

國立交通大學

資訊工程學系

碩士論文

互動式多媒體樣板在異質性手持行動裝置
上之呈現及其樣板轉譯器之實作



The Design and Implementation of Multimedia Template
Translators for Heterogeneous Mobile Device

研究生：王宇涵

指導教授：陳登吉 教授

中華民國九十四年七月

互動式多媒體樣板在異質性手持行動裝置
上之呈現及其樣板轉譯器之實作

The Design and Implementation of Multimedia Template
Translators for Heterogeneous Mobile Device

研究生：王宇涵 Student：Yu-Han Wang

指導教授：陳登吉 Advisor：Deng-Jyi Chen



A Thesis Submitted to
Department of Computer Science and Information Engineering
College of Electrical Engineering and Computer Science
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Master
in
Computer Science and Information Engineering
July 2005
Hsinchu, Taiwan, Republic of China.

中華民國九十四年七月

互動式多媒體樣板在異質性手持行動裝置 上之呈現及其樣板轉譯器之實作

學生：王宇涵

指導教授：陳登吉 博士

國立交通大學資訊工程學系碩士班



目前互動式多媒體日漸普及，在 PC 上看得到的應用已有許多，但手持行動裝置上的應用還不是很普遍，最常見的大概多為遊戲類，而手機上的 MMS 簡訊雖有多媒體的效果，但並無互動的能力。不過，現在手持裝置支援多媒體的能力越來越強，因此開發遊戲類以外的應用，例如：賀卡、簡訊、學習類教材或是試題等等，將會使手持行動裝置的附加應用價值大大提升。

手持行動裝置上的互動式多媒體製作方式並不容易，因此若有一個特殊的轉譯器，可以讓使用者利用現有的一套 PC 上的視覺化編輯工具來做出自己想要的內容，接著只要透過轉譯器，便能產生出適合於各式手持行動裝置的互動式多媒體，將會大大減少製作的難度。除此之外，提供樣板機制，讓使用者可以從網站上的下載喜愛的樣板，只要做些修改就能產生出屬於自己的互動式多媒體，不但可以減少編輯的步驟，更能增加互動式多媒體的吸引力。

因此本研究主要就是實作轉譯器，讓使用者只要利用 PC 上的視覺化編輯工具來編出自己想要的賀卡、簡訊、教材或試題等互動式多媒體的內容，接著就可以自動轉換成手機及 PDA 可以播放的格式，此外，還提供了適用於手機及 PDA 的樣板機制，讓使用者可以更輕易產生出適用於 PDA 和手機的互動式多媒體內容。

The Design and Implementation of Multimedia Template Translators for Heterogeneous Mobile Device

Student: *Yu-Han Wang*

Advisor: *Dr. Deng-Jyi Chen*

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

Abstract

Recently, building multimedia application software on mobile handset devices (such as PDA and cellular phone) has received considerable attentions for the release of new models. The demand of more add-on features (such as gaming, e-mailing, messaging, multimedia presentation...and so on) by mobile handset users have resulted in a new competition for mobile handset device manufacturers. To meet their new demands, “how to port (or redesign) multimedia applications on PC environment to the mobile handset device environment” becomes a common focus. Currently, several simple Java games have successfully been implemented on mobile handset devices. So does the MMS message for cellular phone. These multimedia applications basically are re-created under the mobile handset device environment using such as J2ME environment. In general, it’s not an easy task to reproduce these multimedia applications for mobile handset devices using the traditional re-programming approach.

In this thesis research, we propose Multimedia Template Translators for the creation of multimedia applications for mobile handset devices. Specifically, a C++ based Multimedia Template Translator for PDA and a Java based Multimedia Template Translator for cellular phone with supporting MIDP 2.0 features are designed and implemented. A website that maintains multimedia presentation templates created using a visual authoring tool is designed and implemented to demonstrate resulting presentation. Also, a template replacing mechanism for cellular phone and pocket pc is designed and implemented respectively for mobile users to enjoy the template based authoring.

A prototype system has implemented and is used to demonstrate the applicability and feasibility of the proposed Multimedia Template Translators. Although the proposed approach can easily produce the multimedia application for mobile handset devices, it has to be used with a specific visual authoring system for the multimedia presentation template creation.

誌謝

本論文承蒙指導教授陳登吉老師的耐心指導與教誨下，得以順利完成，在此致上對老師無限的感謝。

此外，我要感謝所有曾經在課業上教導我、幫助我的師長、朋友、同學，以及軟體工程實驗室中的每一位成員，不論是已畢業的學長們、學弟妹們或是同窗好友，因為有你們的陪伴和鼓勵，讓我的研究所生活更多彩多姿。

最後，我要感謝我最愛的家人—爺爺、爸爸、媽媽以及兩位可愛的妹妹，因為你們的鼓勵和支持，讓我得以順利完成論文的研究，成就今日的我，謝謝。



目錄

摘要.....	iii
Abstract.....	iv
誌謝.....	v
目錄.....	vi
表目錄.....	viii
圖目錄.....	ix
一、 緒論.....	1
1.1 手持行動裝置之應用價值的探討.....	1
1.2 互動式多媒體之應用.....	1
1.3 研究動機與目標.....	3
1.4 研究方法、步驟.....	4
1.5 研究範圍.....	5
1.6 章節概要.....	6
二、 相關研究.....	7
2.1 手機應用程式開發之相關研究.....	7
2.1.1 為什麼要選用 Java.....	7
2.1.2 J2ME 介紹.....	8
2.1.3 MIDP 介紹.....	9
2.2 PDA 應用程式開發之相關研究.....	11
2.2.1 PDA 應用程式設計語言的選擇.....	11
三、 系統功能需求分析.....	14
3.1 轉譯器之功能需求.....	14
3.1.1 轉譯給 PDA.....	15
3.1.2 轉譯給手機.....	15
3.1.3 上傳機制.....	16
3.2 樣板機制之功能需求.....	17
3.2.1 PDA 之樣板機制.....	18
3.2.2 手機之樣板機制.....	18
四、 系統架構.....	20
4.1 系統架構圖.....	20
4.2 轉譯器架構說明.....	21
4.2.1 轉譯器運作流程.....	21
4.2.2 轉譯器系統模組.....	23
4.2.3 上傳機制模組.....	24
4.3 PDA 樣板系統架構說明.....	25
4.3.1 PDA 樣板系統運作流程.....	25

4.3.2 PDA 樣板系統模組	26
4.4 手機樣板系統架構說明	27
4.4.1 手機樣板系統運作流程	27
4.4.2 手機樣板系統模組	28
五、 系統設計與實作	30
5.1 轉譯器之設計與實作	30
5.1.1 目標程式架構之設計	30
5.1.2 演員屬性設定	31
5.1.3 資源檔格式轉換	32
5.1.4 開場劇情及互動劇情處理	33
5.1.5 轉譯器各模組間的流程設計	35
5.2 PDA 樣板系統之設計與實作	37
5.2.1 樣板機制	37
5.2.2 電子郵件發送中心	39
5.3 手機樣板系統之設計與實作	40
5.3.1 樣板模組	40
5.3.2 簡訊發送中心	41
六、 實作範例	43
6.1 轉譯器	43
6.1.1 轉譯給 PDA	44
6.1.2 轉譯給手機	45
6.1.3 上傳機制	45
6.2 PDA 樣板系統	47
6.3 手機樣板系統	49
七、 結論	53
7.1 總結	53
7.2 未來發展方向	54
參考文獻或資料	55

表目錄

表一	CLDC 及 CDC 最少配備需求.....	9
表二	Embedded VB 與 Embedded VC++之比較.....	12
表三	PC 上視覺化編輯工具的描述檔說明	32
表四	各裝置上之互動式多媒體的資源檔格式.....	33
表五	描述檔中劇情的語法說明.....	33



圖目錄

圖一	多媒體繪圖處理功能的手機出貨量.....	2
圖二	多媒體潛在用戶特徵.....	3
圖三	Java 2 版本函式庫關係.....	7
圖四	J2ME 架構圖.....	8
圖五	MIDLet 開發流程圖.....	10
圖六	傳統 Java 程式執行程序.....	10
圖七	J2ME 執行程序.....	11
圖八	EVB 控制項.....	13
圖九	EVC 控制項.....	13
圖十	轉譯器示意圖.....	14
圖十一	上傳機制示意圖.....	17
圖十二	樣板機制示意圖.....	17
圖十三	系統架構圖.....	20
圖十四	轉譯器運作流程.....	22
圖十五	PDA 程式編譯流程.....	22
圖十六	手機程式編譯流程.....	23
圖十七	轉譯器系統模組.....	23
圖十八	本地端之上傳機制架構圖.....	24
圖十九	伺服器端之上傳機制架構圖.....	25
圖二十	PDA 樣板系統運作流程圖.....	26
圖二十一	PDA 樣板系統模組圖.....	27
圖二十二	手機樣板系統運作流程.....	28
圖二十三	手機樣板機制系統模組.....	29
圖二十四	轉譯器目標語言之類別圖.....	30
圖二十五	C++ 語言用來設定演員屬性的部分.....	32
圖二十六	開場劇情記錄格式.....	34
圖二十七	互動劇情記錄格式.....	34
圖二十八	劇情部份的描寫方法(Pseudo-Code).....	35
圖二十九	轉譯器模組流程圖.....	36
圖三十	樣板上傳機制架構圖.....	37
圖三十一	PDA 樣板置換流程圖.....	38
圖三十二	PDA 樣板機制程式流程圖.....	39
圖三十三	電子郵件發送流程圖.....	39
圖三十四	電子郵件發送中心程式流程圖.....	40
圖三十五	手機樣板置換流程圖.....	40
圖三十六	手機樣板模組程式流程圖.....	41

圖三十七	簡訊發送流程圖.....	41
圖三十八	智勝國際編輯手編輯環境.....	43
圖三十九	互動式多媒體內容在 PC 上呈現的效果	44
圖四十	互動式多媒體內容在 PDA 上呈現的效果	44
圖四十一	互動式多媒體內容在手機上呈現的效果.....	45
圖四十二	上傳機制示意圖.....	46
圖四十三	樣板網站內容.....	46
圖四十四	PDA 使用者 A 使用 PDA 樣板機制流程圖	48
圖四十五	PDA 使用者 B 接收到互動式多媒體之流程圖.....	49
圖四十六	手機使用者 A 使用手機樣板機制流程圖.....	51
圖四十七	手機使用者 B 接收到互動式多媒體之流程圖.....	52



一、緒論

1.1 手持行動裝置之應用價值的探討

古老的手機是為了聯繫而設計的，因此最重要的功能就是接電話及撥打電話，但是隨著時代的進步、科技的發達，手機的功能不再只是為了通話，越來越多的附加功能產生，包括照相、錄音、遊戲、GPRS 等等，使用者可以利用手機上網瀏覽網站、接收及傳送電子郵件或 MMS 多媒體簡訊，甚至使用 MSN Messenger 與朋友聊天溝通，此外，Java Game 更是大受使用者歡迎，要在手機上玩「大富翁」、「打磚塊」等 PC 遊戲都已經不是問題了，再加上現在照相手機已經非常普遍，幾乎成為手機的必備功能之一。這些附加功能的出現，均提高了手機的應用價值，因此，如果能有更多貼心有趣的應用程式，除了善加利用上述附加功能，還有新奇的內容及用途，將會更加吸引使用者。

除此之外，Pocket PC、PDA 手機、智慧型手機也都有普遍化的趨勢，且功能比手機更為強大，使用者可以把它當成小型個人電腦來使用，就如同行動助理一般，因此，如果能有一些應用程式來增加其趣味性，讓它們除了當助手外，還可以達到休閒娛樂的效果，將可以提高使用者的使用頻率，所以應用程式的存在價值也是不容置疑了。

手持行動裝置的被需要，最主要的原因就在於其「行動」的功能，因為大家會希望原來必須使用「固定位置」的裝置來做的事情，如使用個人桌上型電腦、電話機，可以變成隨時隨地都可以進行〔4〕，此外，行動教學的議題也不斷被討論，目的就在於希望學習者的學習可以不受地點、時間的影響，不僅在教室裡、課堂上可以學習，任何閒暇的時間在任何地點都可以用來學習，因此「行動裝置」的存在價值更是肯定的。

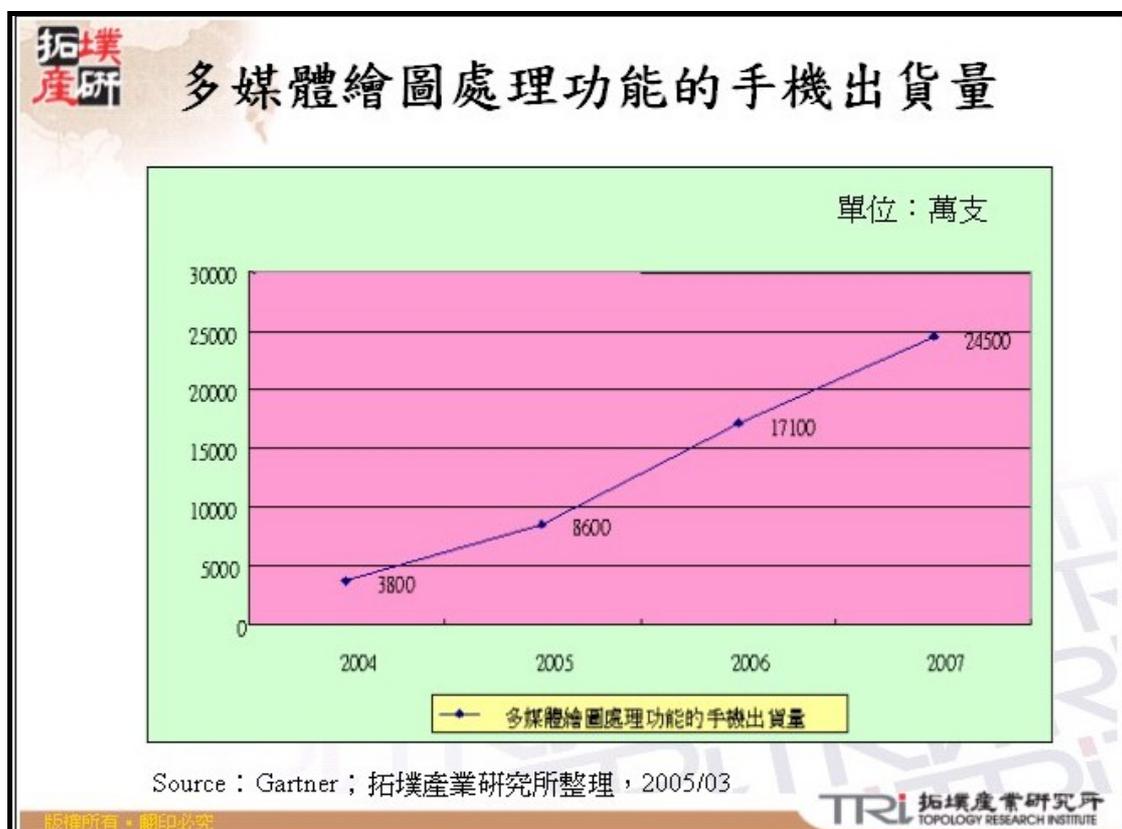
再者，手持行動裝置的體積有越來越小的趨勢，幾乎人人都可以隨身攜帶一個以上的手機或 PDA 等手持行動裝置，因此使用它們的機率將會大大地提高，除了電話功能或是利用它們來處理公事以外，其他的時間都可能會花費在使用應用程式上，所以在手持行動裝置上應用程式的價值相對提高許多，如何開發一個有價值又有意義的應用程式的確值得研究。

