

國立交通大學

資訊工程學系

碩士論文

支援 MPEG-4 標準的視覺化編輯工具



MPEG-4 Creation Based On A Visual Authoring
System

研 究 生：周宜靖

指 導 教 授：陳登吉 教授

中 華 民 國 九 十 四 年 七 月

支援 MPEG-4 標準的視覺化編輯工具

MPEG-4 Creation Based On A Visual Authoring System

研究生：周宜靖 Student：Yi-Jing Chou

指導教授：陳登吉 Advisor：Deng-Jyi Chen

國立交通大學
資訊工程學系
碩士論文

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

支援 MPEG-4 標準的視覺化編輯工具

學生：周宜靖

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

MPEG-4 是 MPEG 組織所發表關於互動式多媒體的標準。MPEG-4 標準具有物件化、低位元率、檔案壓縮率高等等的優點，尤其是 MPEG-4 對互動方面的支援，更是其他類型多媒體所不足的地方。

目前 MPEG-4 檔案的產生方式主要有兩種，一種是手動撰寫 MPEG-4 場景描述語言，另一種是透過現有的 MPEG-4 編輯工具來輔助製作 MPEG-4 檔案，但這兩種方式都必須瞭解 MPEG-4 場景概念，技術門檻並不低，因此較適合程式設計師來使用。對 MPEG-4 需求者來說，想要製作 MPEG-4 檔案，可以學習 MPEG-4 場景概念來開發製作，但此方法對沒有程式背景的人有一定的困難度，因此並不實際，而另一個方法便是讓程式設計師代為製作 MPEG-4 檔案。所以目前並沒有一個簡單容易的 MPEG-4 檔案製作方式。

因此，本研究的動機就是針對 MPEG-4 需求者不易製作 MPEG-4 檔案的缺點，提供改進的方法。與其讓程式設計師”代工”製作 MPEG-4 檔案，不如發展一套適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具，讓 MPEG-4 需求者隨心所欲的去修改 MPEG-4 場景的內容，克服 MPEG-4 需求者開發製作 MPEG-4 的問題，降低製作的門檻，如此，將可提高 MPEG-4 的接受度。

MPEG-4 Creation Based On A Visual Authoring System

Student : *Yi-Jing Chou*

Advisor: *Dr. Deng-Jyi Chen*

Department of Computer Science and Information Engineering

National Chiao Tung University

Abstract

MPEG-4 is a current standard for video encoding and presentation announced by MPEG organization. It adopts the object-oriented concept. Low bit-rate and high file compression ratio are its major advantage with supporting of object interaction.

Currently, there are two ways to create MPEG-4 files, one is to write the script language that can encode to MPEG-4, and the other way is to use some existing MPEG-4 authoring tools. These two ways, however, all need to master the conception of MPEG-4 scene description. Therefore, only some very technical users group can create MPEG-4 files. Thus, these approaches are suitable for users who have programming-skill. For users who want to produce MPEG-4 files with no good programming skill, the study of MPEG-4 description language is quite difficult for them. Eventually, they usually need MPEG-4 programmers to help them. There are no easy ways to create MPEG-4 files.

In this thesis research, we proposed an MPEG-4 creation based on a visual authoring tool that is suitable for users with no programming background. Users can create MPEG-4 files by themselves using only drag and drop approach. Specifically, we designed and implemented a translator that takes the script language generated from an existing visual authoring tool to BIFS script language. With the audio and video encoders, an MPEG-4 file can be created for the presentation under the MPEG-4 player. Application examples are used to demonstrate the applicability and feasibility of the proposed approach.

