須做縮小的動作。同樣的我們期望在協調完成後，使用者可以以最大尺寸的方式觀看，故現下有兩個方式選擇：

- (a). 遷就於 PDA 高度
- (b). 遷就於 PDA 寬度

由於 PDA 是高度大於寬度的顯示畫面，我們優先選擇使用遷就於高度的方法，

讓畫面最飽滿，畫面如圖 5- 16 所示，因為原始影像是寬度大於高度的規格，如果如此協調，影像會發生拉長、失真的現象，對於使用者長時間使用相當吃力，而且不易清楚檢視影像的內容，所以我們選擇第二種方法，遷就於 PDA 的寬度。



圖 5-16 遷就高度的影像協調

以 PDA 可顯示的寬度當作影像的寬度，盡量讓使用者能夠清楚地觀賞，但是我們讓影像佔了這麼大的部分，如果在同一個頁面有其他的文字、圖片該如何協調？如圖 5- 17 左邊的圖相同的情形，其他的物件變的沒有地方可以擺放。



除了影像還有其他物件

分頁後協調

圖 5- 17 影像的協調

解決的方法可利用之前新增分頁的方法，先將第一頁指定給影像使用，其他剩下來的文字、圖片的物件搬移到下一個頁面，這樣一來使用者觀看影像後可至下一頁面閱讀說明文字或圖片，如有不清楚的地方可再回到第一個頁面再次觀看影片，而第二個頁面必須再一次額外的協調動作，以避免文字和圖片發生超出顯示範圍或覆蓋的情形，詳細的影像協調步驟與虛擬碼如下，協調完成後結果應為圖 5- 17 右邊顯示的情況。如果同一頁面有兩個影像存在的話，和文字的協調方法類似，對於 Y 軸數值較小的先做協調，然後剩下的影像和其他物件另外做一次協調動作，流程大略如圖 5- 18 所示，用這樣的方式來解決多影像在同一畫面的情形。

```

if(Have_Movie)
{
    //step1. 影像以 PDA 寬度縮小，並放置於(0,0)
    //step2. 將其餘演員送至新增頁面
    //step3. 新增文字[Info]並連結至新頁面
    //step4. 呼叫另一次的 Main_Adaptation()處理第二頁面
}
Main_Adaptation()
{
    Ad_Text();
    Ad_Image();
    Ad_Movie();
    ...
}

```

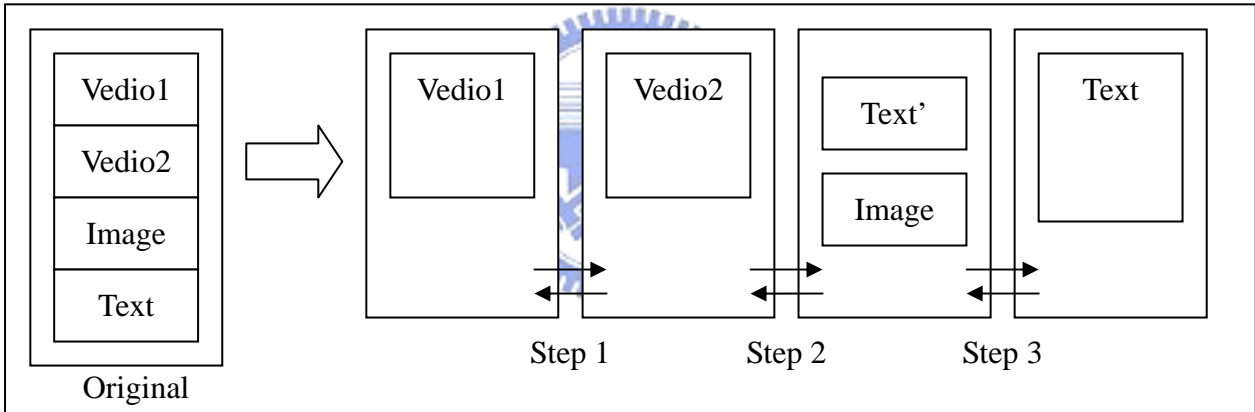


圖 5-18 多個影像協調步驟

5.3 應用端播放器的實作

應用端播放器主要沿用先前設計來展示多媒體英文試題的 XMG Player，在本研究中為了轉換多媒體教材的功能需求，新增了演員動畫效果，加強可協調的範圍。首先變動 XMG 描述檔的設定規則，如圖 5-19 所示，將 path 這個標籤功能性加強，當指定給 path 的值只有一組座標，XMG Player 視為不移動的演員，不過當一次給定多個座標，代表這個演員在演出時有移動的效果，播放器在播放的時候，先統計總共有幾個座標，設定一個計時器(timer)，每隔一小斷時間變更演員的位置，然後重畫整個畫面，以達到移動的效果。

```
- <Story>
- <ActorInfo>
  <actor type="IMAGE" name="Undefined_0_Actor000" file="@Res003.JPG" frames="1" path="84,14,90,17,93,19,97,23,99,27,104,31,110,37,116,39,121,43" />
  <actor type="IMAGE" name="Undefined_0_Actor001" file="@Res004.JPG" frames="1" path="45,26" />
  <actor type="IMAGE" name="Undefined_0_Actor002" file="@Res009.JPG" frames="1" path="8,56" />
  <actor type="IMAGE" name="Undefined_0_Actor003" file="@Res001.JPG" frames="1" path="42,97" />
  <actor type="IMAGE" name="Undefined_0_Actor004" file="@Res005.JPG" frames="1" path="122,87" />
  <actor type="IMAGE" name="Sound_0_Actor005" sound="@Res039.mp3" sndloop="1" file="@Res007.JPG" frames="1" path="97,199" />
  <actor type="IMAGE" name="Sound_0_Actor006" sound="@Res043.mp3" sndloop="1" file="@Res042.JPG" frames="1" path="122,87" />
  <actor type="IMAGE" name="Sound_0_Actor007" sound="@Res045.mp3" sndloop="1" file="@Res044.JPG" frames="1" path="42,97" />
  <actor type="IMAGE" name="Undefined_1_Actor" file="@Res008.jpg" frames="1" path="8,37" />
  <actor type="IMAGE" name="Undefined_1_Actor000" file="@Res018.JPG" frames="1" path="113,95" />
  <actor type="IMAGE" name="Undefined_1_Actor001" file="@Res002.JPG" frames="1" path="219,149" />
  <actor type="IMAGE" name="Undefined_1_Actor002" file="@Res011.JPG" frames="1" path="102,229" />
  <actor type="IMAGE" name="Undefined_1_Actor003" file="@Res012.JPG" frames="1" path="151,161" />
  <actor type="IMAGE" name="Undefined_1_Actor004" file="@Res021.JPG" frames="1" path="198,235" />
  <actor type="IMAGE" name="Undefined_1_Actor005" file="@Res029.JPG" frames="1" path="11,238" />
</ActorInfo>
- <Scenario>
  - <event name="prelude" vscrollbar="false" hscrollbar="false">
    <animate name="BackgroundImage_0" action="SHOW" />
    <animate name="Undefined_0_Actor" action="SHOW" />
    <animate name="Undefined_0_Actor000" action="SHOW" />
    <animate name="Undefined_0_Actor001" action="SHOW" />
    <animate name="Undefined_0_Actor002" action="SHOW" />
    <animate name="Undefined_0_Actor003" action="SHOW" />
    <animate name="Undefined_0_Actor004" action="SHOW" />
    <animate name="Sound_0_Actor005" action="SHOW" />
  </event>
  - <event name="Undefined_0_Actor003" vscrollbar="false" hscrollbar="false">
    <animate name="BackgroundImage_0" action="SHOW" />
    <animate name="Undefined_0_Actor" action="SHOW" />
    <animate name="Undefined_0_Actor000" action="SHOW" />
    <animate name="Undefined_0_Actor001" action="SHOW" />
    <animate name="Undefined_0_Actor002" action="SHOW" />
    <animate name="Undefined_0_Actor003" action="SHOW" />
    <animate name="Undefined_0_Actor004" action="SHOW" />
    <animate name="Sound_0_Actor005" action="SHOW" />
    <animate name="Sound_0_Actor007" action="SHOW" />
  </event>
</Scenario>
```