1.2 互動式多媒體之應用

使用動態的多媒體來做內容的呈現會比用靜態的圖片或文字來呈現還要活潑生動，多媒體是結合文字、圖形、聲音及動畫等優越展示能力，且較容易表達出內容的涵義，動畫的效果還有助於增加觀看者的注意焦點並可強調重點，若應用於教學上，不僅可以引起學習者的學習動機，學習者還可以透過多媒體接觸到

更多元的學習內容，提高學習效率。而透過互動的方式，使用者可以從操作過程中得到回饋，並根據不同的操作方法而得到不同的結果，如此會讓多媒體的內容不致於太過單調沒有變化，變化多端的內容將會更有趣且引人注目。因此，互動式多媒體使用，可以讓應用程式的內容變得更加生動活潑且有趣。

目前在 PC 上看到的互動式多媒體應用已有許多，常見的如：各式遊戲、電子賀卡、課程教材及試題、簡報等等，但在手持裝置上的應用還不是很多，最常見的就數遊戲類了，如手機上的 Java Game、PDA 上的新接龍及 JawBreaker 等等，都可算是互動式多媒體的應用。而 MMS 簡訊現在大受使用者歡迎，因為它具有多媒體的功能，可以在簡訊中加入圖片及聲音，甚至有動畫的效果，但其雖為多媒體，但並無互動的能力。



圖一 多媒體繪圖處理功能的手機出貨量

由圖一〔13〕我們可以知道，往後將會有更多手機都具備多媒體的處理功能，而且手機上支援多媒體的能力也會越來越強大，PDA 亦是如此，因此除了可以在手持行動裝置上開發遊戲類的互動式多媒體以外，互動式多媒體在 PC 上的其他應用，如電子賀卡、簡訊、學習類教材或是試題，也都可以嘗試在手持行動裝置上開發，此外，甚至可以考慮與家庭監視系統整合，提供即時簡訊或 e-mail 通知服務並傳送監視器的即時影像及時間地點到使用者指定的手持行動裝置上，讓使用者可以馬上得到家裡的情況，都能提高互動式多媒體在手持行動裝置上的應用價值。

1.3 研究動機與目標

由圖二〔13〕得知，使用多媒體手機的潛在用戶多為學生或是從事銷售業的上班族，因為他們對於手機的操作方法較為熟悉，且對於手機的新功能接受度高，因此比較容易願意去嘗試新手機，汰換手機的機率自然就提高了。但是他們大多數為多媒體使用者，而非製作者，因為要製作適用於手持行動裝置的多媒體應用程式並不容易，也不是他們的專長。

因此多媒體的來源大部份是從網路上下載，目前提供相關服務的網頁已經有許多，無論是手機廠商，如 okwap〔24〕，或是通信業者，如 emome〔22〕、i-style〔23〕，都有專屬網頁來提供多媒體簡訊服務或是 Java 遊戲下載，甚至也有許多熱心的網友提供各式各樣的 Java Game 讓別人下載。



圖二 多媒體潛在用戶特徵

因為手持行動裝置上的互動式多媒體製作並不容易，也不好上手，對於沒有程式背景的使用者來說更是困難，因此，互動式多媒體的應用類型會受到限制，使用者僅可以使用網路上提供的應用程式，如遊戲、MMS 簡訊等，對於 1.2 節提到的其他應用則很少在手持行動裝置上看到。但之前提到，如果能把互動式多媒體更廣泛地應用在手持行動裝置上，此外，使用者又可以很輕易地自行製作自己喜歡的互動式多媒體類型，將能提高手持行動裝置的應用價值，因此，本研究希望提出一個既簡單又方便的方案來製作手持行動裝置上的互動式多媒體，讓手持行動裝置上的互動式多媒體更普遍化。

以下我主要提出兩個方法來輔助使用者能更迅速製作手持行動裝置上的互動式多媒體：

➤ 轉譯器：

設計一個轉譯器來搭配 PC 上的視覺化編輯工具，此轉譯器主要的功用就是將此視覺化編輯工具編輯好的內容轉換成手持行動裝置可以播放的應用程式，如此一來，使用者便可以使用 PC 上的編輯工具來製作自己喜愛的互動式多媒體，接著只要透過轉譯器做轉換的工作，適用於手持行動裝置的互動式多媒體便能夠輕易產生了，大大降低製作的難度。

➤ 樣板機制：

內容的產生要從無到有往往需要很大的功夫，雖然有 PC 上視覺化編輯工具的幫助，但對使用者來說可能還是一個難題，因為不知道要從何下手才能產生漂亮的內容，因此需要一些提示或想法來做參考，所以本研究亦提供適用於手持行動裝置上互動式多媒體的樣板機制，讓使用可以到網路上尋找喜愛的樣板，只要做些修改便可以產生屬於自己的互動式多媒體，不僅減少編輯的步驟，更可以增加使用者使用互動式多媒體的興趣。

1.4 研究方法、步驟

為了完成適用於手機及 PDA 的轉譯器及樣板機制，我分成兩大部份來進行研究：

■ 轉譯器：

1. 選擇一個適當的視覺化編輯工具，並研究其描述檔的內容，分析描述檔的語法 [3]。
2. 選擇適合於手機及 PDA 的應用程式開發語言。
 - 手機：手機上的互動式多媒體是以 Java 應用程式來做呈現，其格式是 Jar 檔及相對應的描述檔 Jad 檔，因此我選擇將 PC 上的編輯工具之描述檔轉成 Java Code，並自動轉成適用於手機的 Jar 檔及 Jad 檔。
 - PDA：PDA 上的互動式多媒體是由 C++ 的執行檔(.exe) 再加上資源檔來形成，因此我選擇將 PC 上的編輯工具之描述檔轉成 C++ Code，再製作成適用於 PDA 的執行檔。
3. 研究並分析描述檔與目標程式的相對應關係，並規劃轉譯器的架構。
4. 實作手機及 PDA 的轉譯器

5. 實作上傳機制來將轉換好的應用程式上傳到樣板網站，當成樣板供其他使用者下載參考使用。

- 因為互動式多媒體包含了圖片及聲音檔，所以檔案大小都不算小，因此我打算利用 FTP 的方式來達到上傳的功能。
- 在上傳完成後，以 Http Request 的方式來告知 Web Server，以便即時更新 Web Server 上的網頁內容，避免發生不同步的狀況。

■ 樣板機制：

1. 研究適用於手機的樣板機制：

- 由於手機的限制較多，功能有限，直接在手機上開發樣板機制並不容易，因此我選擇提供樣板網站的方式，讓使用者在手機上透過瀏覽器在樣板網站上更換互動式多媒體內容中的一個或多個圖片及聲音，來達到樣板置換的功能，製作完後再自行下載到手機上播放。
- 提供代發簡訊的功能，讓使用者在完成樣板的置換後，可以發送簡訊給其他使用者，並告知互動式多媒體的下載網址，讓其他使用者也可以到網站上下載到手機上播放。

2. 研究適用於 PDA 的樣板機制：

- 直接在 PDA 中提供樣板的機制，讓使用者可以在 PDA 上更換互動式多媒體內容中的圖片及聲音，並馬上看到更換後的效果。
- 提供發送電子郵件的功能，讓使用者在完成樣板的置換後，可以將此互動式多媒體透過 e-mail 的附加檔案方式傳送給其他使用者。

1.5 研究範圍

本研究有一些前提和名詞定義，敘述如下：

- ◆ PC 上的視覺化編輯工具—智勝國際編輯手〔11〕
 - 以一幕一幕的場景為基礎，使用者較容易了解且易上手。
 - 用拖拉視覺化的方式來完成多媒體教材的製作。
 - 多媒體內容製作者不需要寫任何程式即可編輯出豐富劇情（scenario）的互動式多媒體內容。

- 擁有豐富的多媒體素材資料庫，亦提供加入其他多媒體資源的能力，提供更精彩的呈現效果。
- 提供劇情互動能力的編輯功能。
- ◆ 互動式多媒體：
 - 多媒體泛指圖片、聲音、文字等，互動則是會根據使用者的行為而有不同的回應，因此互動式多媒體是指這些多媒體具備與使用者互動的行為。
- ◆ 轉譯器：
 - 在讀取描述檔後，根據其內容產生出擁有相同效果的相對應程式碼，並製成執行檔，使其能在手機或 PDA 上播放。
- ◆ 手持行動裝置：
 - 手機：包括手機、智慧型手機。
 - PDA：包括 PDA 手機及 Pocket PC。
- ◆ 限制：
 - 手機：需支援 Java MIDP2.0 [10] (例如：Motorola、Nokia、BenQ、SonyEricsson 等等廠牌的中高階手機)
 - PDA：OS 需為 Microsoft Windows CE for Pocket PC

1.6 章節概要



在本章先探討在手持行動裝置上開發應用程式的價值，接著再提到目前互動式多媒體的應用及普及性，以及應用在手持行動裝置上的優點，最後再提出本論文中系統設計和製作的動機及目標，以及本論文系統的一些前提和名詞定義。

接著在第二章，分別研究了手機及 PDA 上應用程式的開發程式語言，並加以介紹。

在第三章中，將一個理想系統的規格需求歸納整理出來，主要分為轉譯器及樣板機制兩大部份。

在第四章，首先介紹本系統之整體架構，接下來分別介紹轉譯器、PDA 樣板系統及手機樣板系統的架構。

在第五章，針對系統中重要功能、模組及流程加以細部的解說，同樣地，也分為轉譯器、PDA 樣板系統及手機樣板系統三大部份。

第六章中，呈現系統操作流程及功能範例。

第七章為總結，提出此篇論文的結論及未來發展方向。

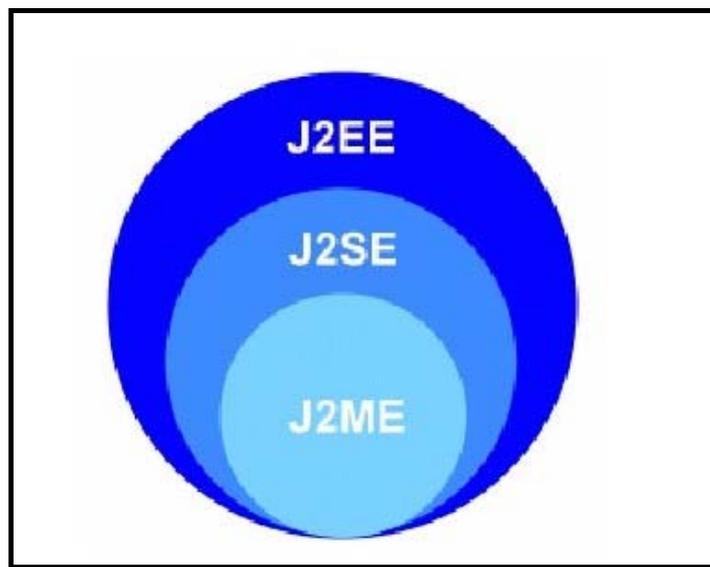
二、相關研究

2.1 手機應用程式開發之相關研究

2.1.1 為什麼要選用 Java

我們知道 Java 的一個重要特性就是跨平台，也就是說工程師只要撰寫一次程式碼，便可以在各種不同的平台上執行，亦即「Write Once, Run Everywhere」[5]，這個特性對手機上應用程式的開發更是重要，為什麼這麼說呢？因為現在手機進步越來越快，使用量也越來越大，所以有越來越多的廠商投入手機的開發及生產，但每一家廠商的手機可能有自己的作業平台，若想要在手機上開發應用程式，就必須針對每一種作業平台來開發在上面可以執行的應用程式，這個工程將十分浩大，此外若有一種新的作業平台的手機再推出，則必須再修改或重新開發新的應用程式，這樣的作法就變得不切實際。我們希望開發應用程式主要花時間和精力在應用程式的功能及可用性，而不是在平台的研究，因此選用 Java 來作為手機上的開發程式，如此，只要手機廠商在其作業平台上提供符合規格的 Java 虛擬機器（Java Virtual Machine）即可執行我們所開發的應用程式。

Java 的函式庫是被分類過的，也就是說所有的函式是被分門別類地放在不同的類別下，因此稱為類別函式庫。在 Java 規格中有一組核心類別函式庫（Core Class, 即 `java.*`）以及擴充類別函式庫（Extended Class, 即 `javax.*`），根據這兩種類別所支援的程度，Sun Microsystems 區分出三種不同的 Java 版本，如下圖所示：



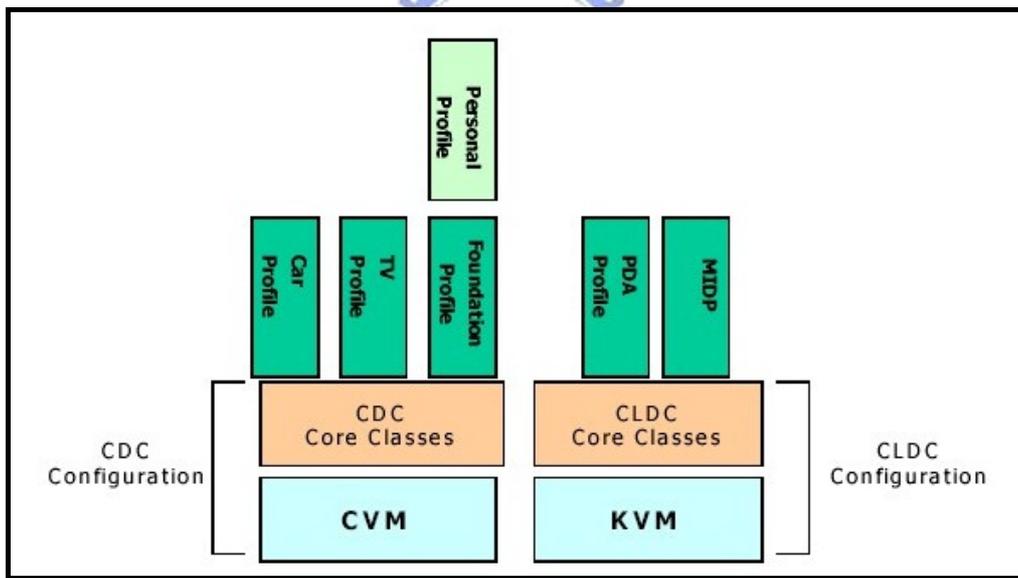
圖三 Java 2 版本函式庫關係

圖三說明了越往同心圓外面，支援的類別函式庫越完整，反之，越往同心圓內部則支援的類別函式庫越少。而 J2ME 便只有標準類別函式庫的子集合，但另外再加了一些支援嵌入式系統的擴充函式庫，如 javax.microedition.io.* 類別函式庫，主要是用來提供嵌入式系統、掌上型系統等較小型的作業平台的開發套件。

本研究所使用的 Java 版本便是 J2ME。

2.1.2 J2ME 介紹

在 J2ME(Java 2 Platform, Micro Edition)中為了達到更有彈性的架構，提出了 Configuration 和 Profile 兩種概念。所謂 Configuration 是針對某一個範圍內的裝置所定義的一些基本且通用的類別函式庫，例如有線裝置和無線裝置是使用不同的 Configuration，而 Profile 則是對於特定類型的裝置定義一組通用的函式庫，因此每一種不同類型的裝置架構即使架構在同一個 Configuration 上也可能擁有不同的 Profile，例如手機和 PDA 兩者架構在同樣為行動裝置的 Configuration 上，但 Profile 並不相同。



圖四 J2ME 架構圖

上圖即為 J2ME 的架構圖，J2ME 在設計規格時，遵循著「對於各種不同裝置而造出一個單一的開發系統是沒有意義的事」這個基本原則 [12][5]，因此將嵌入式裝置大致分成兩種，一種是運算功能有限、電力供應也有限的裝置，如手機、PDA 等，針對此類裝置，J2ME 定義了 CLDC (Connected Limited Device Configuration)；另一種是運算功能相對起來比較好、電力供應上相對也比較充足的裝置，如冷氣機、電冰箱等等，針對此類裝置，J2ME 定義了 CDC (Connected Device Configuration)。