誌謝

本論文多蒙指導老師陳登吉教授的耐心指導及多方教誨，得以順利完成，並讓我在做研究的方法及心態上有所增進，在此致上無限的謝意。

此外，感謝所有幫助我的朋友們，尤其是實驗室的學長、同學、學弟妹，他們在我論文的製作期間，給予了我很多寶貴的意見與協助，也帶給我許多的歡笑及難忘的回憶。

最後，我要感謝我的家人，尤其是養育我、栽培我的父母，在我求學的路上一直默默的支持我，讓我全力以赴完成學業與論文，在此獻上由衷的感謝。



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 MPEG-4 系統簡介.....	1
1.1.1 MPEG-4 標準.....	1
1.1.2 MPEG-4 的特色.....	1
1.2 製作 MPEG-4 檔案所遭遇的問題.....	2
1.2.1 以描述語言製作 MPEG-4 檔案.....	2
1.2.2 MPEG-4 需求者與程式設計師的溝通.....	4
1.2.3 測試與修改.....	5
1.3 動機、研究方法與步驟.....	5
1.3.1 動機.....	5
1.3.2 研究方法.....	6
1.3.3 研究步驟.....	6
1.4 章節架構.....	7
二、相關研究與探討.....	8
2.1 MPEG-4 場景概念.....	8
2.1.1 場景描述架構.....	8
2.1.2 物件描述子架構.....	9
2.1.3 整合場景描述與物件描述子.....	10
2.2 MPEG-4 場景描述語言.....	11
2.2.1 BIFS.....	11
2.2.2 BT script.....	12
2.2.3 XMT.....	12
2.2.4 分析比較.....	12
2.3 現有的 MPEG-4 編輯工具.....	14
2.4 解決方案.....	16
三、系統需求分析.....	17
3.1 視覺化編輯工具的選擇.....	17
3.1.1 選擇的依據.....	17

3.1.2 選擇的結果	18
3.2 轉譯器功能需求	19
3.3 MPEG-4 場景描述語言的選擇	22
3.4 整體分析	23
四、 系統設計與實作	25
4.1 系統架構	25
4.1.1 MPEG-4 編輯工具與 MPEG-4 播放器	25
4.1.2 視覺化編輯工具與轉譯器的結合	26
4.1.3 轉譯器模組概觀	27
4.1.4 轉譯器系統程式流程	28
4.2 轉譯器設計	29
4.2.1 使用者介面模組	29
4.2.2 Script 分析模組	30
4.2.3 場景處理模組	31
4.2.4 多媒體資料處理模組	33
4.2.5 MPEG-4 壓縮模組	35
4.3 轉譯器設計 - 場景處理模組	35
4.3.1 場景訊息子模組	35
4.3.2 靜態顯示子模組	37
4.3.3 互動劇情子模組	41
4.3.4 進階設定子模組	44
4.3.5 XML to BT 子模組	46
五、 系統展示	48
5.1 操作流程	48
5.2 大小比較	54
六、 結論	55
6.1 本研究特點	55
6.2 未來發展	56
參考文獻	58

表目錄

表 1	MPEG-4 encoder比較	23
表 2	劇情的處理方式.....	41
表 3	編輯手story與轉換後的MPEG-4 大小之比較.....	54
表 4	編輯手story與轉換後的MPEG-4 大小之比較--直方圖.....	54



圖目錄

圖 1	文字的 BT script 語法及其顯示的效果	3
圖 2	資策會多媒體實驗室的 MPEG-4 編輯工具	4
圖 3	MPEG-4 需求者與程式設計師	5
圖 4	適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具	6
圖 5	傳統視訊影片的組成概念圖	9
圖 6	BIFS Scene 的組成概念圖	9
圖 7	多媒體物件與物件描述架構對應圖	10
圖 8	BIFS 程式架構—以 BT script 為例	11
圖 9	以 XMT-O 語法表示圖片	12
圖 10	圖片之呈現	13
圖 11	以 BT script 語法表示圖片	13
圖 12	以 XMT-A 語法表示圖片	14
圖 13	Digimax 的 MAXPEG Author	15
圖 14	資策會多媒體實驗室的 E-Learning Authoring Tool	16
圖 15	適合於沒有撰寫程式專業能力使用者的 MPEG-4 製作流程圖	16
圖 16	智勝國際編輯手	19
圖 17	轉譯器概觀	20
圖 18	轉譯器五大模組	22
圖 19	系統需求分析後的 MPEG-4 製作流程	24
圖 20	MPEG-4 編輯工具與 MPEG-4 播放器	26
圖 21	視覺化編輯工具與轉譯器的結合	27
圖 22	轉譯器模組樹狀關係	27
圖 23	轉譯器系統程式流程	28
圖 24	使用者介面 — 選擇輸出目錄與輸出類型	29
圖 25	使用者介面 — 顯示轉換進度	29
圖 26	編輯手 Script 分析流程	30
圖 27	各類訊息的流向	31
圖 28	場景處理模組流程	32
圖 29	子模組多次寫檔造成效率低落	33
圖 30	以 XML 元件提高效率	33
圖 31	聲音格式轉換流程	34
圖 32	視訊格式轉換流程	34