圖 5-19 變更 XMG 描述檔格式

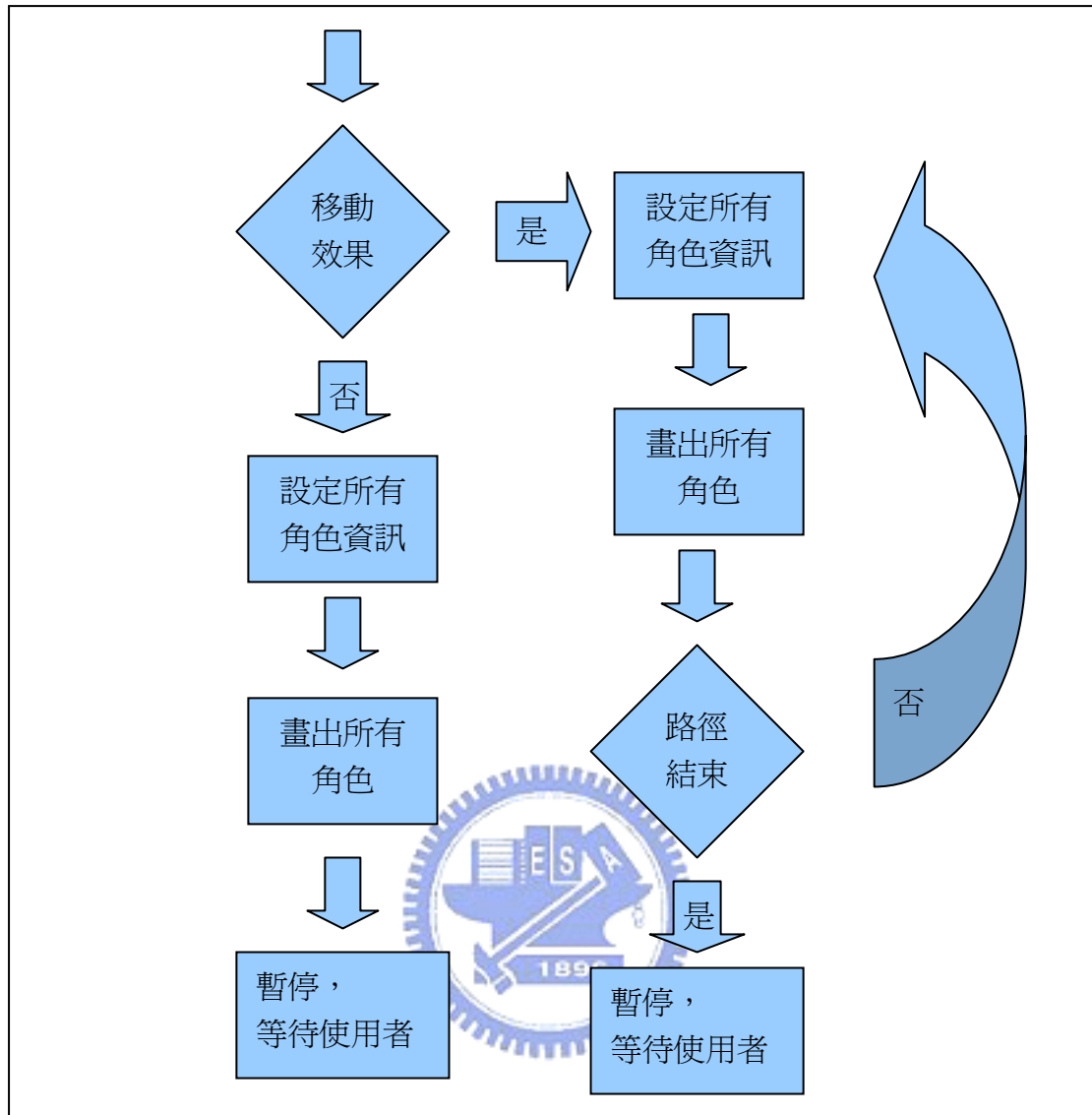


圖 5- 20 動畫演員繪圖流程圖

動畫演員的繪圖流程圖為圖 5- 20 所示，先偵測是否有移動的效果，如果沒有，將所有演員的資訊送給繪圖引擎，將角色畫出來，進入等待使用者互動或停止播放器，如果此頁面有動畫效果，先畫出所有角色初始位置，詢問路徑是否結束，如果還沒結束，會先根據路徑更新下一個的座標，在通知繪圖引擎畫出畫面，一直持續到路徑最後一個座標後才會進入等待使用者的狀態。如果同一個頁面有多個可移動的角色，系統會以路徑最長的為基準，提早結束路徑的演員會持續待在最後路徑結束的位置，不需要每一個角色路徑個數都一樣，增加教材設計的彈性。

5.4 結構完整的協調

綜合前面幾個小節中協調的方法，整個協調步驟如圖 5- 21 所示，分為解

譯描述檔案(Parsing Script File)與改編協調(Content Adaptation)兩個主要的步驟，系統輸入多媒體內容的描述檔案，經過第一個步驟解譯描述檔案的內容，接下來根據所得的演員資訊實際進入協調步驟，等協調完成後輸出最後的結果。

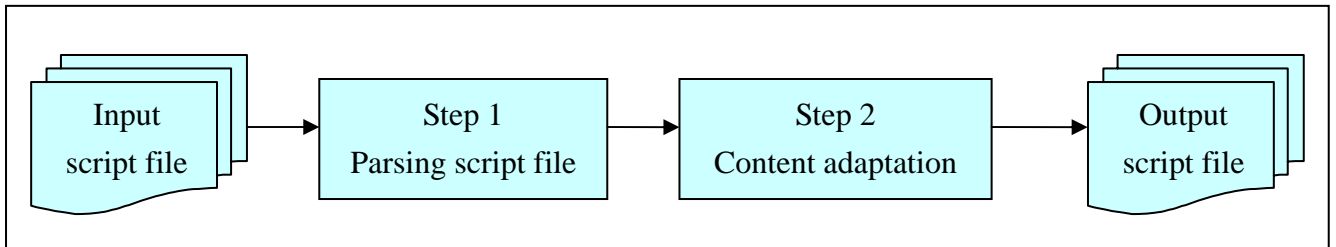


圖 5- 21 協調兩個主要的步驟

當解譯多媒體內容描述檔完成後，除了演員資訊外，頁面間的連結與角色的跳躍目標也應記錄清楚，圖 5- 22 為系統中的劇情管理員，使用 Vector-list 紀錄當某一位演員被觸發時，會跳躍到其他頁面的資訊，在協調過程中，提供給系統使用。

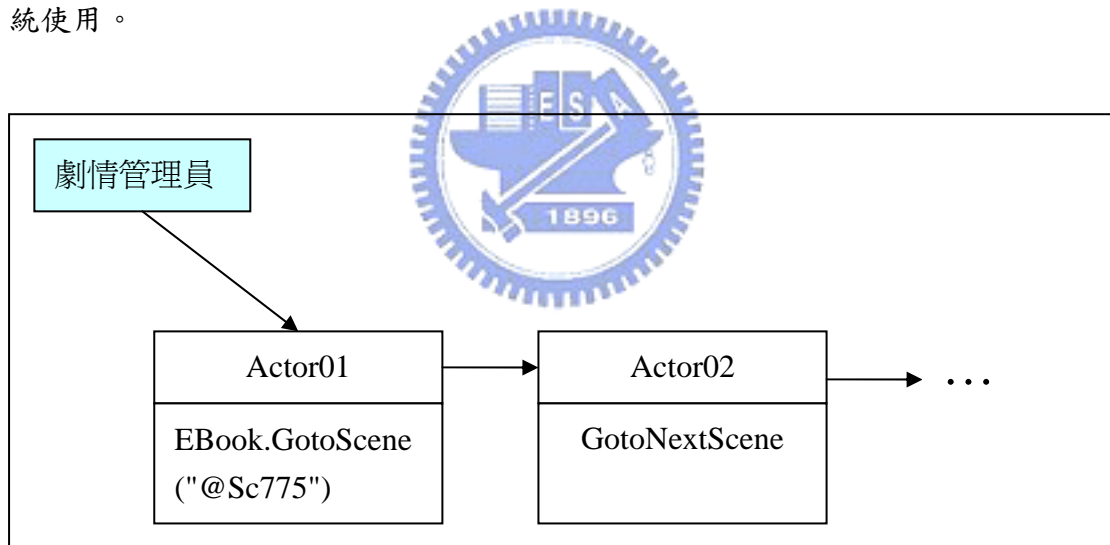


圖 5- 22 劇情管理員

對於協調步驟中會使用到的分頁方式，我們先定義分頁的規則讓系統能根據這個規則判斷是否需要啟動分頁功能，圖 5- 23 為本協調系統定義的分頁規則。Case1 是說明如果文字在協調過程中被偵測到覆蓋到其他的物件。Case2 是表示文字在協調過程中被偵測超出畫面的邊界。Case3 是說明影像與其他多媒體物件在同一個頁面。Case4 說明當兩個或兩個以上的影像在同一個畫面顯示時，如果發生這些情形，系統都會以新增頁面的方式來輔助協調。

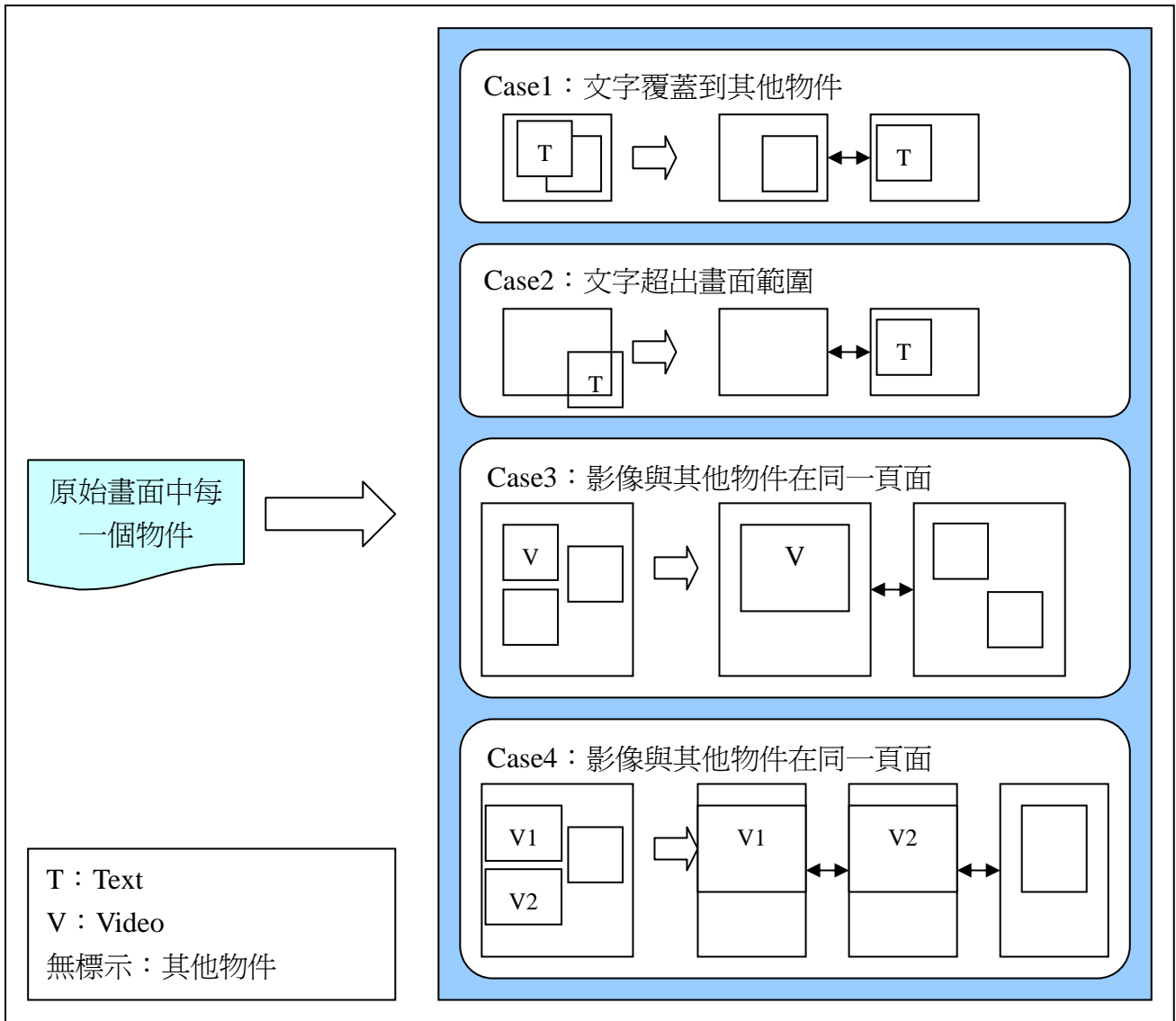


圖 5- 23 協調系統的分頁規則

在教材結構協調中的解譯階段，步驟如下虛擬碼所示，先將描述檔中的演員資訊與連結的關係擷取出來，再來根據第一步驟中擷取出的資訊，使用圖 5- 23 中的分頁規則來判斷每一個頁面需要幾個分頁來呈現，等前面兩個步驟完成後，接下來針對各頁面之間的連結以及多媒體物件的跳躍功能做處理，將結果完整記錄在圖 5- 22 的劇情管理員，重複這三個步驟，直到每一個頁面都被處理完成。

```

bool struct_adaptation()
{
    //Step1: 解譯多媒體的描述檔案，獲得演員與連結的資訊
    //Step2: 根據 Step1 完成後所得的所有資訊，使用圖 5- 23 中的分頁規則，判斷協調系統應該使用多少分頁來完成協調
    //Step3: 根據 Step1 與 Step2 的結果，將每個物件與頁面連結的資訊使用圖 5- 22 中的完整劇情管理員記錄下來
    //Step4: 重複 Step1~3，直到所有頁面都解譯完成
}