表一 CLDC及CDC最少配備需求

	CLDC	CDC
RAM	兩者加起來再加上快閃記憶體(Flash Memory)需要 128K~512K	最少 256K
ROM		最少 512K
電源供應	多為電池，故電力有限	無限制
網路功能	僅支援有限的網路功能	必須支援網路連線

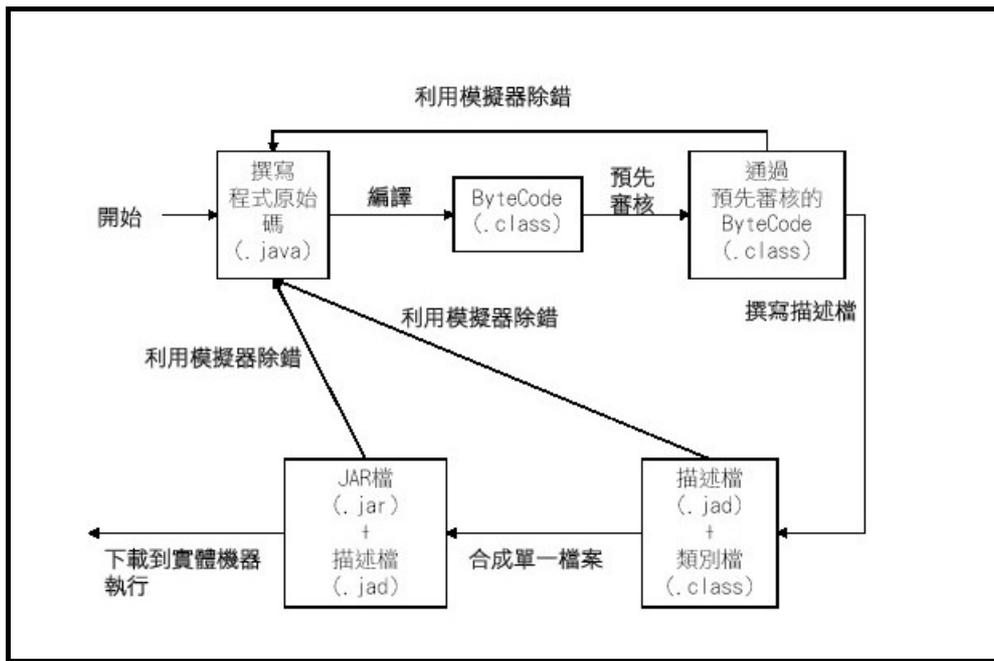
本研究是利用 MIDP(Mobile Information Device Profile)這個 Profile 來開發手機上的應用程式，由圖四我們可以知道 MIDP 是架構在 CLDC 之上，因此所有 CLDC 中的函式庫都可以在撰寫 MIDP 程式時呼叫使用，下一節將會更詳細介紹 MIDP。

2.1.3 MIDP 介紹

MIDP(Mobile Information Device Profile)主要是針對小型且資源受限制的行動裝置而發展的 Profile，可供廠商發展應用的類別函式庫 [5][10]。在 MIDP 內最基本的執行單位是 MIDlet，即一個可以執行的手機應用程式之基本單位，包含了程式執行時所需要的類別檔(.class)及資源檔(如圖片、文字檔等)，而多個 MIDlet 就組成了 MIDlet suite。

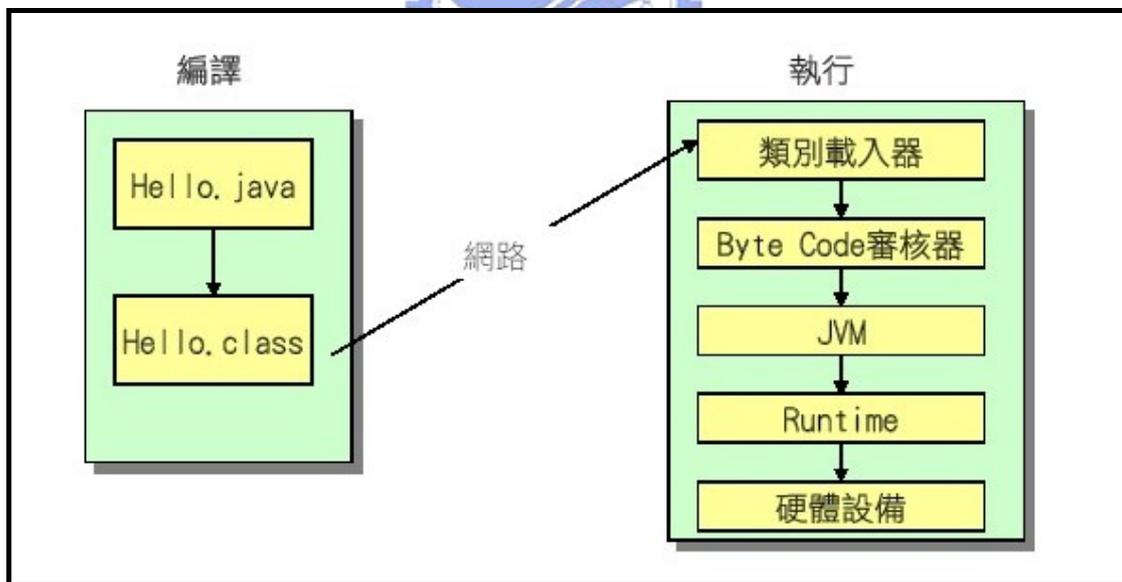
在手機上執行的 MIDlet 應用程式是被包裝成 Jar 檔，且每個 Jar 檔會有一個相對應的描述檔 (Description File, *.jad)，用來描述該 Jar 檔的基本資料以及內含的 MIDlet 之相關資訊(類別名稱、圖示、程式名)，Jad 檔是一個外部檔案，亦即不包含在 JAR 檔內部；除了 Jad 檔這個外部檔案以外，在 Jar 檔內也有一個內部的描述檔，稱為清單檔 (Manifest File)，也是用來描述這個 Jar 檔的基本資料，以及內含的 MIDlet 之相關資訊 (類別名稱、圖示、程式名)。

接著我們來看一個 MIDlet 的開發流程，如下圖所示：



圖五 MIDlet 開發流程圖

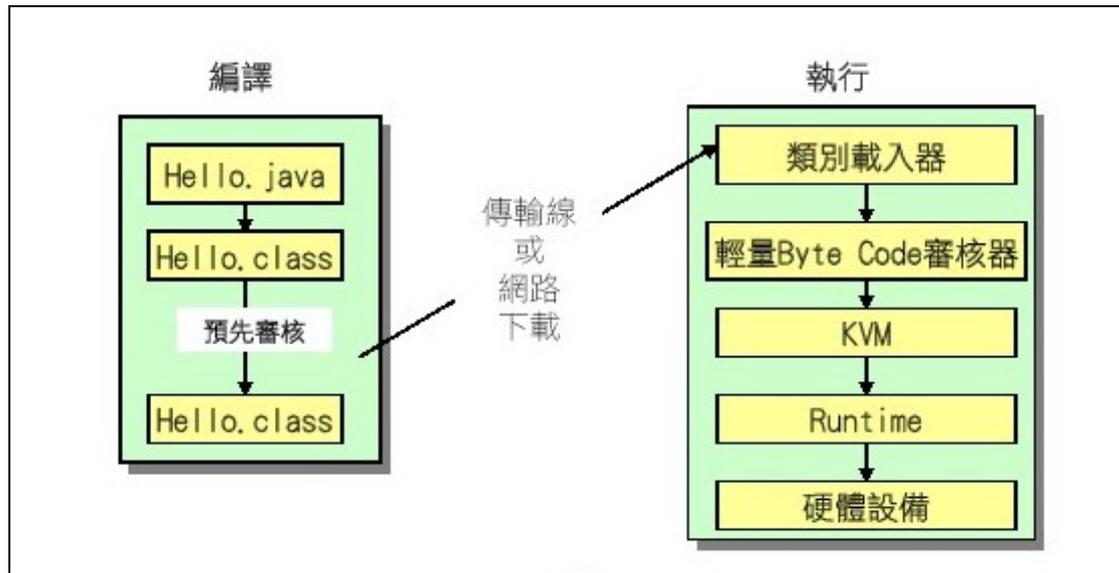
與一般開發 Java 應用程式一樣，需要經過撰寫程式碼、編輯、包裝成 Jar 檔等過程，但開發手機上的 Java 應用程式還需要經過「預先審核」這個步驟，以下說明一下這個步驟的用意。首先我們先來看傳統的 Java 執行程序：



圖六 傳統 Java 程式執行程序

基於安全的考量，任何要被執行的類別檔都要經過 Byte Code 審核器 (Byte Code Verifier) 的驗證，以防止程式被惡意更改，導致傷害到系統，因此要經過審核過程，確定安全後才能被執行，但這個過程如果要放到手機裝置上，可能會浪費許多資源，且執行速度也會減慢許多，因此在開發手機應用程式的過程中加入預先審核的動作，讓審核這個動作可以在 PC 上進行，不僅可以加速程式在

手機上的開啟時間及執行時間，更能減少裝置電量的耗費，以下即為 J2ME 程式的執行程序圖。



圖七 J2ME 執行程序

2.2 PDA 應用程式開發之相關研究

2.2.1 PDA 應用程式設計語言的選擇

目前市面上 PDA 大多數的作業系統均為微軟公司所提供的 Windows Mobile，且其上沒有內建安裝適用於 J2ME 的 KVM，這點與手機大不相同，為了避免使用者在使用應用程式之前還必須先安裝適當的 KVM，對使用者來說會造成另一種困擾，因此雖然 Java 具有跨平台的優點，但目前來看，PDA 的作業環境不如手機那麼多種，統一的程度也比手機好很多，因此 PDA 上的應用程式便不採用 Java，而考慮使用微軟提供的 eMbedded Visual Tool 系列來開發。

此系列包含了 C++ 及 Basic 兩種語言，以下我們先來做個比較 [6]：

表二 Embedded VB 與 Embedded VC++ 之比較

	eMbedded Visual Basic (EVB)	eMbedded Visual C++ (EVC)
執行速度	慢	快
進入門檻	低	高
快速開發	是	否
Runtime	可能需要額外安裝	內建
基本控制項	少	多
製作成 DLL	不可	可

Basic 是直譯式的語言，程式在執行時才會被一行一行地翻譯，而 C++ 已經先經過編譯，並產生執行檔，因此 C++ 的程式執行速度會比 Basic 的速度快；接著來看 Runtime 的部份，EVB 會將程式編譯成副檔名為 vb 的檔案，而這個檔案必須要有 Runtime 的支援才能夠執行，雖然目前的 PDA 上大多內建好 vb 的執行環境，但是某些程式仍可能會使用到更新的 Runtime，因此需要額外安裝，但 C++ 程式就是一支副檔名為 exe 的執行檔，因此不需要去考慮執行環境的問題。

但 EVB 的進入門檻較低，程式開發者比較容易上手，而使用 EVC 開發程式必須從無到有，且需要去了解訊息迴圈、Windows CE 所提供的 API 及物件導向的觀念等等，所以進入門檻相對較高。因此使用 EVB 開發程式會比較快速，而用 EVC 則會比較慢，且若開發不當，可能執行速度還會比 Basic 程式還慢。

三、系統功能需求分析

為了讓使用者能更容易且快速地製作適用於手持行動裝置上的互動式多媒體，因此本系統主要提供了兩大功能：

- 轉譯器：主要功能是把利用 PC 上編輯工具製作完成的互動式多媒體內容轉換成手持行動裝置上可以呈現的格式。
- 樣板機制：提供一套機制讓使用者可以利用現有的互動式多媒體樣板，經由幾個置換的步驟後產生新的互動式多媒體。

以上兩大功能皆針對手機和 PDA 系統環境的不同而有不同的需求，將詳細敘述於接下來的章節中。

3.1 轉譯器之功能需求

轉譯器主要的功能就是要能夠讀取 PC 上編輯工具的內容描述檔，並且解析描述檔內容，再根據描述檔的描述來自動產生目標程式碼，且此份程式碼經過編譯後產生的執行檔必須適用於手持行動裝置，且產生與 PC 上相同的內容。示意圖如下圖。



圖十 轉譯器示意圖

根據上圖的描述，轉譯器的需求還必須細分為轉譯給 PDA 及轉譯給手機兩個部份，此外，我們希望製作出來的互動式多媒體可以上傳到樣板網站上，供其他使用者下載參考使用，因此，上傳機制的實作也是屬於轉譯器的功能之一，接

下來就將轉譯器分為「轉譯給 PDA」、「轉譯給手機」以及「上傳機制」三個部份，分別於以下三小節作更詳細的需求敘述。

3.1.1 轉譯給 PDA

將 PC 上編輯好的互動式多媒體內容轉譯成適用於 PDA 之互動式多媒體，其需求如下：

(1) 與 PC 上編輯工具之溝通介面

需得知來源描述檔所在路徑、資源檔所在路徑、轉換後執行檔所在路徑及執行檔檔名。

(2) 轉譯器

在 PDA 上可以執行的互動式多媒體為 EXE 檔，因此轉譯器必須可以根據描述檔的描述，將 PC 上製作出來的內容轉換成相對應的 C++ Code，使其在 PDA 上能播放出相同的效果。

(3) 編譯成執行檔

將轉換後的程式碼加入編譯 (compile)，並產生出最終的執行檔。



3.1.2 轉譯給手機

將 PC 上編輯好的互動式多媒體內容轉譯成適用於手機之互動式多媒體，其需求如下：

(1) 與 PC 上編輯工具之溝通介面

需得知來源描述檔所在路徑、資源檔所在路徑、轉換後執行檔所在路徑及執行檔檔名。

(2) 轉譯器

在手機上可以執行的互動式多媒體屬於 Java 應用程式，因此轉譯器必須可以根據描述檔的描述，將 PC 上製作出來的內容轉換成相對應的 Java Code，使其在手機上能播放出相同的效果。