圖 33	圖片格式轉換流程.....	35
圖 34	MPEG-4 壓縮流程.....	35
圖 35	Ratio 變數對場景的影響	36
圖 36	初始化 BT script 架構	37
圖 37	靜態顯示子模組流程.....	38
圖 38	三種不同類型物件的 BT script 程式架構	39
圖 39	物件描述子 BT script 程式架構	40
圖 40	劇情訊息樹狀圖.....	41
圖 41	編輯手描述劇情的例子.....	41
圖 42	劇情的 BT script 程式碼概念圖	42
圖 43	物件播放 BT script 程式碼概念圖	43
圖 44	加入演出前初始化區塊、演出後消失區塊.....	45
圖 45	修改初始化區塊的 BT script 程式碼	45
圖 46	XML 格式轉換成 VRML 格式的 BT script	47
圖 47	編輯手—教材管理.....	48
圖 48	編輯手—場景設定.....	49
圖 49	編輯手—內容設定.....	50
圖 50	編輯手—劇情設定.....	50
圖 51	編輯手—編輯完成的結果.....	51
圖 52	編輯手—存成 MPEG-4 檔案.....	51
圖 53	轉譯器—使用者介面.....	52
圖 54	轉譯器—轉換進度之顯示.....	52
圖 55	PC 版本在播放器上之結果.....	53
圖 56	PDA 版本在播放器上之結果.....	53

一、緒論

1.1 MPEG-4 系統簡介

1.1.1 MPEG-4 標準

隨著網路環境的逐漸成熟與各類型多媒體的蓬勃發展，可以發現即時互動的應用越來越受到重視，但是此種即時互動的應用，需要媒體能以低位元率(low-bit rate)的方式作傳輸，才能適應於各種不同頻寬的網路環境中，環顧目前的各種多媒體的標準，MPEG-4 具有低位元率、互動性、壓縮率高等等的優勢，非常適合即時互動的應用。另外，數位電視以及其他需要互動式內容的應用，MPEG-4 標準也扮演者舉足輕重的地位〔3〕〔7〕〔18〕。

MPEG 組織由 1993 年開始制訂 MPEG-4 標準，在 1998 年出現第一個版本，到了 2002 年已經進入第三個版本，並且為 ISO/IEC 所採用。ISO/IEC 將 MPEG-4 定義為” Coding of audio-visual objects” 的技術〔1〕，有別於之前的多媒體標準(像是 MPEG-1、MPEG-2)，MPEG-4 捨棄傳統視訊壓縮的方式，改以物件化為基礎，企圖以這樣的技巧增加檔案的壓縮率，並達到具有互動能力、整合各類型多媒體、支援異質網路環境的功能。

所謂的物件化，就是將媒體資料當作是 Audio-Visual 物件，並具有自己的呈現方式、編碼方式、以及傳輸方式，也就是說，把所有的物件都分別地壓縮與編碼，並且用不同的串流達到傳輸的目的。另外，MPEG-4 把整個呈現的畫面視為這些 Audio-Visual 物件的組合，這樣一來，使用者可以操作不同的 Audio-Visual 物件，組合成不同呈現的場景，使得 MPEG-4 的畫面變得更豐富。