```

圖 5- 24 為簡略的示意圖，利用上述的分頁功能的步驟，判斷各頁面會有的分頁發生以及分頁的數量，將這些頁面新增完成後，再把頁面中演員和其他頁面的連結功能修正完成，確保每一個連結不會被遺漏，另外如果連結的目標已發生分頁，應修正連結的目標至分頁的第一個頁面。

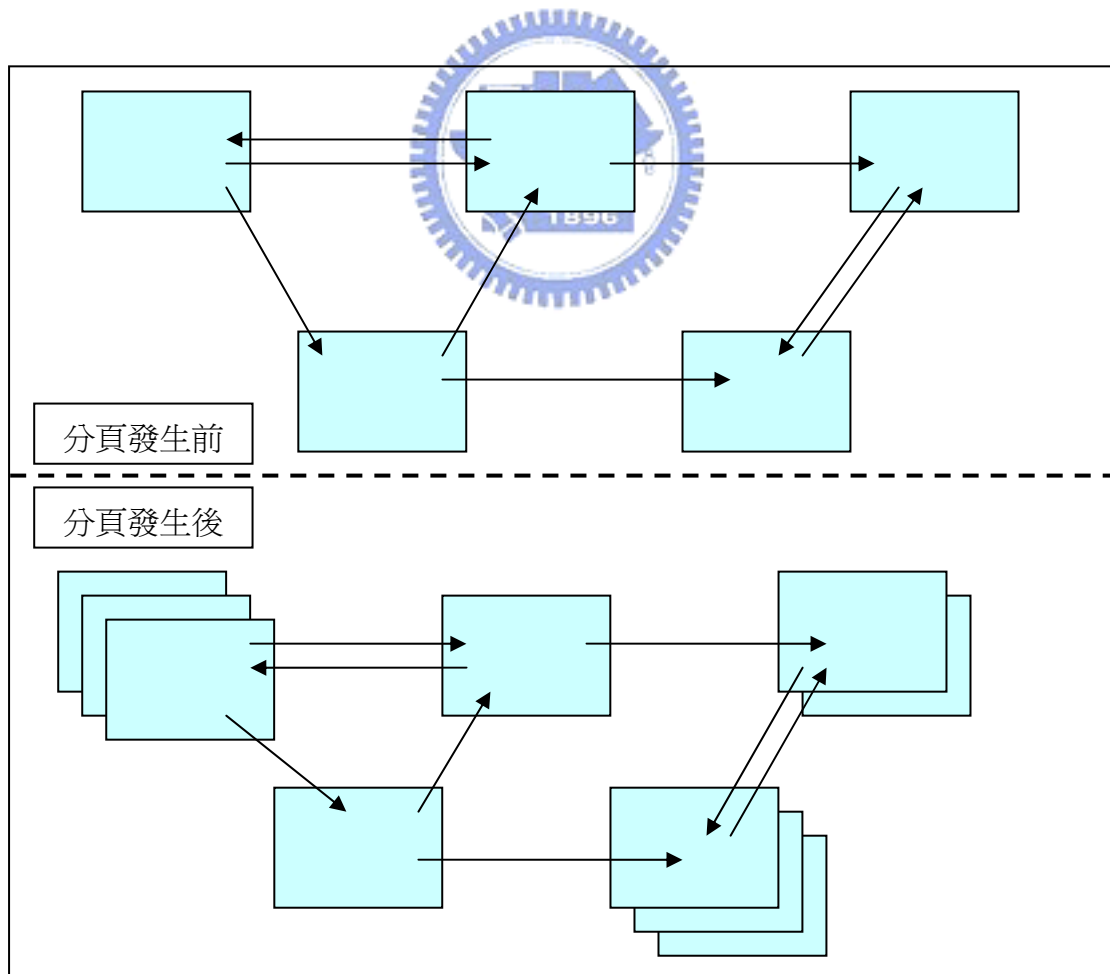


圖 5- 24 發生分頁後的情形

由前述的解譯與分頁步驟完成後，已將各個多媒體物件放置於應該放置的頁面，也因為之前的步驟，頁面間的連結與多媒體物件觸發的連結功能修正完成，接下來就是對於各個頁面做協調的步驟。根據 5.2 小節中對於文字、圖片、影像的協調步驟，完成針對 PDA 環境所適合的呈現方式。

以下我們舉一個交通大學入口網站為例子去驗證結構協調的方式，圖 5- 25 為原始的網頁內容，一般的 PDA 畫面並不能完整的顯示，所以協調系統會根據前面提到的第一步驟去判斷分頁的數量，結果如圖 5- 26 中所示，對於每一個原始頁面，找出需要多少分頁數量去呈現。



圖 5- 25 原始的網頁內容

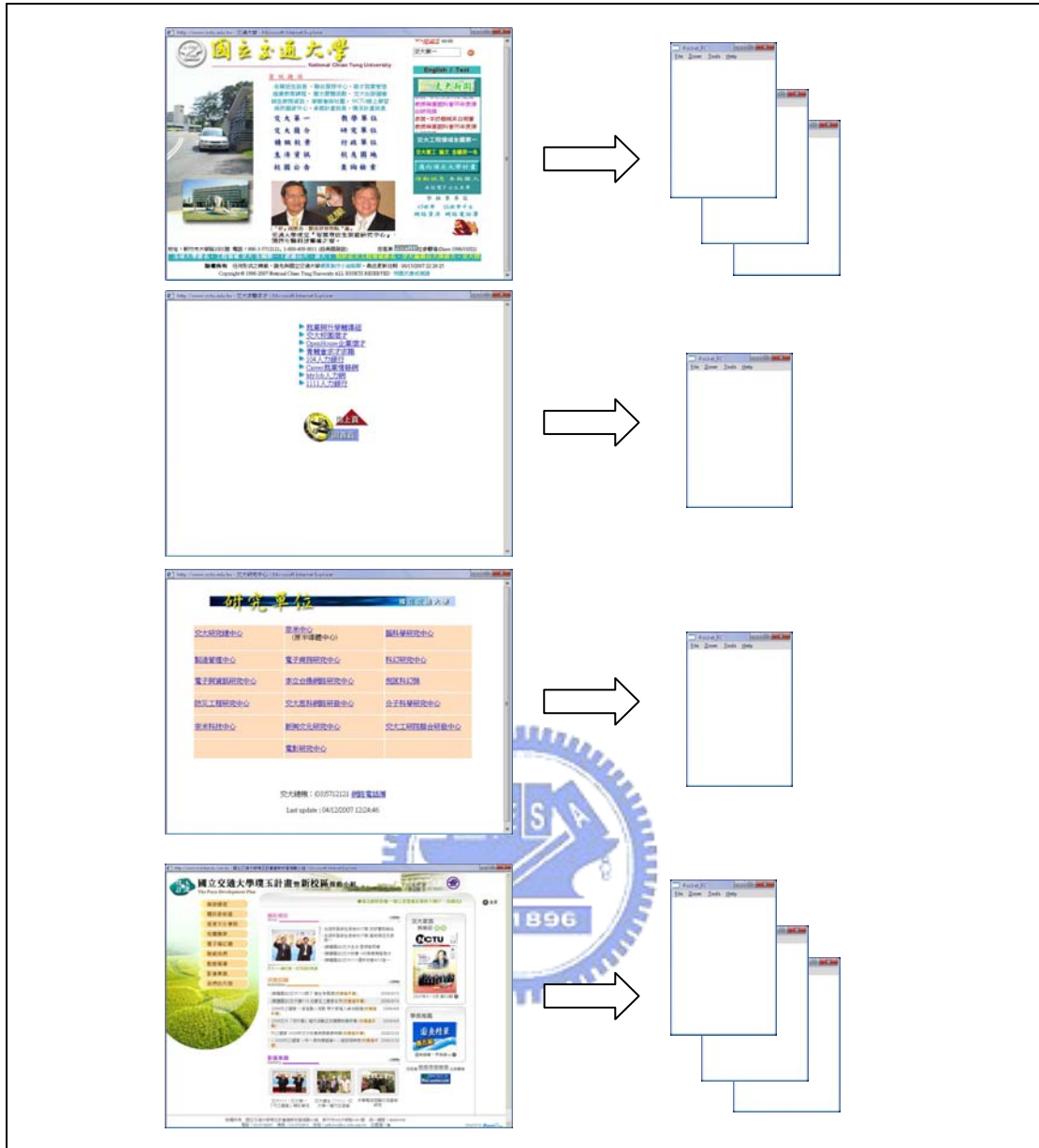


圖 5-26 每一個頁面所需要的分頁數量

根據多媒體描述檔案中所提供的資訊與分頁判斷規則，協調系統將每一個多媒體物件放置於所應該存在的頁面，其中包括已被協調系統分割的多個頁面，系統首先會將被分割的頁面之間做串接，再來依據圖 5-22 的劇情管理員提供的資訊，將每一個演員的跳躍到其他頁面的功能補上，如果跳躍的目標為單一頁面，則實做其連結即可，如果跳躍的目標為被協調系統切割的分頁群，則查詢劇情管理員找出分頁的第一頁面，將連結的目標指向第一頁面。圖 5-27 表示協調系統將每一個頁面中的每一個有跳躍功能的物件，其跳躍的目標確實的保留或者修正其目標至分頁中的第一個頁面。