(3) 編譯並製作 Jar 檔

將轉換後的程式碼加入編譯 (compile)、預先審核 (Preverify)，再撰寫 Jad 描述檔，最後產生出 Jar 檔。

3.1.3 上傳機制

當 PC 上編輯好的互動式多媒體內容轉譯成適用於 PDA 或手機之互動式多媒體後，接著上傳到 Web Server 上供其他使用者可以下載，其需求如下：

(1) 取得轉譯器製作完成的目標執行檔

取得使用者欲上傳的互動式多媒體之目標執行檔及相關的資源檔。

(2) 與使用者之溝通介面

提供欄位讓使用者輸入 Server 的 IP 及 PORT，並顯示目前上傳進度以及在上傳完成後告知使用者。

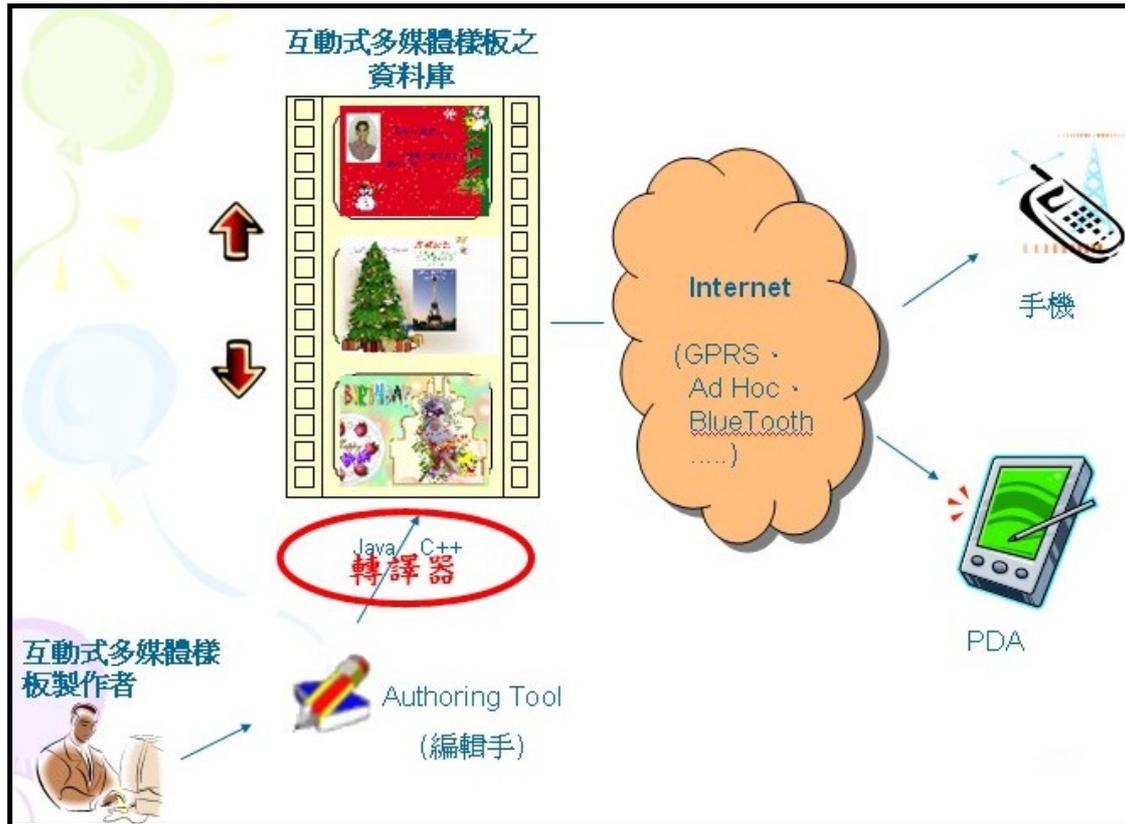
(3) 上傳到 Web Server

將各相關檔案利用 FTP 的方式上傳到 Web Server 上，並在上傳完成後告知 Web Server 這個上傳的事件已完成。

(4) 產生相對應的網頁

Web Server 端接收到使用者上傳的檔案以後，需改變原本的網頁，使得新上傳的樣板可以即時顯示於網頁中，並提供其超連結給使用者下載。此外，手機的樣板機制是在 Web Server 上進行，因此也必須產生相對應的樣板置換網頁。上傳機制示意圖如下圖。





圖十一 上傳機制示意圖

3.2 樣板機制之功能需求

樣板機制的主要功能是提供使用者可以置換互動式多媒體樣板中的資源檔 (Resource)，如圖片、聲音、文字等等，並產生新的互動式多媒體，示意圖如下圖。



圖十二 樣板機制示意圖

同樣地，我們也將樣板機制分為手機及 PDA 兩個部份來分別描述其需求。

3.2.1 PDA 之樣板機制

PDA 的樣板機制是實作在 PDA 內，主要需求為置換的動作以及發送電子郵件，詳細敘述如下：

(1) 取得互動式多媒體樣板中要被取代的資源檔

透過使用者的點選動作來得知要被取代的資源檔為何。

(2) 取得使用者想使用的資源檔

使用者可以透過錄音、照相或來產生資源檔，亦可直接選擇 PDA 內建的檔案庫中的圖片或聲音檔，此外，若使用者想更換樣板中的文字內容則提供對話框讓使用者自行輸入文字。

(3) 置換

將使用者想使用的資源檔取代原本的互動式多媒體樣板中的資源檔，並重新播放置換完成的互動式多媒體。

(4) 發送電子郵件

將互動式多媒體利用附加檔案的方式，傳送給使用者填入的電子郵件信箱位址。



3.2.2 手機之樣板機制

由於手機的限制較多，因此樣板機制的設計是利用遠端的方式來達成，也就是說需透過手機內的瀏覽器連上 Web Server，在 Web Server 上進行樣板置換的動作，因此需要一個 Web Server 以及相關網頁來處理置換的動作以及發送簡訊的工作，詳細敘述如下：

(1) 手機樣板機制專屬網站

需要一個手機樣板機制的專屬網站來負責手機樣板置換的工作，手機樣板機制的的所有工作均在此網站上完成。

(2) 取得互動式多媒體樣板中要被取代的資源檔為何

根據使用者在網頁上的選擇來得知要被取代的資源檔為何，並提供相對應的超連結，讓使用者可以繼續接下來置換的動作。

(3) 取得使用者想使用的資源檔

利用上傳的方式讓使用者上傳資源檔。

(4) 置換

將使用者上傳的資源檔取代原本的互動式多媒體樣板中的資源檔，並製作新的互動式多媒體 Jar 檔及 Jad 檔供使用者下載。

(5) 發送文字簡訊

取得使用者填入的手機號碼，並發送文字簡訊給該手機，在文字簡訊中寫入置換好的互動式多媒體所在的 URL，若接受端想觀看此互動式多媒體，則自行根據 URL 連上網站下載。



四、系統架構

4.1 系統架構圖

在這個小節裡，我們描述出整體系統的模組與架構。



圖十三 系統架構圖

每一個模組代表一個獨立的功能，視使用者要使用哪一個功能而有不同的呼叫方式及執行流程。

- PDA 轉譯器：負責將描述檔內容轉換成 C++ 語言程式碼，並製作適用於 PDA 的執行檔的模組。
- 手機轉譯器：負責將描述檔內容轉換成 Java 語言程式碼，並製作適用於手機的執行檔的模組。
- 樣板上傳機制：用來將轉換好的互動式多媒體上傳到 Web Server 的模組。

- PDA 樣板機制：提供適於用 PDA 的樣板機制，讓使用者有另一種製作 PDA 上的互動式多媒體的方法。
- 手機樣板機制：提供適用於手機的樣板機制，讓使用者有另一種製作手機上的互動式多媒體的方法。
- 電子郵件發送中心：配合 PDA 樣板機制，讓使用者可以將 PDA 上的互動式多媒體以附加檔案的方法傳送給其他 PDA 使用者。
- 簡訊發送中心：配合手機樣板機制，讓使用者可以將手機的互動式多媒體的相關訊息以文字簡訊的方式傳送給其他手機使用者。

我們將整個系統再細分成三個子系統—轉譯器、PDA 樣板系統以及手機樣板系統，在接下來的章節，我們均分別針對這三個子系統做更詳細的介紹：

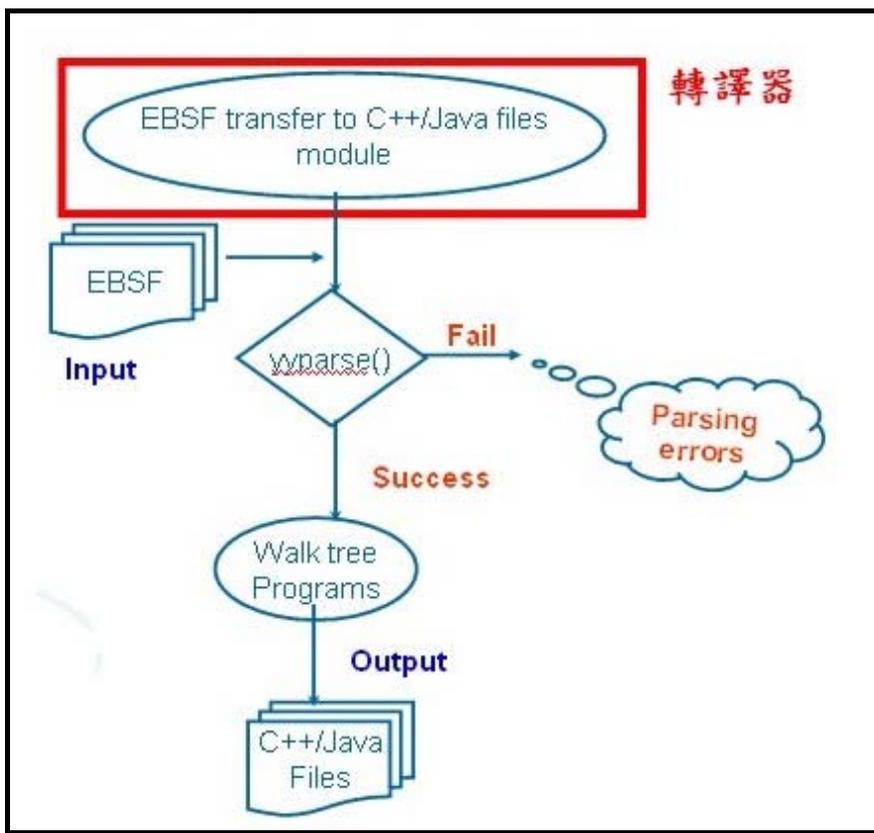
- 轉譯器：包含了 PDA 轉譯器、手機轉譯器以及樣板上傳機制。
- PDA 樣板系統：包含了 PDA 樣板機制以及電子郵件發送中心。
- 手機樣板系統：包含了手機樣板機制以及簡訊發送中心。

各個模組的細部架構我們將在接下來各小節一一描述。

4.2 轉譯器架構說明

4.2.1 轉譯器運作流程

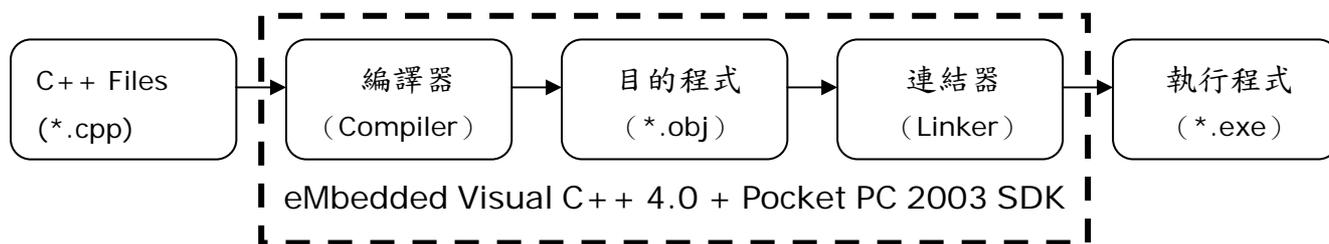
轉譯器的輸入是視覺化編輯工具的描述檔 (EBSF, e-Bestwise Script Files)，在讀入描述檔後會進行語法的分析並建立相對應的結構樹，若分析失敗則顯示錯誤訊息，若成功則開始尋訪結構樹以獲得相關資訊，再根據這些資訊來產生目標程式碼。



圖十四 轉譯器運作流程

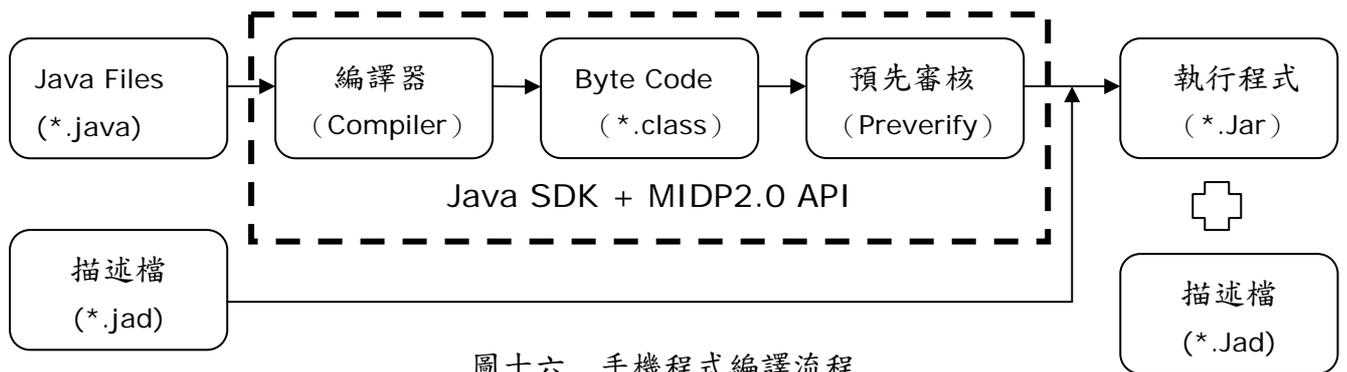
當相對應的程式碼都產生後，再呼叫編譯工具來進行編譯的工作，接著產生執行檔，手機和 PDA 使用的編譯工具並不相同，以下就個別介紹：

- PDA：轉譯給 PDA 的目標程式碼為 C++，在產生 *.cpp 檔後，再使用 Command-Line 的方式呼叫 eMbedded Visual C++ 4.0，並搭配 Pocket PC 2003 SDK 來進行編譯的工作，其流程如下：



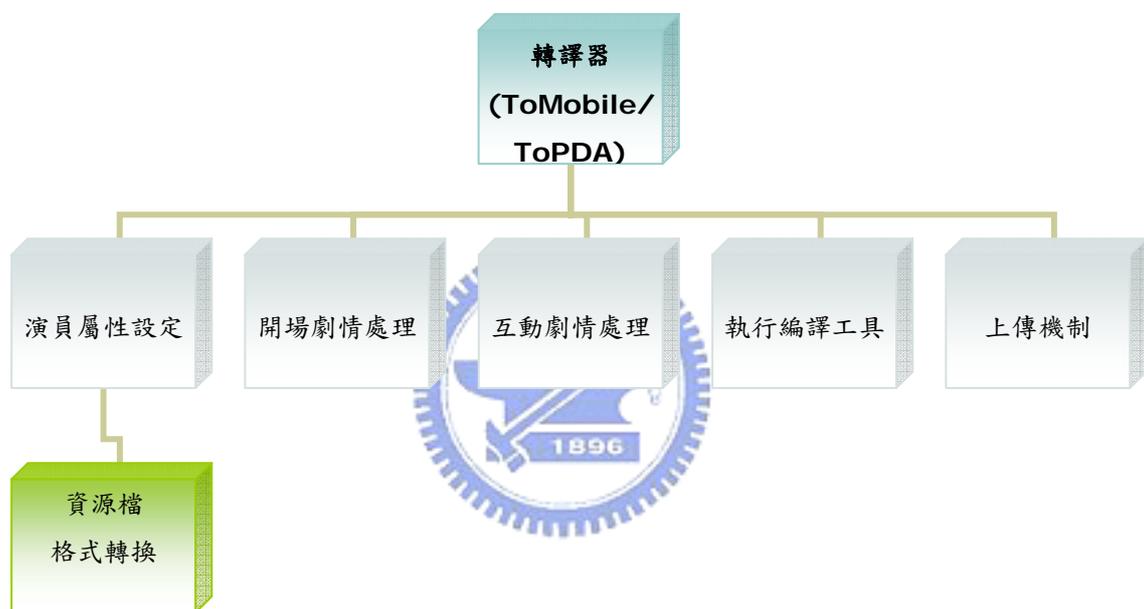
圖十五 PDA 程式編譯流程

- 手機：轉譯給 PDA 的目標程式碼為 Java，且編譯出來的 Jar 檔，還需要有一個相對應的描述檔 (*.jad) 來描述 Jar 檔，因此除了產生 *.java 檔外，還有另一個 *.jad 檔，接著再利用 Java SDK 及 MIDP2.0 提供的函式庫來進行編譯，流程如下：



圖十六 手機程式編譯流程

4.2.2 轉譯器系統模組



圖十七 轉譯器系統模組

根據描述檔的語法及描述，一個互動式多媒體的內容主要包括了各個演員的屬性、開場劇情的表現方式以及各演員間互動的關係，此外，還要將轉換好的程式碼編譯成執行檔，並且提供上傳的機制，因此轉譯器便分成五大模組，分別用來處理上述內容的轉換以及編譯和上傳的工作。