1.1.2 MPEG-4 的特色

具互動能力：

使用者對多媒體的需求，除了傳統視訊的播放外，更希望能有互動的能力，而以物件為基礎的 MPEG-4 標準，就提供了與使用者互動的能力。所謂的互動，指的就是多媒體在播放過程中，會根據使用者的操作給予回應，而不再是呆板的順序播放，這一點，是 MPEG-4 與其他家族成員(像是 MPEG-1、MPEG-2)不同的地方之一。

✚ 整合各類型多媒體：

MPEG-4 整合真實與虛擬的場景。像是圖片、聲音、影片這些真實世界的自然物件，或是利用 VRML 及其他方式繪圖所產生的合成物件，MPEG-4 的場景可以由這些不同性質的多媒體所組合而成，支援 2D 與 3D，打破儲存媒體的限制，因此，MPEG-4 場景的多樣性與豐富性亦是其重要特色。

✚ 支援異質網路環境：

在 MPEG-4 的標準中，並沒有定義 MPEG-4 串流必須在哪種網路環境下傳輸，取而代之的是定義 DMIF 的介面〔5〕，只要符合 DMIF 的介面的規範，MPEG-4 串流便可以再不同的網路環境下傳輸，具有跨越各種異質網路環境的能力。

✚ 場景描述語言描述劇情：

由於 MPEG-4 場景的組成十分複雜，因此在 MPEG-4 標準中以場景描述語言定義了物件彼此時間或空間上的關係，也就是場景劇情的表現。常見的 MPEG-4 場景描述語言有 BIFS、BT script、以及 XMT，其中 BIFS 是 binary format，而 BT script 與 XMT 是 textual format，都與 MPEG-4 場景概念有密切相關，詳細的場景概念會在 2.1 節說明。

1.2 製作 MPEG-4 檔案所遭遇的問題

1.2.1 以描述語言製作 MPEG-4 檔案

目前 MPEG-4 檔案的產生方式，大致上分成兩種類型，一種是手動撰寫 textual format 的 MPEG-4 場景描述語言，另一種是利用現有的 MPEG-4 編輯工具來輔助 MPEG-4 檔案的製作〔9〕，但這兩種方式都比較適合程式設計師。

所謂手動撰寫 textual format 的 MPEG-4 場景描述語言，也就是手寫 BT script 或是 XMT，再透過 MPEG-4 encoder 轉成 MPEG-4 檔案，這樣的方式必須對 BT script 或是 XMT 相當瞭解，知道如何運用描述語言所提供的能力去構成所需要的場景。另外，不管是用 BT script 或是 XMT，甚至是 BIFS，這些場景描述語言終究是架構在 MPEG-4 標準之下，因此想要撰寫這些場景描述語言，必須對 MPEG-4 場景概念有一定程度的瞭解才可達到。

利用手寫場景描述語言的方式製作 MPEG-4 檔案，可以發現幾個缺點。首先，從圖 1 可以知道，手動撰寫 textual format 場景描述語言是相當複雜的。對於一個場景的描述，若只是以文字編輯器來編輯，將會花很多時間來編寫描述檔，並且容易發生文法錯誤，在修改及維護時也相當的不方便。另一個缺點則是描述檔的量可能會很大，由於 MPEG-4 場景的複雜度通常來說比其他類型的多媒體大上許多，所以場景描述語言的量動輒數千行，甚至數萬行，對於程式設計師來說亦是一件頭痛的事。最後，這種手寫描述檔的方式並不適合沒有撰寫程式專業能力的使用者，具有程式背景的人才可以此種方式製作 MPEG-4 檔案。