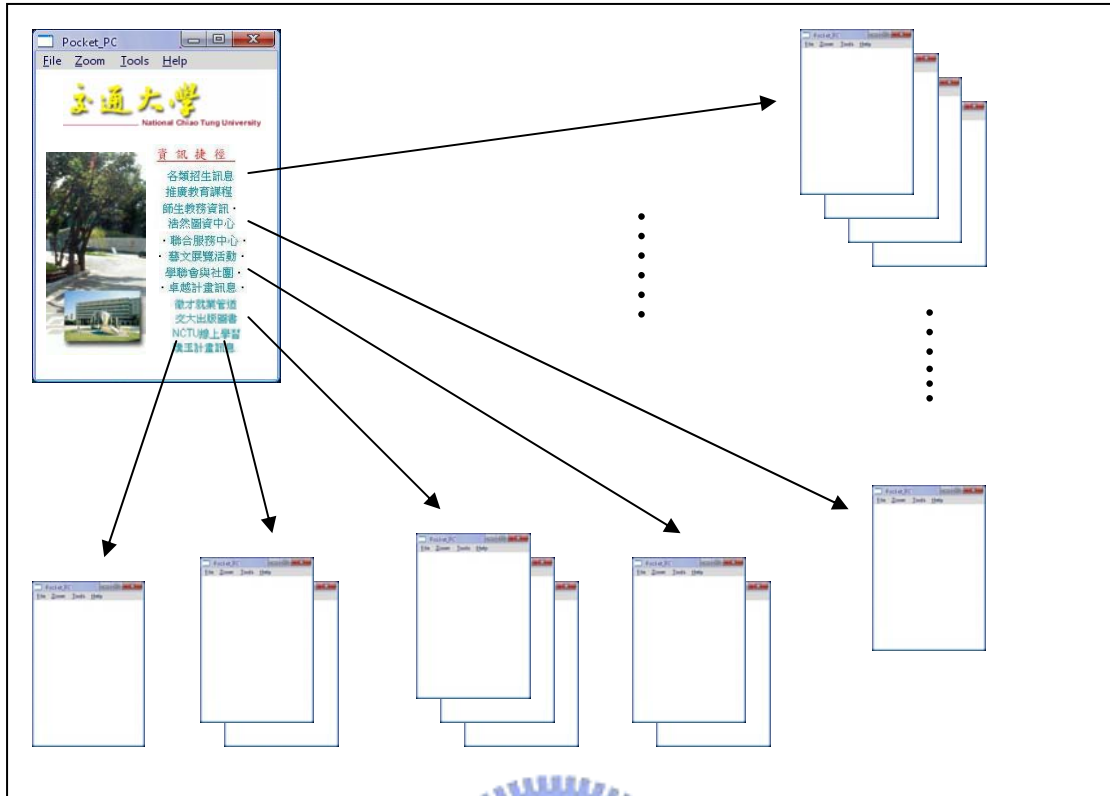


圖 5-27 將頁面中物件的連結目標確實完成

反覆執行上述的步驟，直到所有的頁面中所有的多媒體物件跳躍功能被保留與修正，接下來對於每一個頁面執行針對 PDA 畫面的協調動作，利用 5.2 小節中文字、圖片、影像的協調方式，反覆執行直到每一個頁面都協調至適合的呈現方式，如圖 5-28 所示。

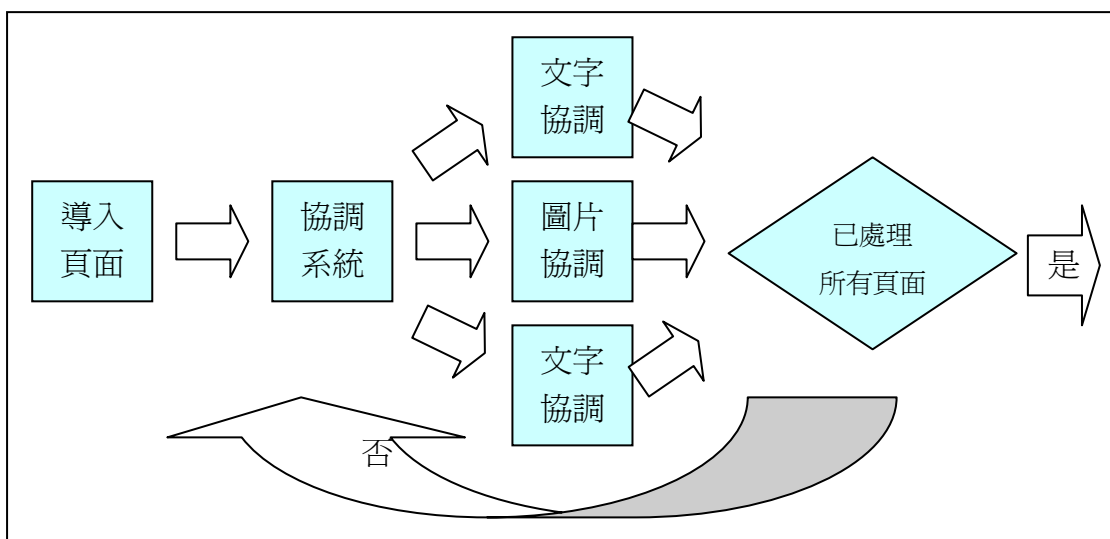


圖 5-28 協調每一個頁面

同樣的以圖 5- 25 的交通大學入口網站為例，經過上述這兩個主要步驟：解譯與處理連結步驟、頁面協調步驟，完成後其結果應該會如圖 5- 29 的樣子，圖中列了幾個多媒體物件跳躍至其他頁面的例子，除了新增頁面之間的連結狀況外，每一個頁面中跳躍到其他頁面的連結也會因為協調系統的分頁功能做改變，如圖中右半部中間的三個分頁群，會將目標修正為連接至分頁的第一個分頁，讓使用者從第一個頁面循序檢視，如果多媒體物件跳躍的目標僅僅使用一個分頁就可以呈現，就直接根據原來從描述檔所得的資訊，指向原本的目標即可。



圖 5- 29 交通大學網頁協調後結果

5.5 應用於其他種類之多媒體內容

根據前面幾個小節描述的文字協調、圖片協調、影像協調步驟，完成的結果圖片部分必定不會超出畫面的邊界，文字部分的字型顯示以讓使用者可以清楚地閱讀的基礎下協調，如果發生超出邊界或覆蓋的情形，便以新增分頁的方法解決，不影響閱讀的效果，影像部分則是綜合文字和圖片的協調方式，把影像放置第一頁面，其餘的物件再做一次協調，直到所有物件都協調完成。這樣的協調步驟可解決大部分的問題，不過目前的協調對象只針對多媒體製作工具編輯手所產生的多媒體內容，其中的原因是內容描述檔的限制，目前的協調系統無法解譯其他種類之多媒體內容描述檔，如 Flash，以下會提出可能達成協調 Flash 的原因。

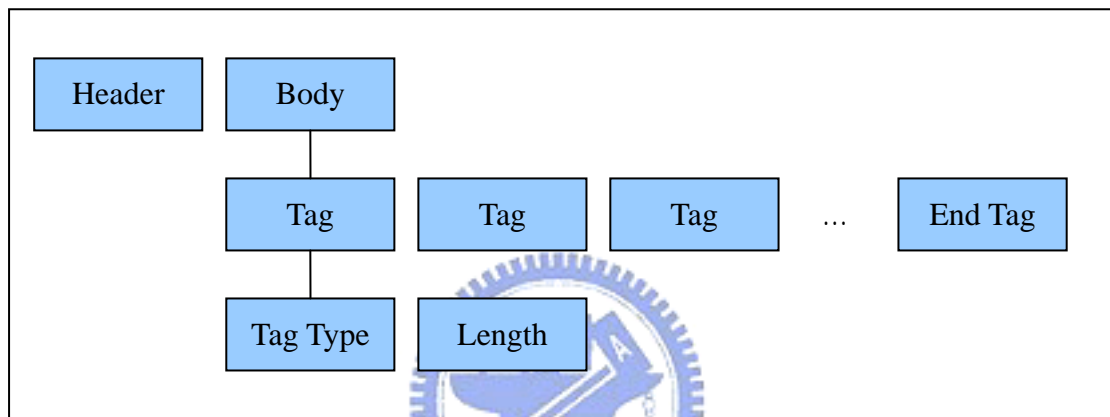


圖 5-30 Flash 簡略檔案結構

圖 5-30[22]是 Flash 簡略的檔案結構，一個檔案由一個標頭(Header)和主體(Body)組成，由許多的標籤(Tag)組成主體，一個標籤裡的內容又由標籤類型(Tag Type)和標籤長度(Tag Length)組成，標籤分為兩種：一種為定義標籤(Definition Tag)，另一種為控制標籤(Control Tag)，定義標籤主要用來定義所有物件的資訊，如文字、聲音、影像、背景...等等資訊，控制標籤定義所有換幕、互動、流程...等等資訊，這就是大略的資料結構。在此提出 Flash 的原因是，如果能在 Flash 來源檔案和本研究的協調系統之間建立一解譯和對應的機制，將定義標籤的內容對應至本系統中的自訂角色資料結構，控制標籤對應到本系統的劇情管理員，在根據本研究中的協調步驟，理應能將 PC 版本的 Flash 檔案轉換協調至 PDA 能播放的版本，依照此方式，更進一步延伸到其他類型之多媒體內容製作工具，應當能達到相同的效果。