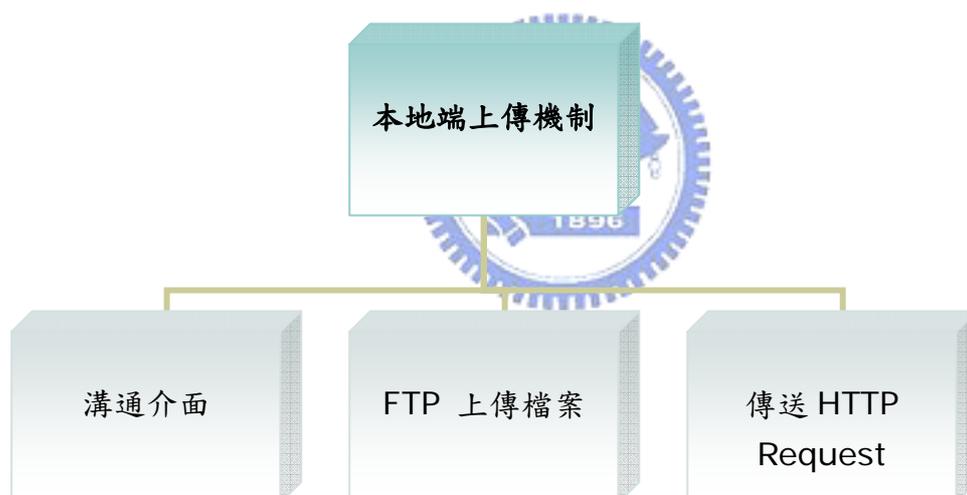
- 演員屬性設定：所謂演員是指互動式多媒體中演出的成員，包含了圖片、動畫、聲音、文字等，而每個演員都有專屬的各種屬性，包括資源檔來源、演出開始及結束位置、大小、演出路徑、進階屬性等等，此外，資源檔的格式以及圖檔的尺寸大小都可能因目標裝置不同而必須要做相對應的轉換和改變，亦在這個模組做處理。
- 開場劇情處理：當互動式多媒體中的每一幕場景一開始啟動播放，此時的内容呈現稱為開場劇情，而這個模組就是根據描述檔的描述來處理並設定開場劇情的内容。

- 互動劇情處理：演員之間互動的行為稱為互動劇情，亦即每個演員會觸發的劇情，此模組就是根據描述檔的描述來處理並設定各段互動劇情的內容。
- 執行編譯工具：將產生出來的 C++ 或 Java 程式碼轉換成相對應的 exe 檔或 Jar 檔。
- 上傳機制：即將製作好的互動式多媒體上傳到 Web Server 成為樣板。此部份的系統架構我們在下一節會做更詳細的介紹。

4.2.3 上傳機制模組

上傳機制分為本地端及伺服器端，本地端即為互動式多媒體編輯者的那一端，也就是上傳的那一方，而伺服器端即 Web Server，也就是接收端。以下分別說明兩端的系統架構圖。

- 本地端：與轉譯器連結，將製作好的樣板上傳到 Web Server。

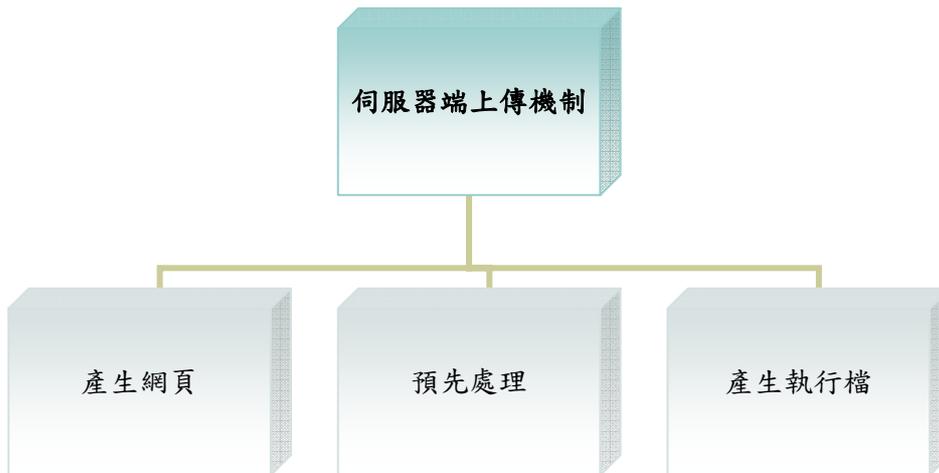


圖十八 本地端之上傳機制架構圖

本地端上傳機制主要分成三個模組，分別為溝通介面、FTP 上傳檔案以及傳送 HTTP Request：

- 溝通介面：負責和轉譯器及使用者溝通，取得要上傳的檔案路徑以及 Web Server 的 IP 和 Port。
- FTP 上傳檔案：利用 FTP 的方式，將所有檔案上傳到使用者指定的 Web Server 上。
- 傳送 HTTP Request：在檔案全部上傳完畢後，傳送 Request 給 Server 端的上傳機制，以便處理接下來的動作。

- 伺服器端：接收到樣板後所需要做的處理動作。



圖十九 伺服器端之上傳機制架構圖

伺服器端上傳機制主要分成三個模組，產生網頁、預先處理以及產生執行檔：

- 產生網頁：為使用者上傳的互動式多媒體樣板產生相對應的網頁，並且提供超連結和手機樣板機制相關連結網頁。
- 預先處理：必須根據目前 Web Server 上的資訊來決定新上傳的樣板名稱及存放位置，因此可能需要替此樣板做一些修改及調整。
- 產生執行檔：將此互動式多媒體樣板編譯成執行檔（註一）。

註一：為什麼不在本地端使用轉譯器時便產生 Jar 檔或 exe 檔，然後再將檔案上傳？這是因為樣板上傳到伺服器端後必須由伺服器端來做管理，因此 Jar 檔或 exe 檔的名稱及相對位置必須由伺服器端來決定，故在本地端上傳前並不編譯的工作，僅做轉換程式碼的工作，等上傳到伺服器後再由樣板機制處理器來編譯。

4.3 PDA 樣板系統架構說明

4.3.1 PDA 樣板系統運作流程

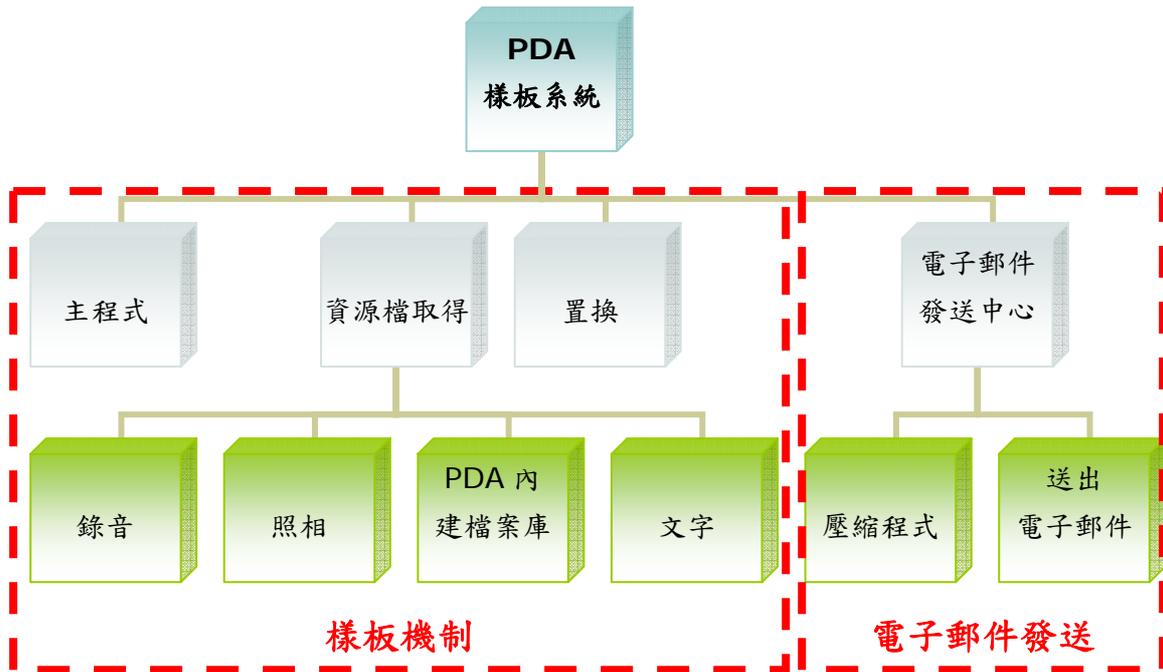


圖二十 PDA 樣板系統運作流程圖

PDA 樣板系統直接實作在 PDA 中，而主要流程如圖二十所示，使用者在播放互動式多媒體後，在切換到樣板機制模式後，即可以直接點選此互動式多媒體中的圖片、文字或聲音來做置換的工作，若置換成功則可以立即看到置換後的結果，接下來使用者可以選擇是否要將此互動式多媒體以電子郵件傳送給其他人；亦或是使用者也可以不需經過樣板置換的動作，直接將此互動式多媒體以電子郵件傳送給其他人。

4.3.2 PDA 樣板系統模組

根據上一小節的描述，我們將 PDA 樣板系統分成幾個模組，如下圖所示：

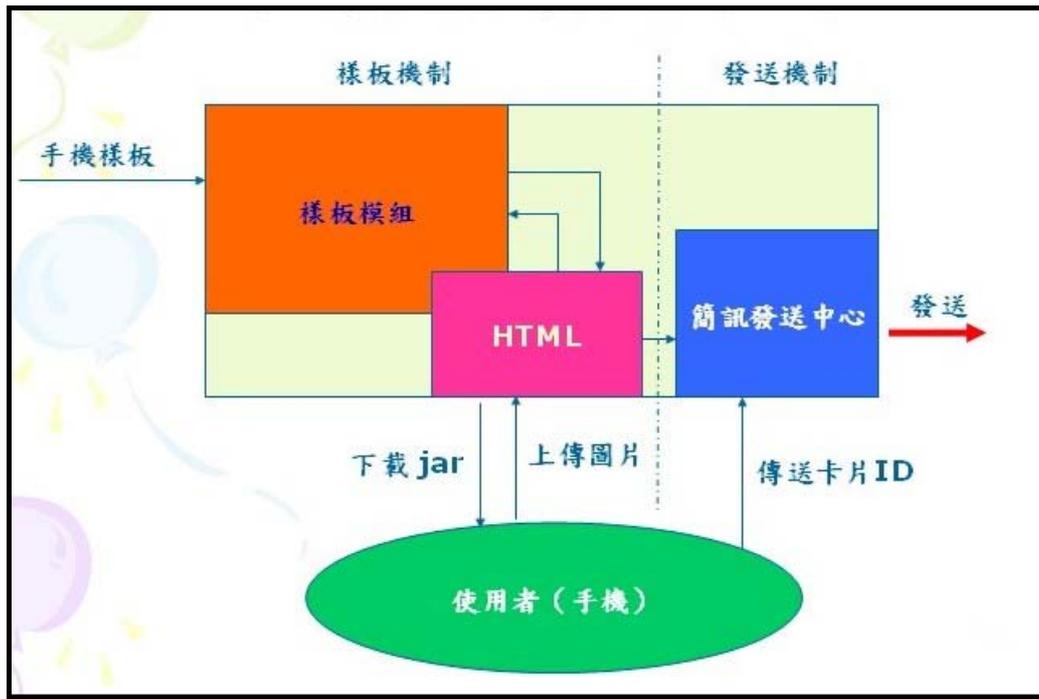


圖二十一 PDA 樣板系統模組圖

- 主程式：用來取得使用者目前所想要換的資源檔為何。
- 資源檔取得：使用者有四種取得新資源檔的方式，分別為錄音、照相、使用 PDA 內建的檔案庫、文字輸入，根據使用者的選擇來得到要用來置換的資源檔為何。
- 置換：用新的資源檔取代原來互動式多媒體中的資源檔。
- 電子郵件發送中心：必須先將互動式多媒體壓縮成 Zip 檔，接著再取得對方的電子郵件信箱地址及信件主旨，最後以附加檔案的方式將互動式多媒體送出。

4.4 手機樣板系統架構說明

4.4.1 手機樣板系統運作流程

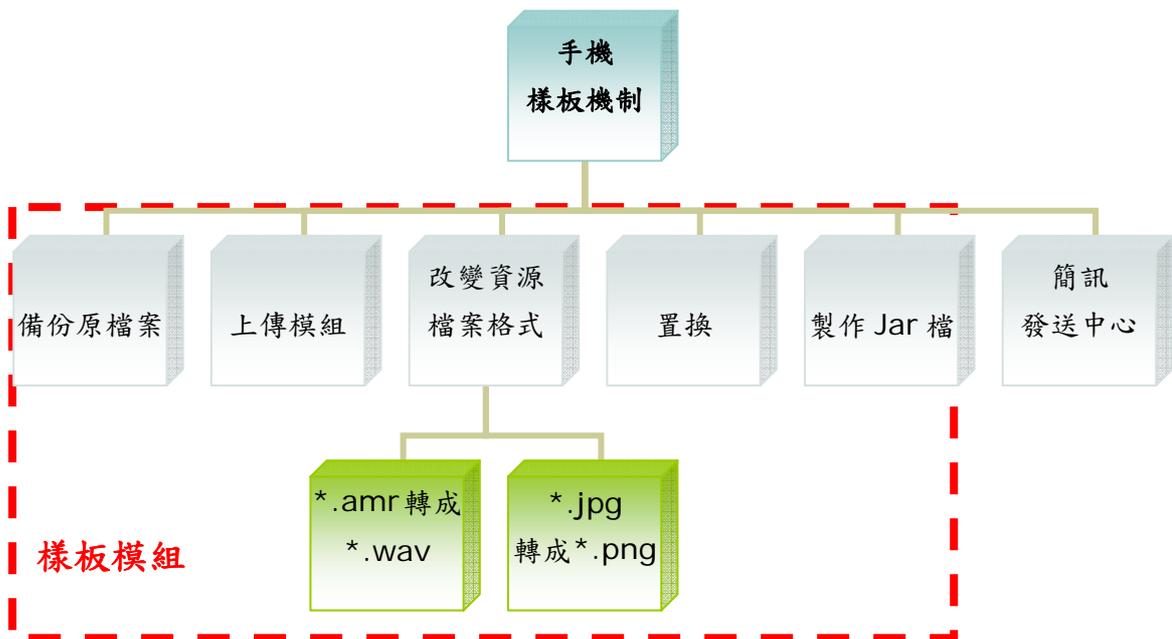


圖二十二 手機樣板系統運作流程

手機樣板系統實作在樣板網站(即 Web Server)上，因此必須利用遠端的方式，連接上樣板網站進行。手機樣板在上傳到樣板網站後會進入樣板模組，並產生相對應的 HTML 網頁，因此，使用者並不會直接和樣板模組溝通，而是透過 HTML 網頁作為操作介面，再由 HTML 網頁與樣板模組溝通，此外，使用者可以透過 HTML 網頁呼叫簡訊發送中心來發送簡訊，亦可在手機上直接呼叫簡訊發送中心來發送簡訊。