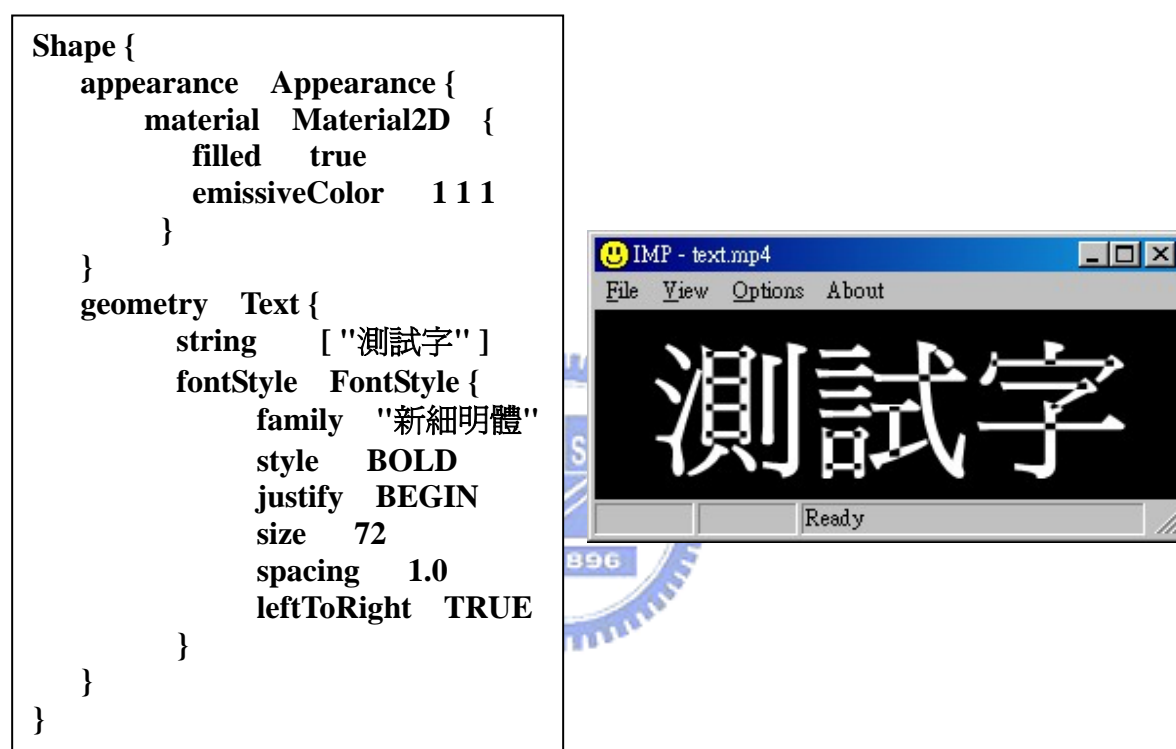
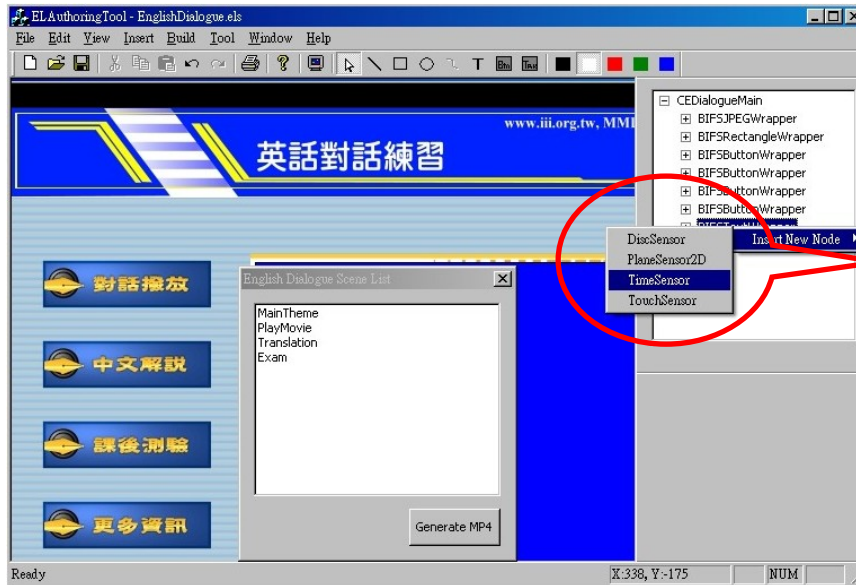