六、實作範例

本章將使用範例來呈現並檢閱系統功能，並利用圖片的畫面來介紹系統的運作流程，依照運作流程主要介紹有三個部分：

- 管理者 web 操作介面
- PDA 應用端播放器呈現畫面
- 協調前後的畫面比較

6.1 管理者 web 操作介面

- 連線至管理者入口網頁，使用管理者的帳號登入。

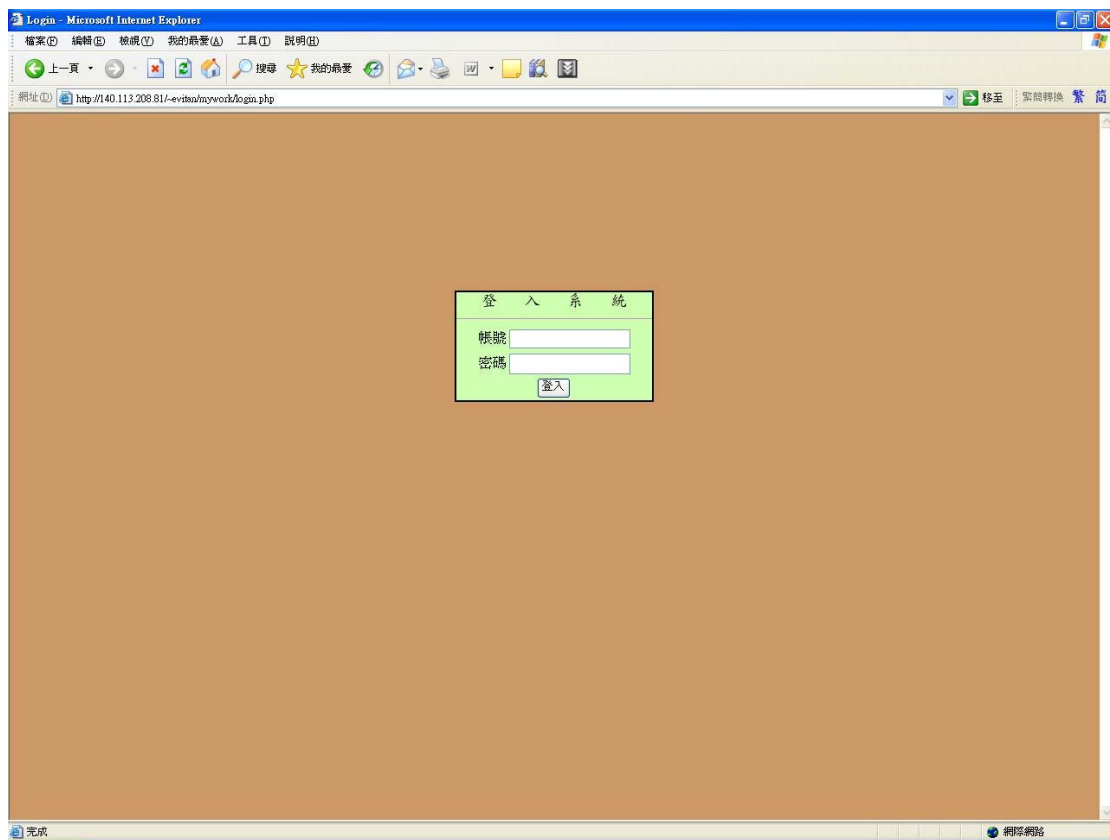


圖 6-1 管理者登入畫面

- 登入畫面後自動列出所有清單，管理者選擇需要的教材編號進行協調，右方的表格中會列出管理者選擇的編號以供確認。

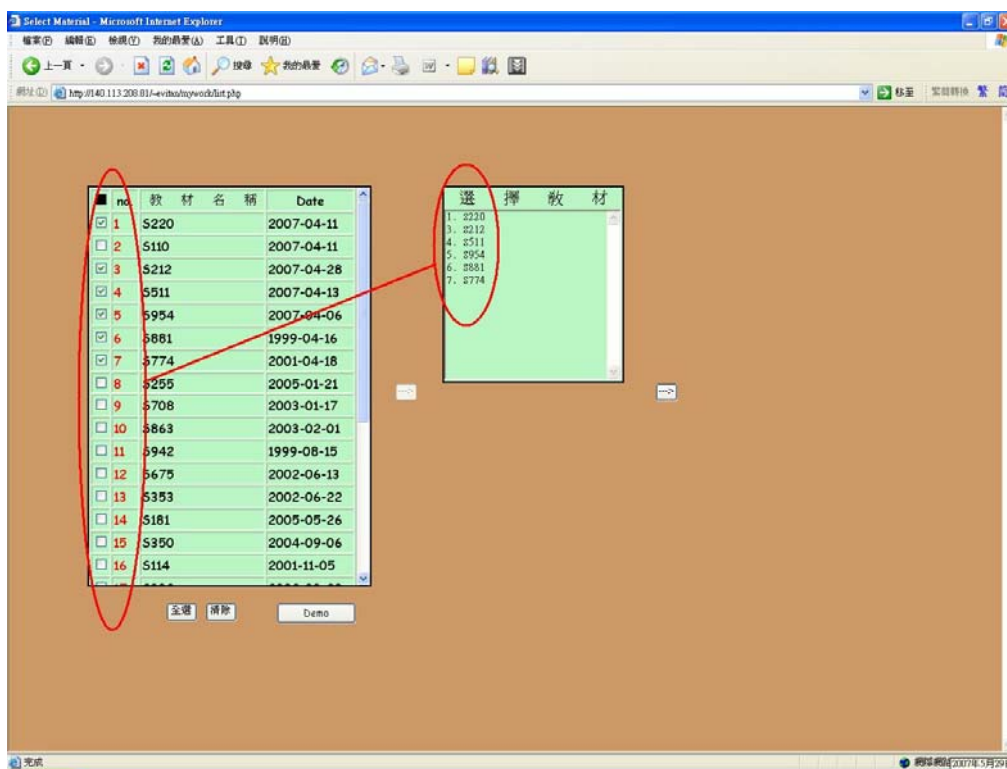
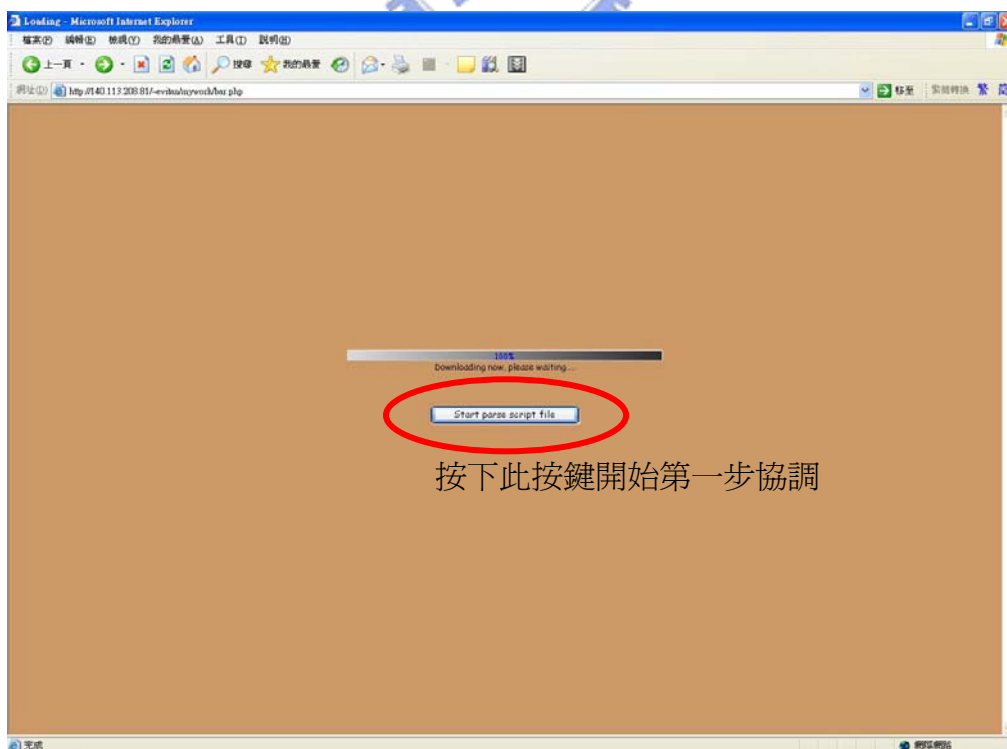