4.4.2 手機樣板系統模組

根據上一小節的描述，我們將手機樣板系統分成幾個模組，如下圖所示：



圖二十三 手機樣板機制系統模組

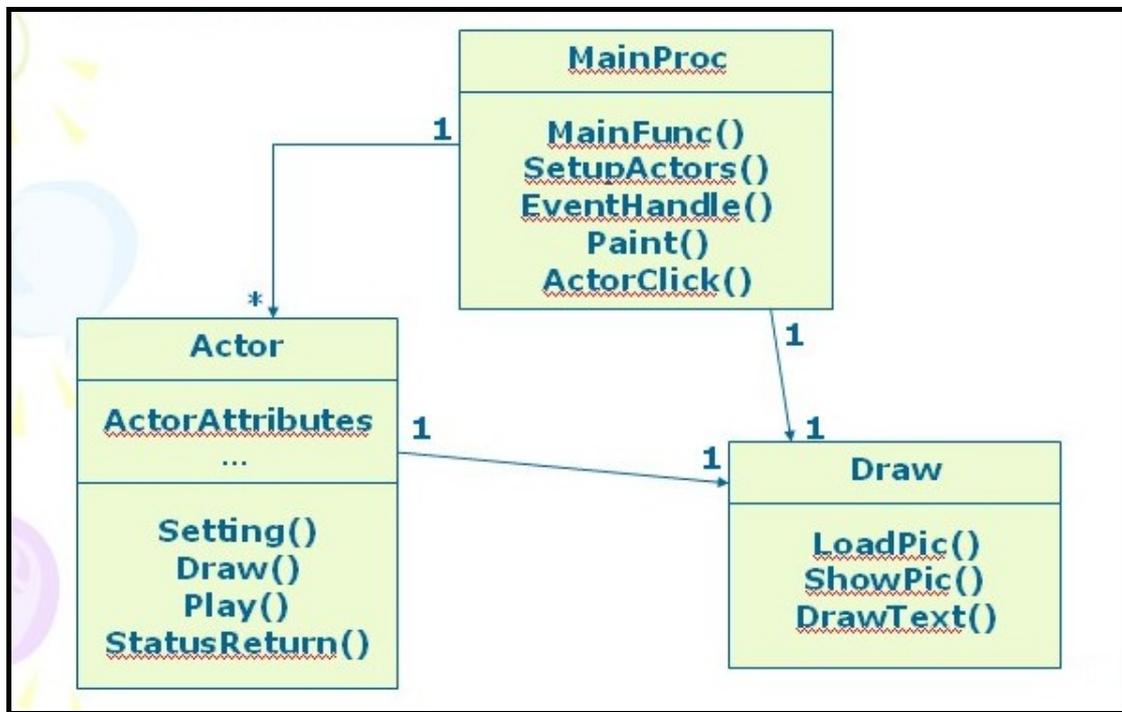
- 備份原檔案：將要被取代的資源檔作備份。
- 上傳模組：用來接收使用者上傳的資源檔。
- 改變資源檔案格式：將使用者上傳的資源檔作格式的轉換，以適用於手機上互動式多媒體的製作，例如：在手機上錄音產生的音樂檔為 amr 格式，必須轉換成 wav 格式；在手機上照相產生的圖檔為 jpg 格式，必須轉換成 png 格式，此外，圖片的尺寸大小也必須和原互動式多媒體中的圖檔相同，因此必須做調整。
- 置換：用新的資源檔取代原來互動式多媒體中的資源檔。
- 製作 Jar 檔：將置換完成的互動式多媒體重新製作成 Jar 檔。
- 簡訊發送中心：發送文字簡訊，簡訊中註明此互動式多媒體所在的 URL，以供接收簡訊端下載。

五、系統設計與實作

5.1 轉譯器之設計與實作

5.1.1 目標程式架構之設計

在本系統中，轉譯給手機的部份採用 Java 語言，而轉譯給 PDA 的部份採用 C++ 語言，雖然是屬於不同的語言，但對於作為本系統的目標語言來說，有許多共通點，因此不需要針對兩種語言而有不同的設計，只要在實作時各自使用自己的元件及語法即可，以下便是此編譯器目標語言之類別圖。



圖二十四 轉譯器目標語言之類別圖

在目標程式中有三個類別，分別為 MainProc、Actor 及 Draw。MainProc 是主要的類別，在 MainProc 中會有許多 Actor 類別型態的物件，每一個 Actor 物件代表互動式多媒體中的每一個演員；除了 Actor 物件外，還會有一個 Draw 類別型態的物件，用來畫背景圖片，因為背景圖並不屬於演員的一份子。而 Actor 中也會有一個 Draw 類別型態的物件，用來畫此演員本身的圖片。以下針對每個類別做詳細說明：

- MainProc 類別：在這個類別中有幾個重要的函式(function)：
 - MainFunc(): 用來處理開場劇情的部份，採用 Thread 或是 Timer 的方式來設計，因為場景的內容為動態顯示，所以必須不斷地更新

各個演員的屬性，以達到移動或動畫的效果。

- **SetupActors()**：用來設定每個演員的屬性，包括移動路徑、資源檔來源、動畫的 Frame 個數、文字的各项屬性等等。
 - **Paint()**：除了畫背景圖以外，還會呼叫每個 Actor 的 **Draw()** 來畫出每一個演員，此 function 也是必須被不斷地呼叫，以達到即時更新顯示畫面的效果。
 - **EventHandle()**：用來處理互動事件的部份，在接收到使用者的互動動作（如：使用滑鼠點選 PDA 螢幕或是使用手機方向鍵移動選擇方框（註二））後作相對應的處理。
 - 而 **ActorClick()** 則是用來記錄演員的互動劇情，每一個擁有互動劇情的演員都會有相對應的 **ActorClick()**。
- **Actor**：記錄了演員的各项屬性，此外，還有一些與演員相關的函式，例如：專門用來畫圖的 **Draw()** 及用來播放動作（包含移動路徑、音效及圖片 Frame 的變換）的 **Play()**，以及回傳演員目前狀態的 **StatusReturn()** 等等。
 - **Draw()**：主要用來畫圖，因此有 **LoadPic()**、**ShowPic()**、**DrawText()** 等函式。

註二：有些手機沒有觸碰式螢幕的功能，因此若要選取互動式多媒體中的演員則必須使用方向鍵。在手機互動式多媒體的畫面上會出現一個方框，並根據使用者按上下鍵來移動方框的位置，直到選取到使用者想要的演員。

5.1.2 演員屬性設定

在互動式多媒體的內容中存在許多演員，包括圖片、聲音、文字等等，每個演員都有屬於自己的一些屬性需要被設定，這些相關屬性在描述檔裡都會有記錄，如表三所示，因此本系統便根據描述檔的記錄，來寫出相對應的目標語言程式碼（C++ 或 Java 程式），以下便以 C++ 語言為例，舉其中一個圖片演員來作說明，如圖二十五所示。

表三 PC 上視覺化編輯工具的描述檔說明

描述檔內容	註解說明
Position = 45 281 127	此演員初始位置
PicFile = "C0114.jpg"	此演員的圖檔來源
FrameCount = 9	此演員的圖檔是由 9 個 Frame 組成
Segment0 = 5 (2, 4) (26, 114) (67, 36) (78, 22) (39, 56)	此演員的路徑有這 5 個座標
TextStyle = (16,9) (97,124) "標楷體" 1 1 0 0 255 0 0 14 1	此演員上的文字相關屬性
TextCount = 2	此演員上的文字行數
TextLines = "你好" "Hello"	此演員上的文字內容

```

_tcscpy(FileName, __TEXT("\\forPDA\\Sc529res\\C0114"));
num = 9;
HBITMAP images2[num];
loadFrames(DeviceContext,FileName,images2,num);

int path2[]={ 15,140,15,138,24,81,46,63,72,52,85,24};
p_num = 12;

actor = new Actor(images2,num,path2,p_num,102);

_tcscpy(tempText[1], __TEXT("Hello"));
_tcscpy(tempText[0], __TEXT("你好"));

actor->TextSetting(tempText,textnum,2);
actor->SetTextRGB(255, 0, 255);
actor->SetStyle(1,0,0,0);
actor->SetFontAndSize(__TEXT("標楷體"), 18);
    
```

圖二十五 C++ 語言用來設定演員屬性的部分

5.1.3 資源檔格式轉換

在手機和 PDA 上的互動式多媒體中所需用到的資源檔格式和 PC 上是不相同的，因此在轉換程式碼的過程中也需要做格式轉換的工作，表四列出互動式多媒體在 PC、手機及 PDA 中所需的資源檔格式。

表四 各裝置上之互動式多媒體的資源檔格式

	PC	手機	PDA
圖片	多種圖片格式	Png	Bmp
背景圖	Jpg	Png	Jpg
演員配樂	Wav MP3	Wav	Wav
背景音樂	Midi	Midi	Wav

5.1.4 開場劇情及互動劇情處理

在互動式多媒體中最精采的部份莫過於劇情的部份，因此劇情的轉換是相當重要的，而劇情又分為開場劇情及互動劇情，兩種劇情主要的差別在於播放的時機不相同，開場劇情是在場景一開始播放即演出，而互動劇情必須等待使用者觸發，雖然播於時機不同，但其實播放原理是相同的，所以設計方法也是一樣的，在此一併作說明。首先我們先來了解在描述檔中是如何描述開場劇情及互動劇情。

	描述檔內容	說明
開場劇情	<pre>[PRELUDE] { parallel(Actor000.play(), Actor.play()); Actor002.play(); } [END_PRELUDE]</pre>	Actor000 和 Actor 同時演出結束後，換 Actor002 演出
互動劇情	<pre>ANCHOR Actor000 : { LMOUSECLICK: { parallel(Actor.play(), Actor002.play()); Actor001.play(); } . . . }</pre>	在 Actor000 被按下後，Actor 和 Actor002 同時演出結束後，換 Actor001 演出

表五 描述檔中劇情的語法說明

了解描述檔中劇情部份的語法後，我們再來看轉譯器程式在轉換的過程中如何記錄這些劇情：

以上述描述檔內容為例：

■ 開場劇情：

Actor000	Actor	...	Actor002
Parallel			Sequential			...

圖二十六 開場劇情記錄格式

每一粗方格內代表一個劇情，除了記錄參與演出的演員有哪些以外，還要記錄那些演員的演出是依序演出（Sequential）或是同時演出（Parallel），因此，由以上的記錄我們可以知道：共有兩段劇情，第一段劇情是 Actor000 和 Actor 同時演出，接下來是 Actor002 的演出。

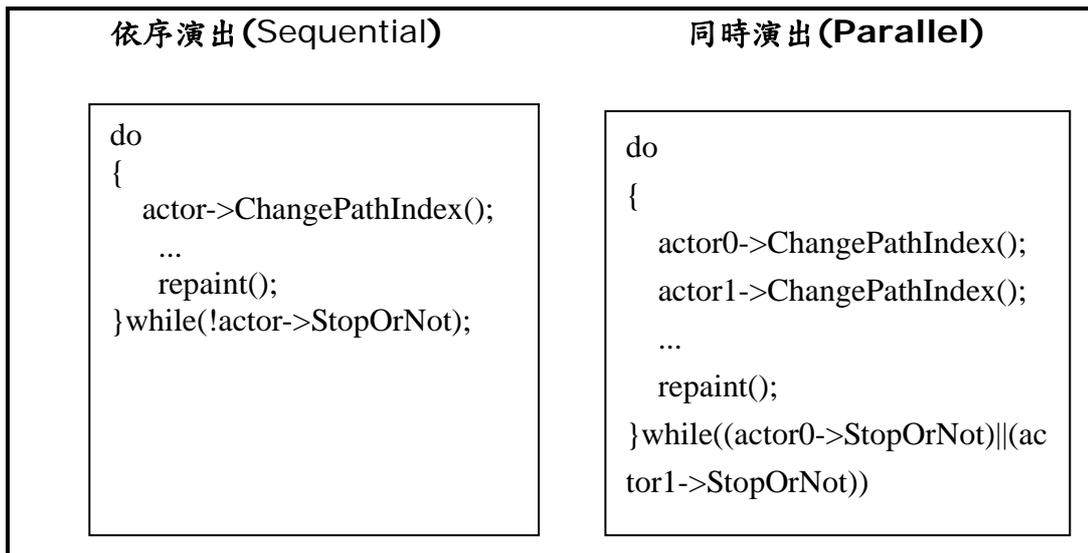
■ 互動劇情：

Actor000			Actor000			
Actor	Actor002	...	Actor001
Parallel			Sequential			...

圖二十七 互動劇情記錄格式

和開場劇情一樣，每一粗方格內代表一個劇情，同樣也必須記錄參與演出的演員以及那些演員的演出是依序演出（Sequential）或是同時演出（Parallel），以外，還必須記錄這段劇情是由於哪個演員被觸發而產生的，因此，從以上的記錄我們可以知道：共有兩段劇情，均是因為 Actor000 被觸發而產生的，第一段劇情是 Actor 和 Actor002 同時演出，接下來是 Actor001 的演出。

由上述的記錄我們就可以開始來設計程式中的運作方式，我們知道開場劇情和互動劇情的差別僅在於播放時機不同，因此我們把開場劇情寫在主程式中，而互動劇情則寫在擁有互動劇情的演員的相對應的 ActorClick() 中，寫法如下圖所示：



圖二十八 劇情部份的描寫方法(Pseudo-Code)

■ 依序演出 (Sequential) :

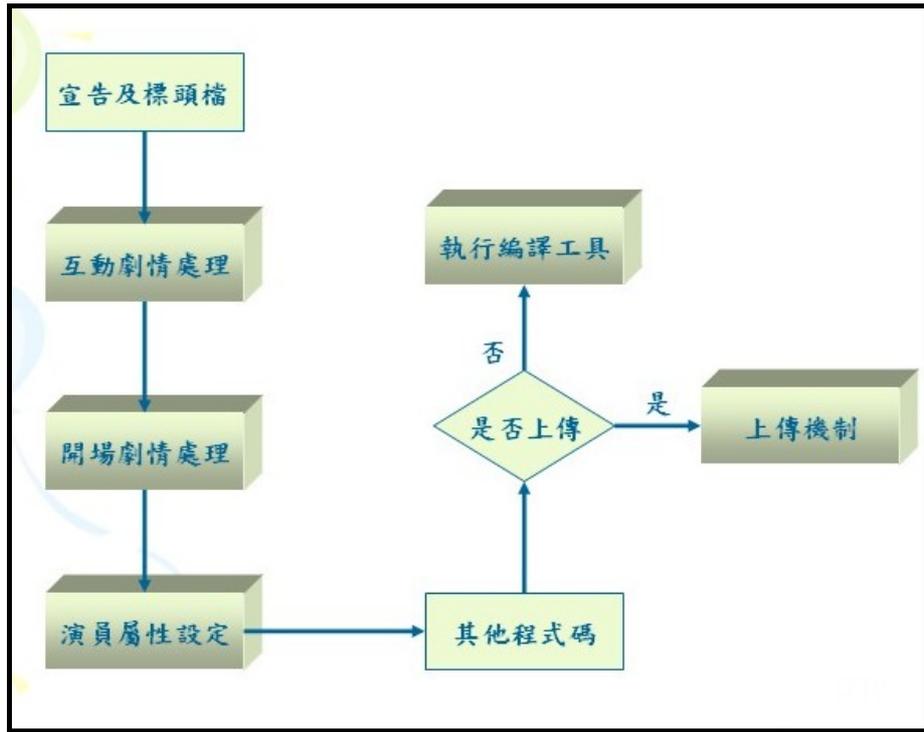
利用迴圈的控制，讓演員不斷改變自己的位置座標，以達到沿路徑移動的效果，並利用 `repaint()` 來重繪改變後的結果，直到演員路徑結束才算此段劇情演出結束，才能換下一段劇情的演出。

■ 同時演出 (Parallel) :

同樣利用迴圈的控制，讓同時演出的演員們一起改變位置座標，以達到這些演員同時沿路徑移動的效果，並利用 `repaint()` 來重繪改變後的結果，直到所有演員的路徑都結束才算此段劇情演出結束，才能換下一段劇情的演出。