圖 1 文字的 BT script 語法及其顯示的效果

利用現有的 MPEG-4 編輯工具來輔助 MPEG-4 檔案的製作，可以使得製作 MPEG-4 檔案的過程變得容易，但是，目前這些輔助 MPEG-4 檔案製作的編輯工具，通常都不會隱藏 MPEG-4 場景的概念，也就是說，程式設計師需要先研習過 MPEG-4 的相關知識，明白 MPEG-4 場景架構的來龍去脈後，才能控制場景的編輯，雖然比手寫 textual format 場景描述語言簡單，不過這種方式對於沒有撰寫程式專業能力的使用者還是不容易，仍然比較適合程式設計師。

圖 2 說明雖然有 MPEG-4 編輯工具來輔助 MPEG-4 檔案的製作，但使用者仍須瞭解 MPEG-4 場景概念。此工具是資策會多媒體實驗室所發展的 E-Learning Authoring Tool [15]，在設定物件的屬性時，使用者必須知道 TimeSensor、TouchSensor 這些節點的意義，並不是完全隱藏住 MPEG-4 場景的特性。



使用者需要有 MPEG-4 場景的概念，例如使用者必須知道 TimeSensor、DiscSensor、TouchSensor 這些節點的使用法

圖 2 資策會多媒體實驗室的 MPEG-4 編輯工具

不論是手寫 textual format 場景描述語言，或是利用現有的 MPEG-4 編輯工具來輔助，我們可以發現這兩種方式共同的缺點就是必須瞭解 MPEG-4 場景概念，因此較適合程式設計師來使用，而對於沒有撰寫程式專業能力的使用者來說，這兩種製作 MPEG-4 檔案的方式並不適合。

1.2.2 MPEG-4 需求者與程式設計師的溝通

由 1.2.1 節的說明可以知道，製作 MPEG-4 檔案需要擁有程式背景，但是，MPEG-4 需求者不一定會具備程式背景。當一個不具備程式背景的 MPEG-4 需求者想要製作 MPEG-4 檔案時，必須以文字描述或畫圖或其他方式，描述 MPEG-4 場景的樣子，再經由與程式設計師的溝通，讓程式設計師代為製作 MPEG-4 檔案。但是由於 MPEG-4 需求者與程式設計師專業領域不同，因此描述 MPEG-4 場景的方式可能會有很大的出入，這樣一來，會造成程式設計師製作的 MPEG-4 檔案並不是 MPEG-4 需求者所想像的樣子，因此，兩者在溝通方面的問題不可輕忽。

另一方面，因為溝通不良導致 MPEG-4 檔案製作錯誤所造成的損失是很大的。一旦程式設計師製作的 MPEG-4 檔案不符合需求時，MPEG-4 需求者和程式設計師得再做一次溝通，讓 MPEG-4 檔案的製作重新開始。由此可知，為了製作 MPEG-4 檔案，MPEG-4 需求者與程式設計師必須在整個製作期間密切的溝通，隨時注意是否偏離需求，不然程式設計師會浪費許多時間在 MPEG-4 檔案的重製上，也會浪費更多的時間來重新確定需求，既延遲 MPEG-4 檔案製作進度又在溝通上浪費人力。