圖 6-2 選擇教材清單畫面

- 根據所選擇的教材編號，向內容資料庫要求下載並要求管理者等待。



按下此按鈕開始第一步協調

圖 6-3 要求下載教材畫面

- 系統已完成第一步的協調動作，等待使用者檢視，進行個人喜好部分微調。

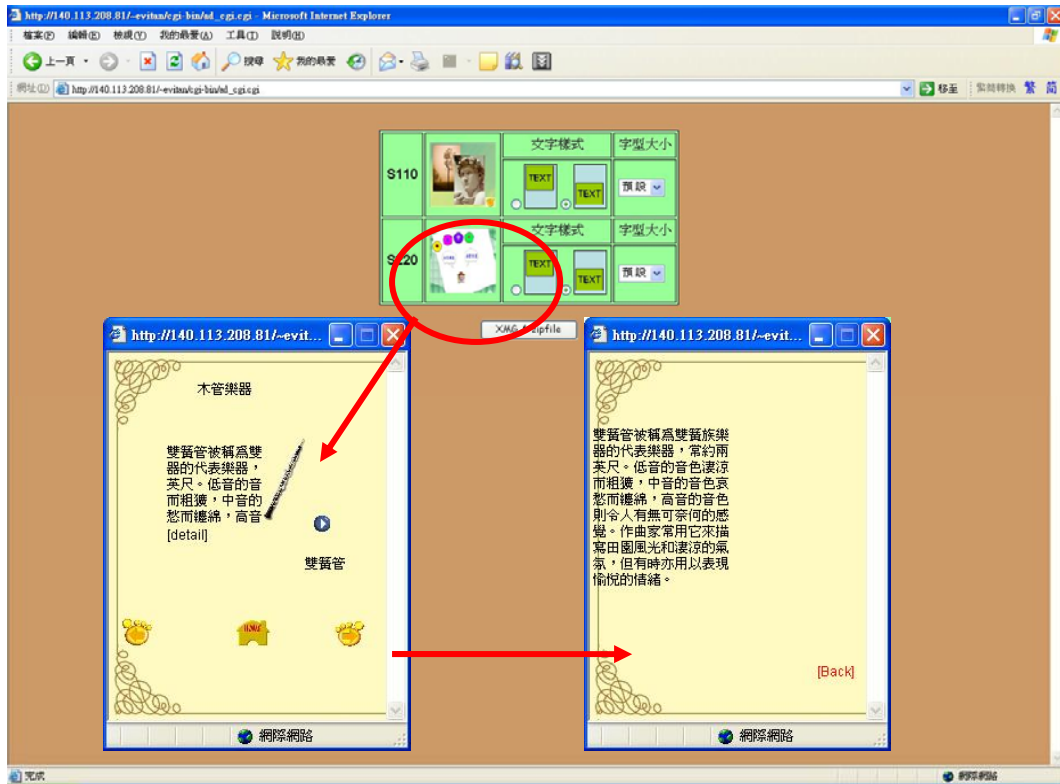


圖 6-4 初步協調結果並檢閱

- 最終的協調輸出成新的版本，並且壓縮完成，準備上傳至內容資料庫。

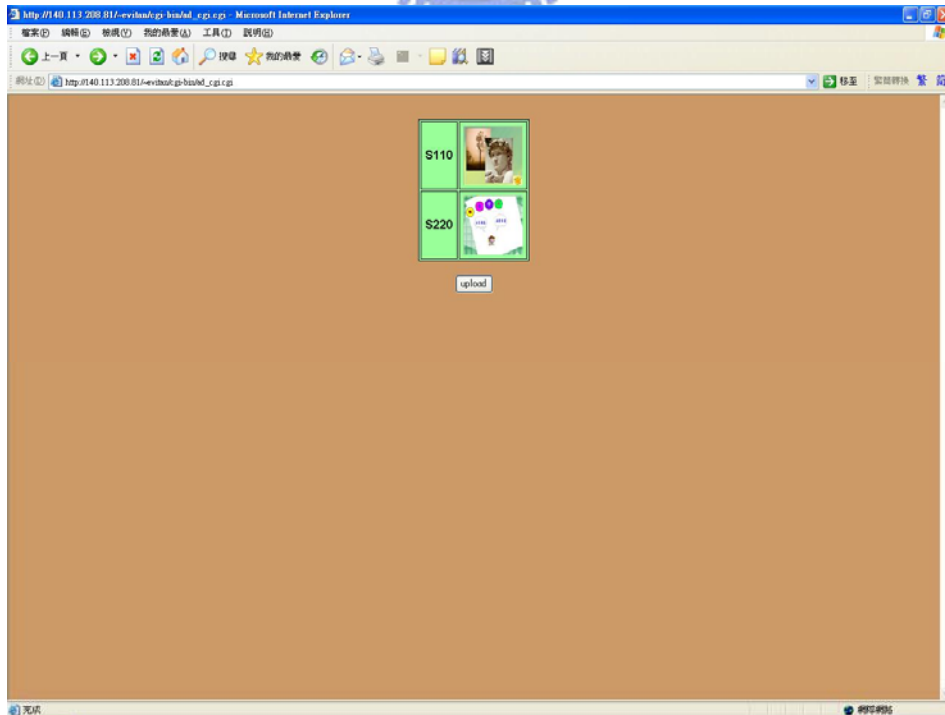


圖 6-5 列出準備上傳的教材編號與預覽圖

- 確定要上傳的編號無誤，執行上傳功能。

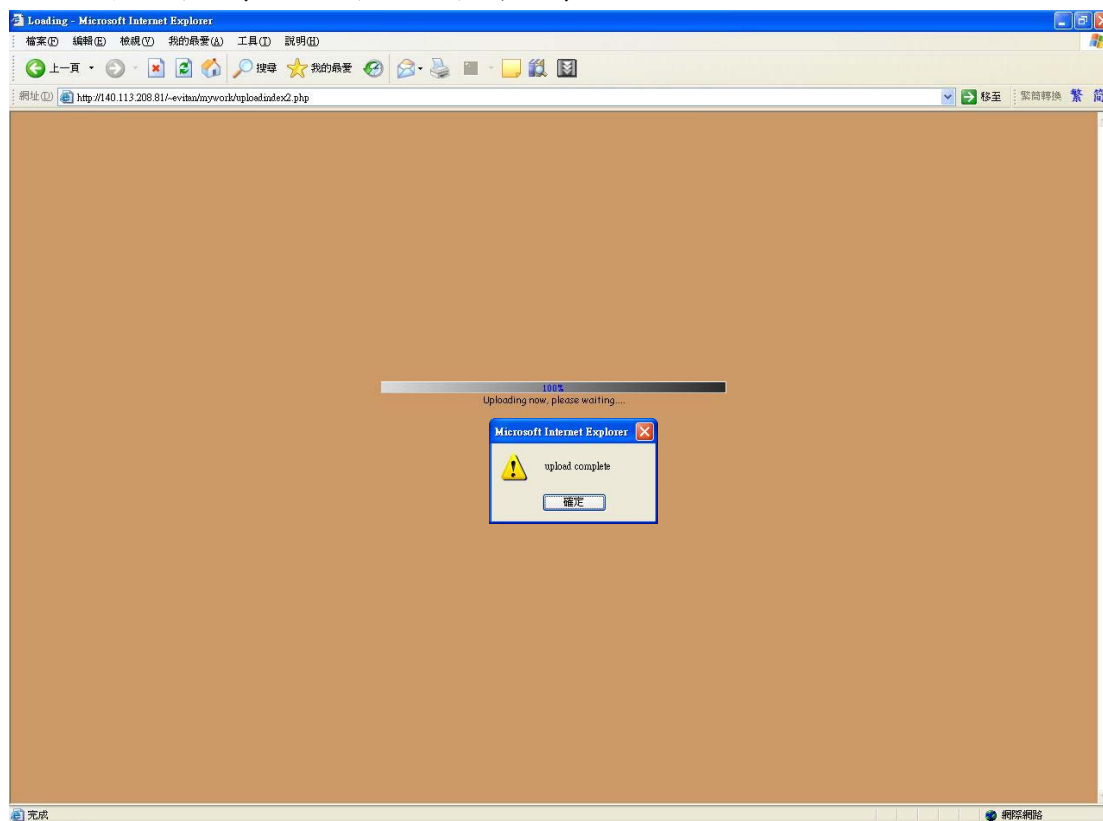


圖 6-6 上傳成功畫面

6.2 PDA 應用端播放器呈現畫面

- 開啟播放器自動連線至登入畫面，輸入有效的帳號、密碼後進入系統。

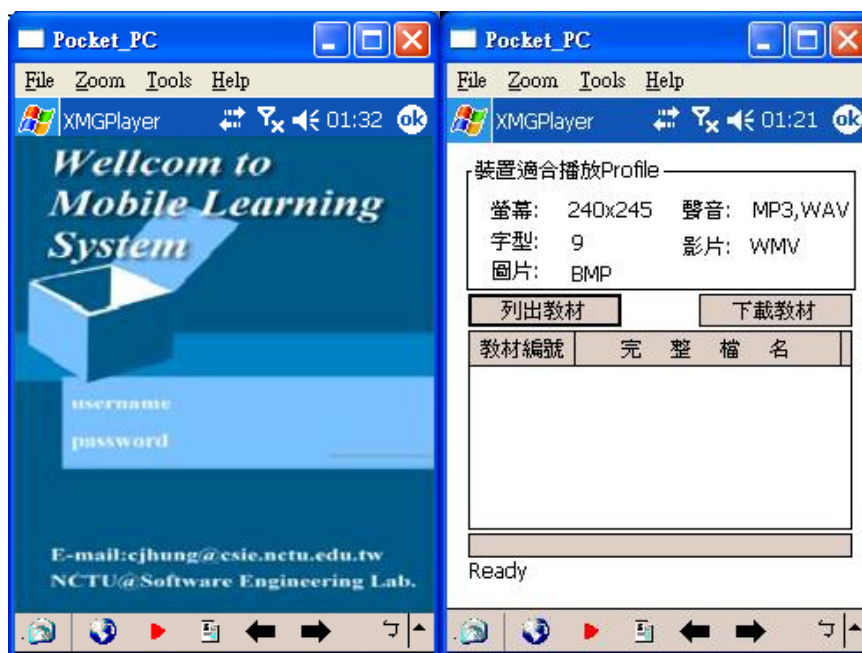


圖 6-7 登入及啓始畫面

- 點選列出教材下方列出現有教材清單，選擇一份教材下載。

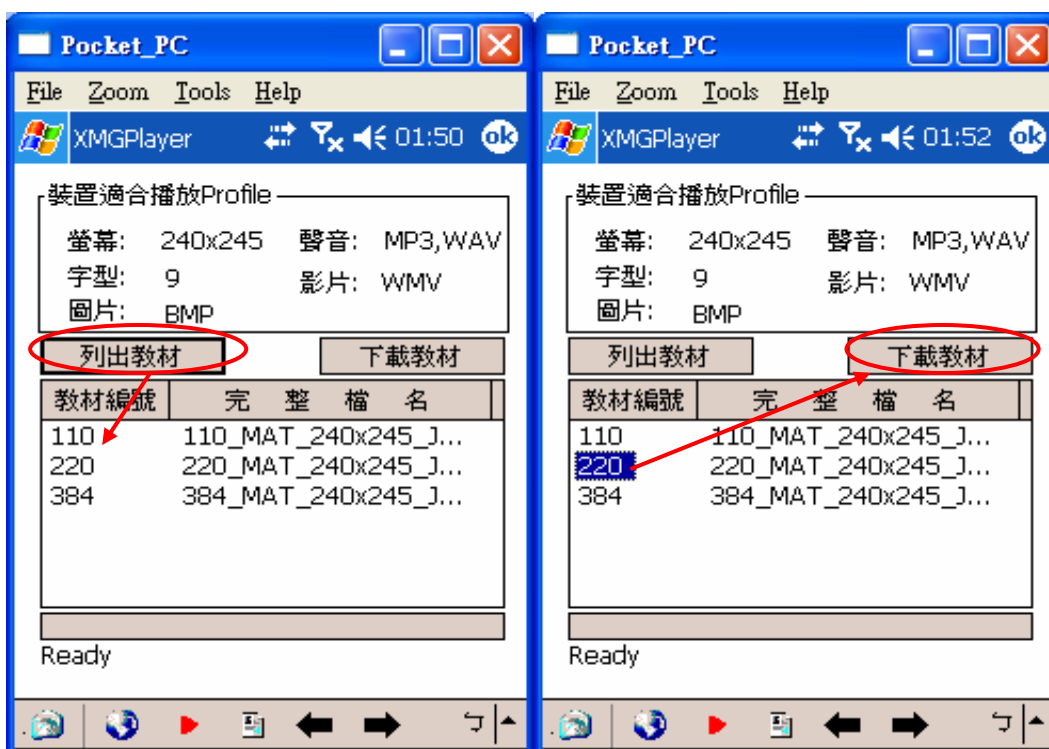


圖 6-8 教材清單與選擇下載

- 下載時下方進度條顯示進度，下載完成後自動解壓縮，然後播放教材。



圖 6-9 下載過程與等播放過程

- 點選播放按鈕自動從播放器清單第一筆開始播放，或者可點選另出清單，選擇一份教材播放。

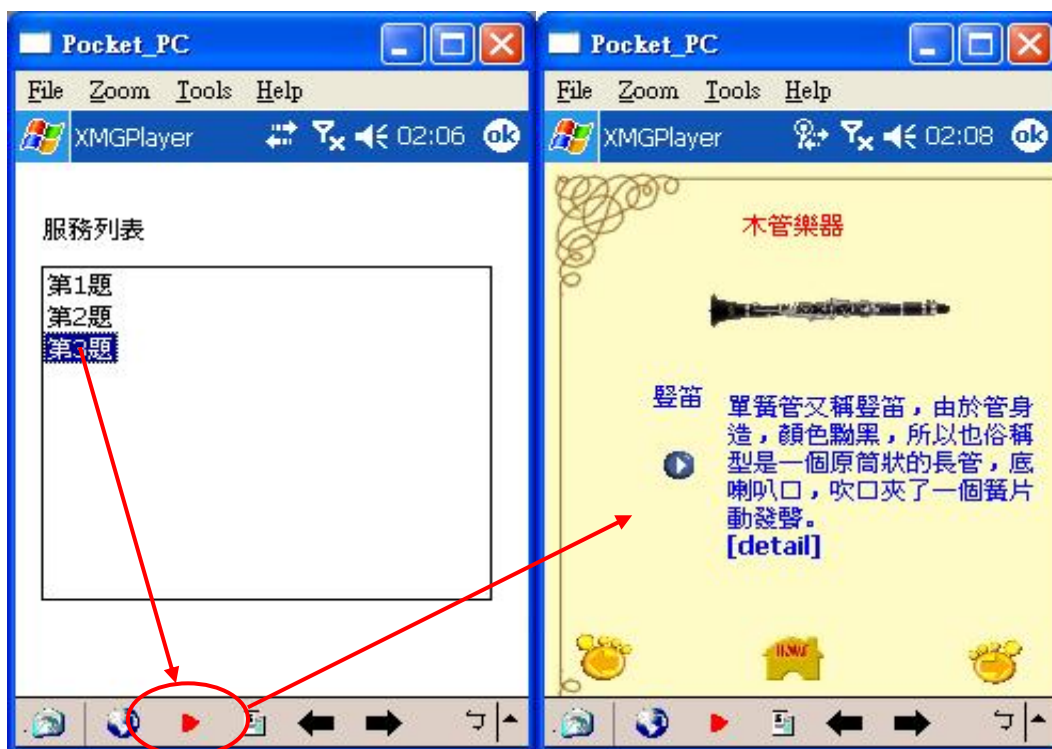


圖 6-10 選取教材播放

- 除清單模式外，還可利用播放下一份教材的按鈕，循序播放。



圖 6-11 循序播放教材

6.3 協調前後的畫面比較

利用兩個範例來檢視教材協調前後的比較：

(a). 已製作完成教材

- 教材範例一原始畫面(640*480)



圖 6- 12 已製作完成教材範例一畫面

- PDA PDA 顯示協調完成畫面(240*320)



圖 6- 13 範例一協調後畫面

- 教材範例二原始畫面(640*480)



圖 6- 14 已製作完成教材範例二畫面

- PDA 顯示協調完成畫面(240*320)



圖 6- 15 範例二協調後畫面

(b). 影像的協調範例

- 使用影像的原始教材(640*480)



今天的世界沒有過去那麼多真空，地球上94%的人都知道可口可樂為何物。這意味著真空已被徹底打敗。真空的天敵——阿薩·格裡格斯·坎德勒，可口可樂公司的創始人，出生在150年前。他在有生之年得到了最好的生日禮物：世界各個角落都充斥著帶紅色商標的「曲線」瓶。這裡要講述的，就是一個戰勝真空的故事。