5.1.5 轉譯器各模組間的流程設計

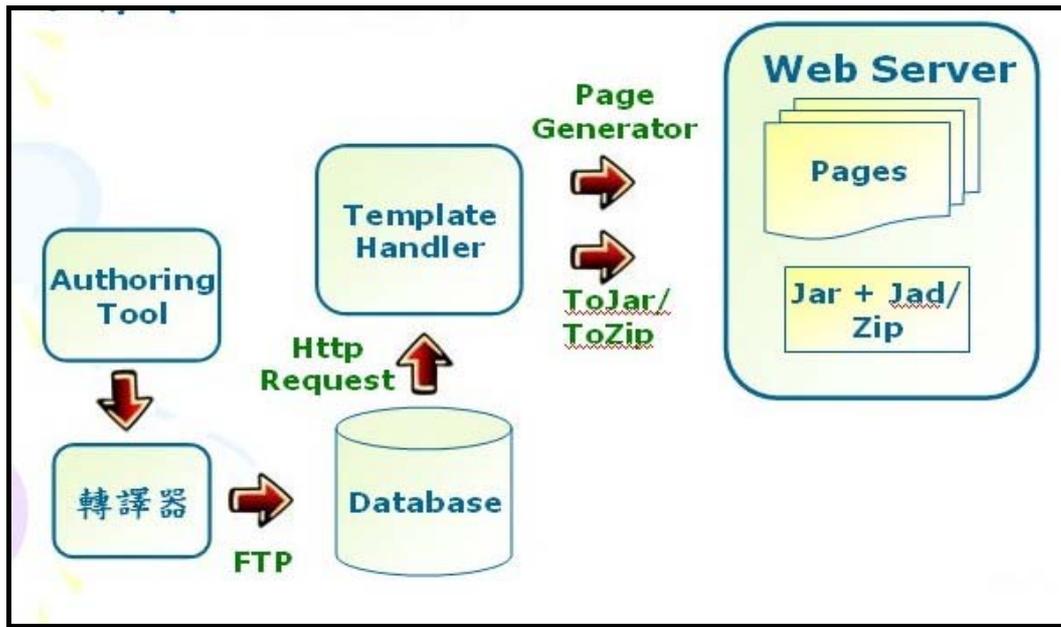
在介紹完轉譯器的架構 (4.2) 以及各部份的設計細節後，接著我們來看轉譯器各個模組之間的流程是怎麼設計的，下圖為模組間的流程圖：



圖二十九 轉譯器模組流程圖

一開始會先將目標語言宣告及標頭檔的部份先寫到檔案中，此部份的程式碼是固定的，因此並沒有一個特殊模組來處理，接著依序是互動劇情處理、開場劇情處理以及演員屬性設定三個模組來處理相對應的程式碼，最後再寫入程式中其他必需的程式碼，此部份的程式碼也是固定的，因此也不需要由一個模組來處理。

當程式碼均轉換完畢後便讓使用者選擇是否上傳到 Web Server 上，若使用者選擇不上傳，則會呼叫編譯工具使用模組，直接產生執行檔；若使用選擇要上傳，則呼叫上傳機制模組，由上傳機制模組來處理接下來的流程，下圖即為上傳機制模組之流程圖。



圖三十 樣板上傳機制架構圖

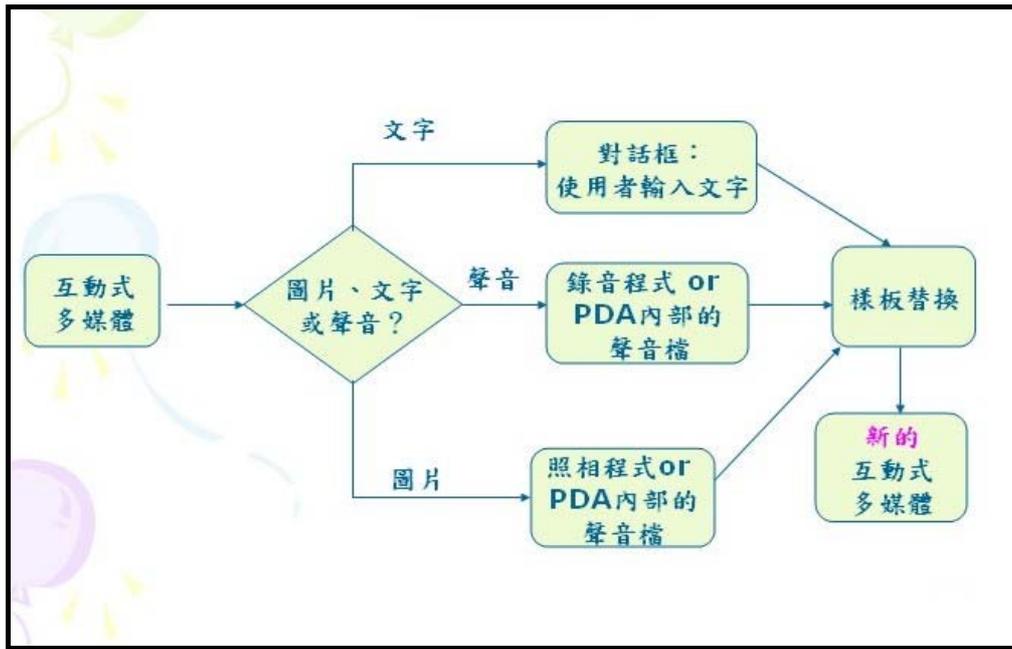
上傳機制是當 PC 上的視覺化編輯工具編好的內容經過轉譯器的轉換後，以 FTP 的方式上傳到 Server 上的資料庫中，接著再傳送一個 Http Request 給樣板機制處理器，告訴它樣板已上傳完成以及樣板名稱為何，再由樣板機制處理器來產生出此樣板的相對應置換網頁以及修改首頁的內容，讓使用者在首頁可以看到此樣板及超連結，此外，還必須將這個樣板轉成 Jar 檔或 exe 檔，以供其他使用者下載。

5.2 PDA 樣板系統之設計與實作

由 4.3.2 PDA 樣板系統模組的介紹，我們可以將 PDA 的樣板系統分為兩大部份來設計和實作—樣板機制及電子郵件發送中心，樣板機制主要是做樣板置換的工作，而電子郵件發送中心則是處理發送電子郵件的工作，接下來的兩小節就分別介紹這兩大部份的詳細細節。

5.2.1 樣板機制

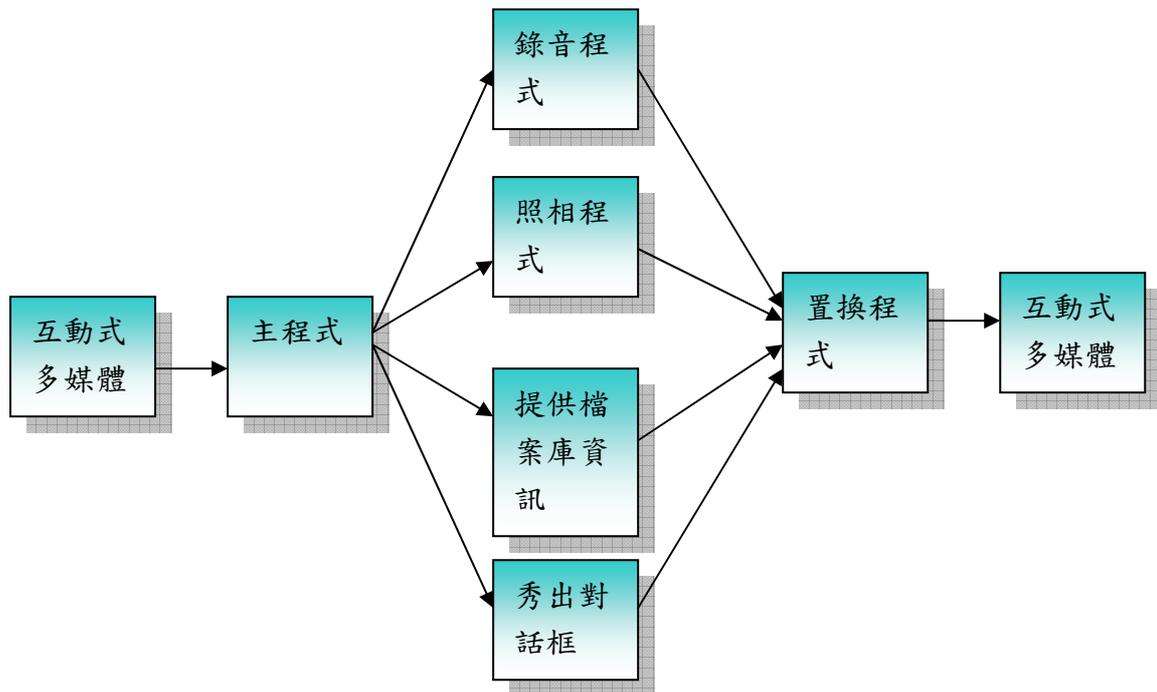
首先，我們來看樣板置換的流程，如下圖所示：



圖三十一 PDA 樣板置換流程圖

使用者在播放互動式多媒體時，若進入了樣板模式，則可以開始選擇要被置換的資源檔，若選擇了文字，則系統會跳出一個對話框，讓使用者可以輸入文字，接著做樣板中文字的取代，然後播放置換成功的新的互動式多媒體；若選擇了聲音，則系統會跳出一個對話框，讓使用者選擇要從錄音程式或是 PDA 內部的檔案庫來取得聲音檔，若是錄音程式，則系統會讓使用者錄一段聲音檔，若是 PDA 內部檔案庫則會讓使用者從檔案庫中選取一個聲音檔，接著做樣板中聲音檔的取代，然後播放置換成功的新的互動式多媒體；若選擇了圖片，則系統亦會跳出一個對話框，讓使用者選擇要從照相程式或是 PDA 內部的檔案庫來取得圖片檔，若是照相程式，則系統會讓使用者照一張相，若是 PDA 內部檔案庫則會讓使用者從檔案庫中選取一個圖片檔，接著做樣板中圖片檔的取代，然後播放置換成功的新的互動式多媒體。

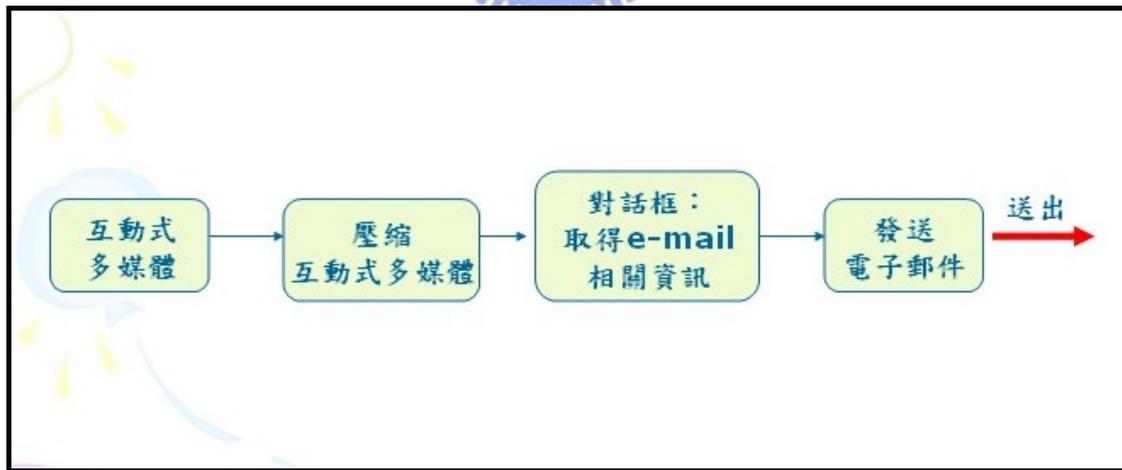
基於以上所敘述的樣板置換流程，我們做以下程式流程的設計：



圖三十二 PDA 樣板機制程式流程圖

5.2.2 電子郵件發送中心

電子郵件發送中心主要功能就是要將互動式多媒體以附加檔案的方式送出，而送出的流程如下圖所示：

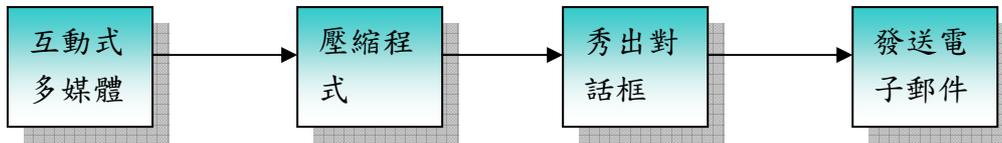


圖三十三 電子郵件發送流程圖

使用者在播放互動式多媒體時，若選擇了發送電子郵件的選項，則可以將目前正在播放的互動式多媒體發送出去，無論是置換前或是置換後均可以使用此項功能。在選擇發送電子郵件後，因為 PDA 上的互動式多媒體除了一個 exe 檔，還包含了此執行檔必需的多媒體資源檔，因此系統會先將整個互動式多媒體做壓縮，使其成為一個 Zip，以便做為電子郵件的附加檔案，接著再出現對話框，取

得電子郵件位址及郵件主旨，最後再送出。

基於以上所敘述的電子郵件發送流程，我們做以下程式流程的設計：



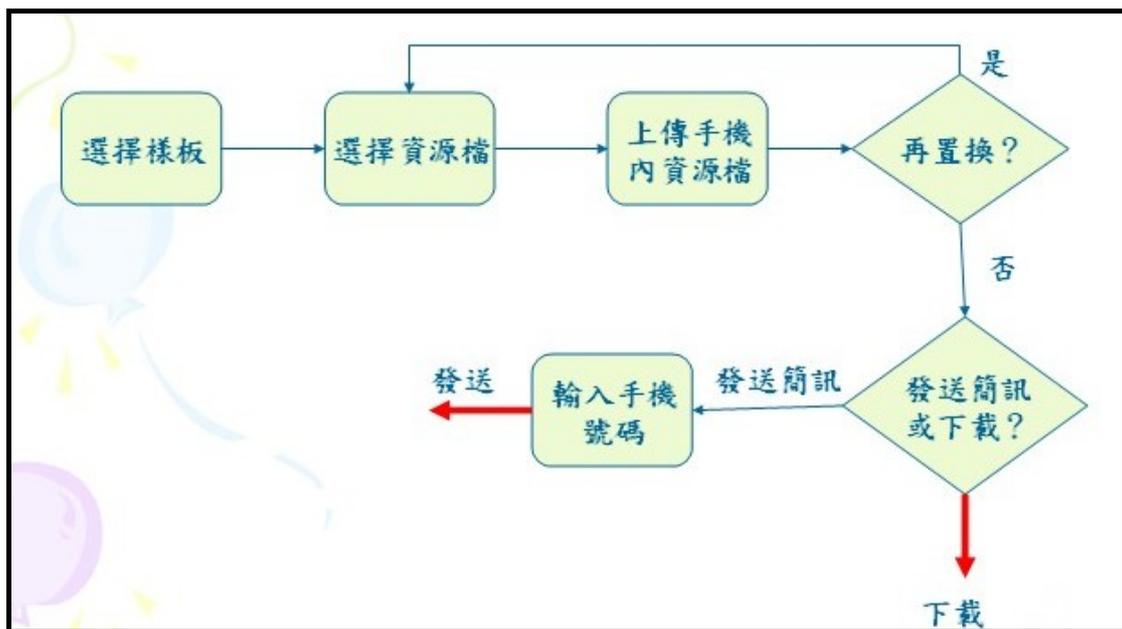
圖三十四 電子郵件發送中心程式流程圖

5.3 手機樣板系統之設計與實作

由 4.4.2 手機樣板系統模組，我們同樣可以將手機的樣板系統分為兩大部份來設計和實作—樣板模組及簡訊發送中心，樣板模組主要是做樣板置換的工作，而簡訊發送中心則是處理發送文字簡訊的工作。和 PDA 樣板機制系統最大的差別在於手機的樣板機制在樣板網站上進行，因此使用者必須透過手機內的瀏覽器來和樣板模組做溝通，資源檔必須要先上傳到樣板網站上才能做置換的動作，置換好的互動式多媒體也必須下載到手機才能觀看到結果。接下來的兩小節就分別介紹這兩大部份的詳細細節。