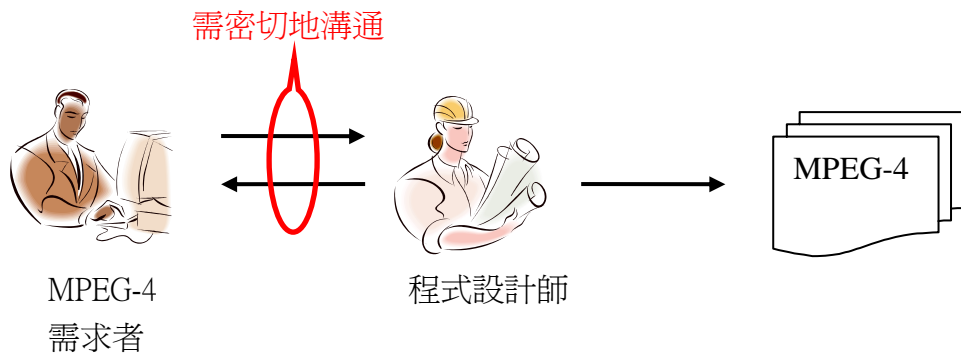


圖 3 MPEG-4 需求者與程式設計師

圖 3 是 MPEG-4 需求者與程式設計師的關係圖，MPEG-4 需求者勾勒出對 MPEG-4 檔案的需求後，交給程式設計師來執行製作，在這個過程中，兩者必須頻繁的溝通，來避免不必要的誤解。

1.2.3 測試與修改

MPEG-4 需求者對於 MPEG-4 檔案的需求經常會在程式設計師將 MPEG-4 檔案製作完畢後，看到呈現的樣子才會做修改。在程式設計師重新修改 MPEG-4 檔案時，MPEG-4 需求者只好耐心地等待檔案的完成，如果可以讓 MPEG-4 需求者直接參與 MPEG-4 檔案的編輯，不但可以減少與程式設計師的溝通，減少誤解的發生，也可以隨心所欲的去修改 MPEG-4 場景的內容，提高創作的空間及自由度。

1.3 動機、研究方法與步驟

1.3.1 動機

由 1.2 節的說明可以知道，目前 MPEG-4 檔案的產生方式主要有兩種，一種是手動撰寫 MPEG-4 場景描述語言，另一種是透過現有的 MPEG-4 編輯工具來輔助製作 MPEG-4 檔案，但這兩種方式都必須瞭解 MPEG-4 場景概念，技術門檻並不低，因此較適合程式設計師來使用。對 MPEG-4 需求者來說，想要製作 MPEG-4 檔案，可以學習 MPEG-4 場景概念來開發製作，但此方法對沒有程式背景的人有一定的困難度，因此並不實際，而另一個方法便是讓程式設計師代為製作 MPEG-4 檔案。所以目前並沒有一個簡單容易的 MPEG-4 檔案製作方式。

因此，本研究的動機就是針對 MPEG-4 需求者不易製作 MPEG-4 檔案的缺點，提供改進的方法。與其讓程式設計師”代工”製作 MPEG-4 檔案，不如發展

一套適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具，讓 MPEG-4 需求者隨心所欲的去修改 MPEG-4 場景的內容，克服 MPEG-4 需求者開發製作 MPEG-4 的問題，降低製作的門檻，如此，將可提高 MPEG-4 的接受度。

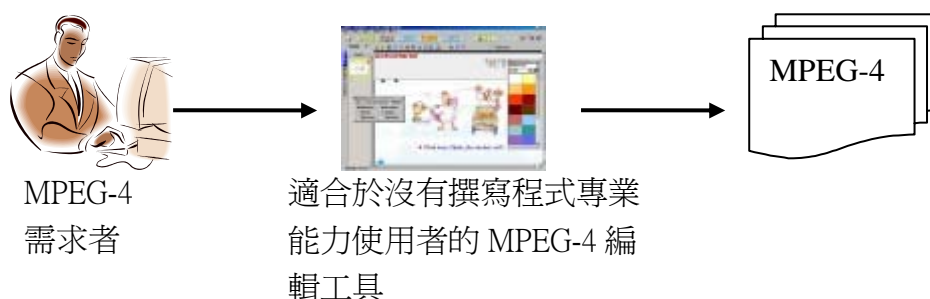


圖 4 適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具

本研究的目標，如圖 4 所示，就是發展一套適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具，來產生 MPEG-4 檔案。MPEG-4 需求者直接編輯 MPEG-4 檔案，可以減少與程式設計師溝通上的困難，提高編輯的效率。