圖 6-16 原始影像教材畫面

- PDA 顯示協調完成畫面(240*320)



圖 6-17 協調影像教材完成後畫面

6.4 結構協調的範例

- 協調前畫面連結的情形



圖 6-18 原始教材連結情況

● 協調完成後連結情形

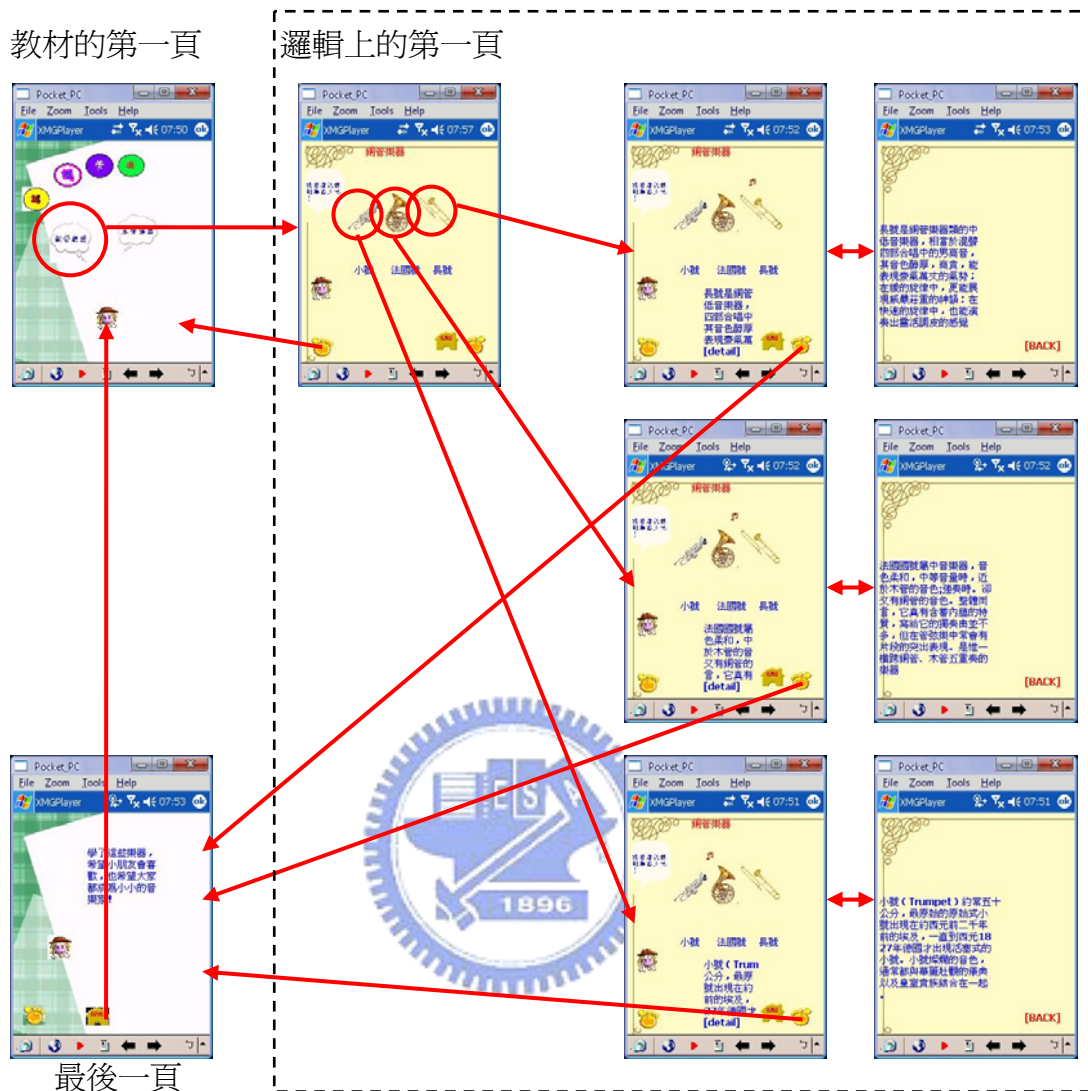


圖 6-19 結構協調的範例

圖 6-18 為原始的教材連結情形，經過協調後，如圖 6-19 中所示，將銅管樂器分割為七個頁面，當需要連結到銅管樂器此頁面時，會連結至邏輯上的第一頁，也就是最上方中間的圖，而其他的分頁向下展開，加入額外的連結，如果分頁中有跳躍到其他分頁的連結，如圖中跳躍到最後一頁的連結，則確實保留次要連結。

七、 結論

7.1 總結

本系統主要是將本實驗室先前的研究成果，以樣版是多媒體英文試題轉換協調系統為基礎，加以改良並且延伸出新的功能，套用其他的多媒體內容轉換協調系統，和前系統主要不同與改良的部分如下表 7- 1 所示：

表 7-1 與前系統相異與改良部分

來源部分 與 輸出部分	<p>(a). 在樣版式、單一頁面的英文試題的協調方法上，針對此樣版設計特定的呈現方法，雖然可達到最佳的顯示，但是一旦更改編輯時的樣版，勢必要針對新的樣版，再設計新的顯示方式，此一更動需要改寫程式核心部分，基於資料和程式分離的設計原則，發展能對於不固定呈現方式的多媒體內容的協調的方法。</p> <p>(b). 文字協調部分，除了沿用前研究對於英文文字的部分，新加入了對於中文字型的處理部分，並且在新增分頁擺放文字位置的功能中，針對背景圖片做額外的判斷，使文字能擺放置更適合的位置。</p> <p>(c). 圖片協調部分，使用單一比例的方式縮小圖片，確保圖片在協調過程中不失真，而同時使用寬度與高度的比例協調座標位置，充分利用 PDA 所能提供的有限空間。</p> <p>(d). 影像協調部分，前研究對於影像的協調僅限於樣版，如果在同一頁面同時存在影像、文字、圖片，甚至於多個影像，原有的方式則無法達到協調的目標。</p> <p>(e). 多媒體內容結構協調，由於多媒體內容多頁面的設計，再加上利用分頁的方式做協調的動作，每個頁面之間的連結多且複雜，完整保留原有的連結、加入新增頁面的連結，最重要的是修正已使用分頁手法協調的頁面之間的連結，以確保原有的多媒體內容結構的完整。</p>
播放器部分	<p>(a). 在英文試題製作時，對於角色移動出場的功能，並沒有要求，而多媒體教材的製作上，此功能就較常使用，所以在 PDA 的 XMG player 方面，新增加了這個需要的功能。</p> <p>(b). 顯示多媒體格式：PNG 格式。</p>