5.3.1 樣板模組

首先，我們先來看手機樣板置換的流程，如下圖所示：

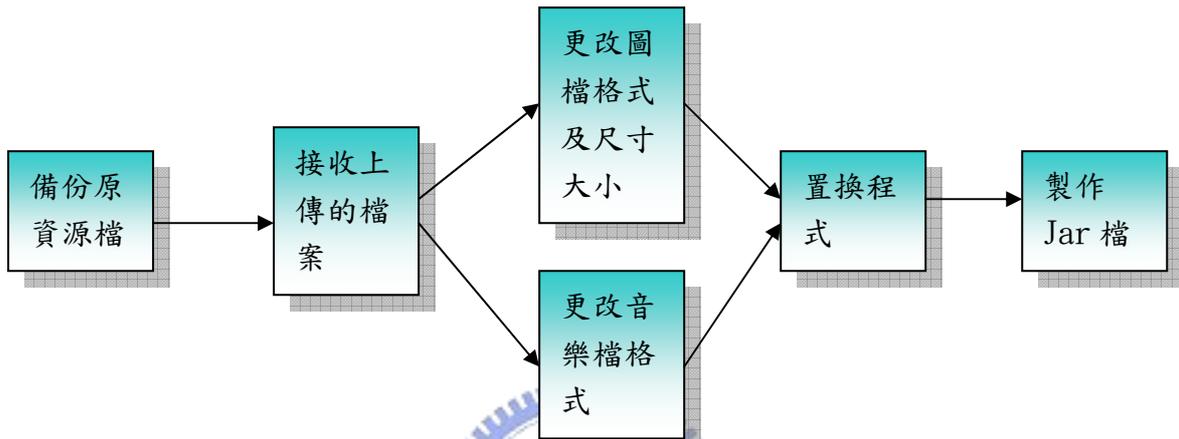


圖三十五 手機樣板置換流程圖

使用者連到手機樣板網站後先選擇想要置換的樣板，接著系統會列出該樣板

中可被置換的資源檔，包括背景圖、圖片、聲音、背景音樂等等，接著當使用者選擇了要被取代的資源檔後，系統會提供上傳的畫面，讓使用者可以上傳自己喜歡的資源檔，接著再做樣板置換的動作。使用者可以置換樣板中的一個或多個資源檔，直到不想再置換為止，接著系統會讓使用者選擇要下載回原手機還是由系統代發文字簡訊告知其他使用者這個互動式多媒體樣板的資訊，若選擇了發送簡訊，系統會先詢問對方的手機號碼，最後把文字簡訊發送出去。

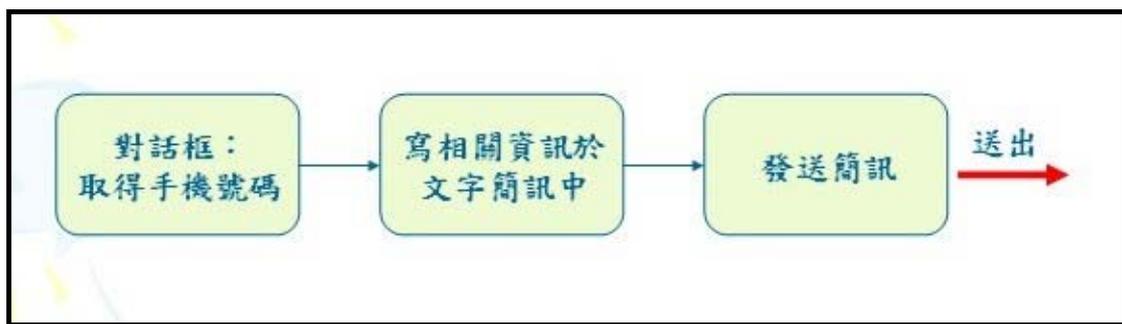
基於以上所敘述的樣板置換流程，我們做以下程式流程的設計：



圖三十六 手機樣板模組程式流程圖

5.3.2 簡訊發送中心

簡訊發送中心主要功能就是要將使用者置換好的互動式多媒體的相關訊息以文字簡訊的方式告知其他人，而送出的流程如下圖所示：



圖三十七 簡訊發送流程圖

在使用者選擇要發送簡訊後，系統會出現對話框讓使用者填入對方的手機號碼，接著再將這個互動式多媒體的 URL 寫入文字簡訊中，最後將此文字簡訊送出。若使用者對這個互動式多媒體有興趣，再自行根據文字簡訊中的 URL 到網站上下載來觀看。

從 PC 發送簡訊給手機的作法大致有兩種，一是與電信業者合作，取得發送

簡訊的權利，如奇摩簡訊或是各電信業者自己的網站，都有代發簡訊的功能，且發送費用是由使用者指定的手機來付費，且接收端收到的簡訊是顯示使用者的手機電話號碼；另一個作法是將一隻手機連接到電腦，再透過 AT Command 來控制該手機，利用它發送簡訊，但發送費用是由與電腦連接的手機付費，且接收端收到的簡訊是亦是顯示這隻手機的電話號碼。因為本研究並未與電信業者合作，因此發送簡訊的方法是採用後者，亦即對手機下 AT Command。

指令的流程如下：

1. AT+CMGF=1 (設定簡訊內容為文字模式)
2. AT+CMGS="09xxxxxxxx" (接收簡訊者的手機號碼)
3. 簡訊內容，最主要為此互動式多媒體的 URL。
4. Ctrl + Z (結束簡訊內容)



六、實作範例

在本章中，我們將以呈現系統畫面的方式，介紹系統中所擁有的功能、機制及其流程。

6.1 轉譯器

PC 上的視覺化編輯工具為智勝國際的編輯手，當使用者編輯完畢後只要按個按鈕即可做轉換，以下先展示編輯手的編輯環境：



圖三十八 智勝國際編輯手編輯環境

利用編輯手將互動式多媒體內容編輯完成後，可以先觀看播放成果，確定沒有問題後再轉成適用於 PDA 及手機的互動式多媒體，我們先來看看 PC 上的播放成果，在接下來的兩小節中再分別呈現此互動式多媒體在手機及 PDA 上的效果。



圖三十九 互動式多媒體內容在 PC 上呈現的效果

6.1.1 轉譯給 PDA

接著我們來看轉譯到 PDA 後的效果。



圖四十 互動式多媒體內容在 PDA 上呈現的效果

6.1.2 轉譯給手機

接著我們來看轉到手機後的效果。



圖四十一 互動式多媒體內容在手機上呈現的效果

6.1.3 上傳機制

使用編輯手編輯好後的互動式多媒體內容，可視為一個樣板並上傳至樣板網站上供其他使用者參考使用，下圖即為上傳過程的示意圖，使用者需先輸入樣板網站的 IP 及 Port，接著按上傳即可。畫面上會出現目前正在上傳的檔案名稱，並在上傳完成後顯示「上傳完成」。



圖四十二 上傳機制示意圖

上傳完成後，使用者便可在樣板網站上看到自己傳上去的互動式多媒體樣板，下圖中的右下角那個樣板即為新上傳的互動式多媒體樣板。



圖四十三 樣板網站內容

6.2 PDA 樣板系統

PDA 樣板置換的操作是在 PDA 上完成的，以下就分為兩個部份來說明，第一部份為使用者 A 開啟了 PDA 中的互動式樣板，並且將背景圖以即時照相的相片置換後將此互動式多媒體送給使用者 B；第二部份為使用者接收了使用者 A 傳送過來的電子郵件，並觀看此互動式多媒體。

1) PDA 使用者 A 開啟互動式多媒體



(1) 開啟互動式多媒體
並選擇樣板模式



(2) 進入樣板模式
並選擇背景圖



(3) 選擇從照相程式來取得檔案



(4) 取好景以後按下照相鍵



(5)背景圖置換完畢
選擇 Send Mail



(6)發送電子郵件之前
必須先壓縮檔案



(7)輸入信件主旨及對方電子
郵件信箱位址



(8)電子郵件發送完畢

圖四十四 PDA 使用者 A 使用 PDA 樣板機制流程圖

2) PDA 使用者 B 收到使用者 A 的電子郵件



(1)收到使用者 A 寄來的郵件
此郵件有一個附加檔案



(2)解壓縮此附加檔案



(3)解壓縮中...



(4)播放收到的互動式多媒體

圖四十五 PDA 使用者 B 接收到互動式多媒體之流程圖

6.3 手機樣板系統

手機樣板置換的操作是在遠端的樣板網站完成的，以下就分為兩個部份來說明，第一部份為使用者 A 連上了樣板網站，並且將背景圖換掉後再將此互動式

多媒體的相關資訊以簡訊送給使用者 B；第二部份為使用者接收了使用者 A 傳送過來的文字簡訊，並下載回手機安裝，接著觀看此互動式多媒體。



原本的互動式多媒體在手機上播放的效果。

以下的操作流程為使用者 A 連上網站後將此互動式多媒體的背景圖換掉，再傳送給使用者 B。

1) 手機使用者 A 連上網站後



(1) 使用者連上樣板網站



(2) 選擇要做置換的樣板



(3) 選擇背景圖



(4) 上傳資源檔



(5) 樣板置換成功

此時有以下三種選擇：

1. 繼續置換。
2. 自行下載到手機播放。
3. 傳送給其他人。



(6) 繼續置換，此時背景圖已被換過



(6) 輸入對方的手機號碼



(7) 傳送成功

圖四十六 手機使用者 A 使用手機樣板機制流程圖

2) 手機使用者 B 收到使用者 A 的簡訊



(1) 收到簡訊，內容為互動式多媒體的 URL



(2) 下載並安裝



(3) 安裝完成



置換背景圖後的互動式多媒體，也就是使用者 B 在安裝後看到的互動式多媒體內容

圖四十七 手機使用者 B 接收到互動式多媒體之流程圖

七、結論

7.1 總結

本系統提供了一個更簡單方便的方法，讓使用者只要利用 PC 上的視覺化編輯工具或是適用於各手持行動裝置的樣板機制，便可以輕易地製作自己喜愛的手持行動裝置上的互動式多媒體。

主要特點：

- (1) 轉譯器的存在，免去寫程式的步驟：

透過轉譯器的概念，使用者不用親自撰寫程式碼，因此無論使用者是否具備程式背景，只要學會操作 PC 上的視覺化編輯工具，就可以輕鬆地做出適用於手持行動裝置的互動式多媒體。

- (2) 可直接在 PC 上觀看互動式多媒體的內容呈現：

使用者使用 PC 上的視覺化編輯工具製作，當製作完成後可在 PC 上先播放互動式多媒體的內容，觀看此互動式多媒體呈現的效果是不是自己所喜愛的，等確定內容以後再做轉換的工作，如此一來，可以避免為了觀看互動式多媒體在手持行動裝置上的成果而不斷重覆地在 PC 和手持行動裝置之間傳送的动作。

- (3) 樣板機制的規劃，製作手持行動裝置上的互動式多媒體更省時：

因為製作好的樣板已經將內容的呈現模式以及風格均規劃處理好了，只需要使用者做置換圖檔或聲音檔的部份，因此不僅提供使用者另外一個途徑去製作手持行動裝置上的互動式多媒體，此外，有些使用者常常不知道內容該如何從無到有，因此，亦能解決了使用者不知從何下手的麻煩。

- (4) 提供電子郵件及簡訊發送服務：

本系統在手機的樣板系統中提供了簡訊發送服務，在 PDA 的樣板系統中提供了電子郵件發送服務，使用者不僅可以自己享受製作互動式多媒體的樂趣，亦可將自己的成品與朋友分享，提高了手持行動裝置上的互動式多媒體之趣味性。

7.2 未來發展方向

在此我們提出一些對未來研究、發展方向的建議。

- (1) 目前手機上網方式仍為 GPRS 居多，不僅連結速度慢，也不夠穩定，此外手機上瀏覽器的支援仍很陽春，對於 Java Script 等動態網頁顯示的支援還不夠完整，因此手機樣板機制的設計仍不夠友善，期望當手機可以無線網路來上網後，且功能更強大以後，樣板機制的操作可以更加簡單。
- (2) 數位內容必須要被保護，因此安全性的問題需再加強，提供認證系統或加密的功能將是未來的趨勢。
- (3) 目前本系統所製作出來的互動式多媒體及樣板機制限用於 OS 為 Microsoft Windows CE 的 PDA，因此希望未來此系統可以支援 Linux 或其他作業系統的 PDA，使得互動式多媒體的應用更加廣泛。
- (4) 由於 PC 的螢幕畫面較大，而手機和 PDA 的螢幕畫面較小，因此在轉換的過程中需要把圖片及文字做一定比例縮小，如此可能會造成在 PDA 或手機上播放時，有些部份的內容太小或是看不清楚的情況，此外，也可能因為 PDA、手機和 PC 的螢幕長寬比例不盡相同，導致互動式多媒體內容中物件的相對位置會改變，因此未來可以在 Context-awareness 及 Content adaptation 兩方面更深入去研究，使得編輯出來的內容能更適合於 PDA 和手機。

參考文獻或資料

- [1] Edouard Ma Poon, "The Design and Implementation of Dynamic Web-Based Multimedia Templates for Empowering Users of Wireless Pervasive Computation Devices", N.C.T.U Taiwan, dissertation, 2004
- [2] Shang-Ting Yang, "User look & Feel Design for Handset Devices Based on Visual Requirement Authoring and Program Generation Methodology", N.C.T.U Taiwan, dissertation, 2004
- [3] Chorng-Shiuh Koong, "A Component-based Visual Scenario construction environment for non-programming users to create interactive electronic books", N.C.T.U Taiwan, dissertation, 2000
- [4] Maria Virvou, Eythimios Alepis, "Mobile educational features in authoring tools for personalized tutoring", Department of Informatics, University of Piraeus, 2003
- [5] 王森, Java手機程式設計入門, 知城數位科技股份有限公司, 民國九十年
- [6] 楊先民, Pocket PC程式設計自學手冊, 學貫行銷股份有限公司, 民國九十一年
- [7] 江義華, Java Phone J2ME MIDP行動通訊程式設計, 金禾資訊, 民國九十一年
- [8] Bruce E. Krell, Pocket PC Developer' s Guide, 2002
- [9] Steve Makofsky, Pocket PC Network Programming, 2004
- [10] MIDP2.0 spec
- [11] 智勝國際, <http://www.caidiy.com/caidefault.htm>
- [12] <http://weco.net/course/wi/J2ME.pdf>
- [13] 拓璞產業研究所, <http://www.topology.com.tw/TRI/default.asp>
- [14] Delphi K.Top, <http://delphi.ktop.com.tw/>
- [15] SourceFORGE.net, <http://sourceforge.net/>
- [16] Sun, <http://java.sun.com/j2me/>

- [17] eMbedded Programming , <http://www.cj.idv.tw/>
- [18] Virtual Office Systems , <http://www.voscorp.com/>
- [19] Tek-Tips Forums , <http://www.tek-tips.com/>
- [20] The Code Project , <http://www.codeproject.com/>
- [21] 利用Java 撰寫PDA應用程式－王森 ,
<http://www.netmag.com.tw/member/article/010307c.htm>
- [22] 中華電信emome , <http://www.emome.net/>
- [23] 遠傳i-style , <http://www.istyle.com.tw/>
- [24] OKWAP , <http://www.okwap.com/>