在轉換的過程中，對於多媒體檔案做適當地縮小，而協調的演算法除新增了頁面讓使用者操作外，並無新增任何額外的檔案，所以在空間的使用上並無增加，在傳輸時間上也不會增加，多媒體格式的協調方式，保留了靜態(Static)和

動態(dynamic)混和協調方法的研究結果。

本研究中所使用的協調演算法，是以文字、圖片、影像的類別進行畫面的協調，在一個畫面中，首先將圖片擺放上去，由於圖片的大小和座標是根據畫面做縮小的處理，所以不會有超出畫面邊界的問題發生，再擺放文字，計算文字所佔有的面積，如果發生覆蓋超出邊界的問題，利用分頁的方式將文字顯示在其他的頁面，所以文字部分兼顧可閱讀性和不超出邊界、不覆蓋其他物件這些要素，最後如果這一個頁面發現有影像格式的存在，第一個頁面便會先放置影像，其他的到下一個頁面再進行之前所提出的步驟，這樣循著步驟執行，最後會呈現一個較適合的畫面提供給使用者，所以此協調方式的正確性是存在的。依照 5.4 小節中推論，這些協調的步驟是根據多媒體內容的格式和系統所給的物件資訊所決策出來的，對於來源的多媒體描述檔並無直接的關係，所以若能在多媒體描述檔案和系統間建立新的解譯及互動式結構的對應，此協調演算法應當能適用於其他種類的多媒體內容。

本研究最重要的目的之一，就是在於保留教材的原意部分，當協調過程中，能兼顧保留教材製作者的原意和對畫面做較佳化的協調中取一個平衡點，圖 7-1 會對這部分提出說明。

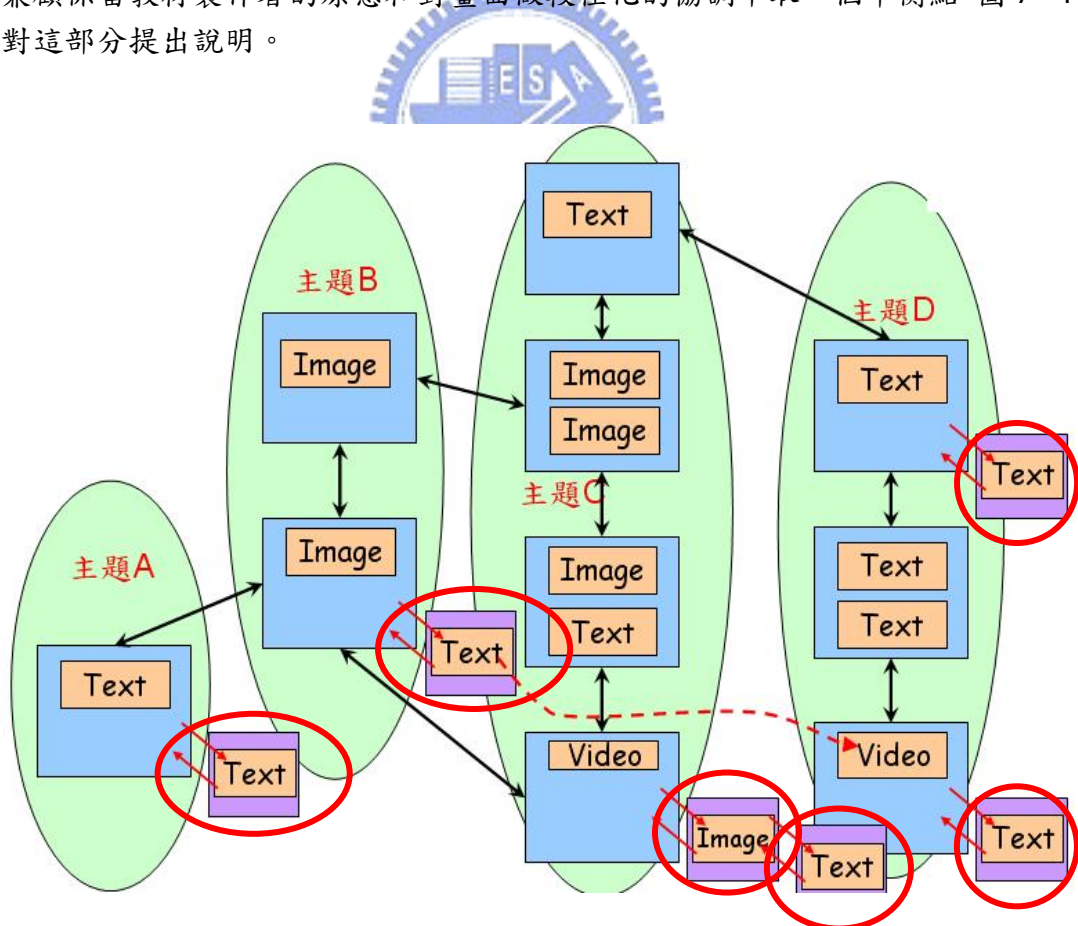


圖 7-1 協調完教材架構圖

架構圖 7- 1 中圈起來的部分是因為協調而新增的頁面，利用新的連結將它們和原本的教材流程串起來，從圖上可以看得很清楚，基本上主要的架構流程是和原本的一樣，並沒有因為協調的過程遭到變更，而分頁後連結的情形，協調系統利用分頁的方式，將一個頁面切割成幾個頁面呈現，在邏輯上，這些新增的頁面應視為同一頁，因為這幾個新增頁面結合起來才能表示原始教材中相同的意義，所以協調系統在教材連結流程中必須做調整，將連結調整到邏輯中的第一頁，才不至於讓流程混亂。



7.2 未來展望

本章將說明本研究對於未來整個系統有哪些已發現的缺點、未解決的問題或未來可發展的方向，以供參考。

- 目前教材的轉換協調只針對 PDA 單一類型，對於其他手持行動裝置，例如行動電話，並無法確定所有功能都能正常運作，所以未來可以針對這部分，將協調系統改良成能夠協調出適用於不同平台的版本，以利在不同的裝置呈現多媒體檔案。
- 由於中文字型的種類太多，本實驗僅僅只討論新細明體一種字型，但是在教材設計上，有可能需要使用多個字型來強調內容，例如中文字的教學就可能要介紹多種不同的樣式，才能讓閱讀者清楚瞭解，未來可將較為常用的加入系統考慮的範圍。
- 由於教材的設計上，有可能一份就多達數十個頁面，而在本系統中的管理者介面是使用線上及時控制的方式操作，轉換協調的時間拉長後，系統可靠性相對的就會降低，例如在無線網路的環境下，在協調過程中，無線網路發生未知的干擾，造成協調的失敗。所以如果能快速的將教材協調完成後上傳，避開長時間不定的因素，增加可靠性。未來可能可以利用分散式的方法縮短作業的時間，而在各部分流程的設計上，程序的優先順序如能夠做更仔細的安排，減少不需要浪費的時間。
- 在多媒體轉換的部分，如能找到更好的轉換引擎，減少轉換的時間。而播放器的繪圖引擎部分，雖說一般的 jpeg、bmp 能正常顯示，png 檔案類型僅能顯示單一圖層，需要多圖層堆疊的功能並不完備，如果能找到新的工具，將不足的補足和新增其他常用的多媒體格式。

參考文獻或資料

- [1] 資策會 FIND
<http://www.find.org.tw/find/home.aspx?page=news&id=1964>
- [2] Tforecasts , <http://www.etforecasts.com/>
- [3] 洪啟彰,“樣板式多媒體內容在手持行動裝置之情境及內容協調與呈現-以樣板式的多媒體英文試題為例子”,交大碩士論文,2006
- [4] Anind K. Dey , Gregory D. Abowd ,“Towards a Better Understanding of Context and Context-Awareness”,_Technical Report GIT-GVU-99-22,1999
- [5] Augusto Celentano , Ombretta Gaggi ,“Context-aware design of adaptable multimodal documents”,2004
- [6] 智勝國際科技 , <http://www.caidiy.com/a1.htm>
- [7] Hp , <http://welcome.hp.com/country/tw/zh/welcome.html>
- [8] Richard C. Leinecker and Tom Archer 原著,黃怡 編譯
Visual C++ 6 進階與程式應用篇,文魁資訊股份有限公司,2001
- [9] CxImage , Davide Pizzolato , <http://www.xdp.it/cximage.htm>
- [10] DevIL , <http://openil.sourceforge.net/>
- [11] Wai Yip Lum , Francis C.M. Lau ,“A Context-Aware Decision Engine for Content Adaptation”, IEEE Pervasive Computing,2002
- [12] Tayeb Lemlouma , Nabil Layaida ,“Context-Aware Adaptation for Mobile Devices”, In IEEE Int. Conf. on Mobile Data Management, Berkeley, CA,USA,2004
- [13] B. N. Schilit and M. M. Theimer , “Disseminating active map information to mobile hosts” , IEEE Network, vol. 8, no. 5, pp. 22-32, Sept./Oct.1994
- [14] 陳志銘¹, 李宜倫², 陳明娟², “以無線網路定位技術支援學習情境偵測之智慧型行動應與字彙學習系統”,國立政治大學圖書資訊與檔案學研究所¹, 國立花蓮教育大學學習科技研究所²

- [15] Dey, A. K., Abowd, G. D., “Toward a Better Understanding Context and Context-awareness, ” Georgia Tech GVU Technical Report , GIT-GVU-99-22, 1999
- [16] Narendra Shaha, Ashish Desai & Manish Parashar ,“Multimedia Content Adaptation for QoS Management over Heterogeneous Networks”, proc of International Conference on Internet Computing 2001,USA,2001
- [17] Zhijun Lei ,Nicolas D. Georganas ,“Context-based Media Adaptation in Pervasive Computing”, in Proc. of Canadian Conference on Electrical and Computer Engineering,2001
- [18] 江書瑩，“互動式多媒體的視覺化劇情編輯機制應用於多媒體試題樣板套用系統的實作”,交大碩士論文,2005
- [19] The Code Project ,
<http://www.codeproject.com/bitmap/cximage.asp>
- [20] phpMyAdmin ,
http://www.phpmyadmin.net/home_page/index.php
- [21] 中文字型 pt , px
<http://blog.sina.com.tw/verklarung/article.php?pbgid=21575&entryid=348872>
- [22] Adobe , Flash , <http://www.adobe.com/tw/>