1.3.2 研究方法

目前視覺化編輯工具的發展隨著電腦的蓬勃發展而有長足的進步。其中，適合於沒有撰寫程式專業能力使用者的視覺化編輯工具，它具有親和易上手的使用者介面，在操作上與使用者的直覺反應很接近，因此使用者不需有程式背景就能輕鬆的使用。本研究找尋一個視覺化編輯工具，此工具不僅能適合於沒有撰寫程式專業能力的使用者來使用，並且又能符合 MPEG-4 特性，這樣一來，利用它來產生 MPEG-4 檔案就會變得直覺化、簡單化，就算是完全不懂 MPEG-4 場景概念的使用者，也可以藉由此視覺化編輯工具製作出自己的互動式多媒體。

爲了讓此視覺化編輯工具能產生 MPEG-4 檔案，本研究撰寫一個適合於視覺化編輯工具的轉譯器，讓使用者先以視覺化編輯工具編輯互動式多媒體後，再透過轉譯器轉爲 MPEG-4 檔案，隱藏 MPEG-4 場景概念，減低使用者編輯互動式多媒體的負擔，增加製作的效率。

1.3.3 研究步驟

爲了達到上述的目標，本論文以下列的步驟進行研究：

(1) 探討 MPEG-4 場景概念

首先，本論文針對 MPEG-4 場景的特性與概念，做進一步的分析與探討，明瞭 MPEG-4 的特性後，可以得知 MPEG-4 場景的基本架構與物件之間的關連。

(2) 選擇適合的視覺化編輯工具

此視覺化編輯工具要適合於沒有撰寫程式專業能力的使用者，並且符合 MPEG-4 的特性。

(3) 分析轉譯器需求

轉譯器是產生最終 MPEG-4 檔案的媒介，利用轉譯器才能轉換成 MPEG-4 的格式。

(4) 根據需求實作出適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具

整合視覺化編輯工具與轉譯器，成爲一套適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具。

(5) 利用這套系統製作 MPEG-4 檔案

1.4 章節架構

第一章爲緒論，先對 MPEG-4 標準做概略性的介紹，接著敘述製作 MPEG-4 所遭遇的問題，再描述本研究的動機、研究方法與步驟。

第二章爲相關研究與探討，針對 MPEG-4 標準的場景架構與描述語言，做相關的研究，並比較目前的 MPEG-4 編輯工具，最後針對它們的缺點提出本研究改進的方法。

第三章爲系統需求分析，根據本研究提出的改進方法，分析所需要的系統，並規劃系統的輪廓。

第四章爲系統設計與實作，將第三章分析完成後的系統需求，做更進一步的設計，並真正的實作出來，包括視覺化編輯工具及轉譯器的設計細節以及實作模組都在此章介紹。

第五章爲系統展示，將完成後的系統，以範例的方式說明操作的流程。

第六章爲結論，總結出本研究的特點以及未來的發展。

二、相關研究與探討

2.1 MPEG-4 場景概念

MPEG-4 系統的場景以 Audio-Visual 物件來構成，爲了表達場景的意義，除了傳遞 Audio-Visual 物件的實體資料流外，也傳遞一些額外的訊息，這些訊息可以貼切地表示各個物件之間的空間與時間的關係，也就是場景劇情的描述，這些用來描述物件關連的訊息就稱之爲場景描述(Scene Description) [1][4][5][6]。MPEG-4 系統的場景描述以場景描述語言來表示，但是除了場景描述外，另外還需要物件描述子(Object Descriptor)才能呈現完整的場景，因此在 MPEG-4 中，定義了物件描述子架構(Object Descriptor Framework) [1][4][5][6]，用來連結劇情資訊與 Audio-Visual 物件的實體資料流，也就是說，我們從場景描述語言中取出劇情資訊，得知每個物件要演出的戲碼，而關於每個物件的實體資料流的位置，則需透過物件描述子的部分，去取得物件實體資料流位置的關連，經由這樣的過程，便可以連結物件，重新展示出場景的原貌。

2.1.1 場景描述架構

MPEG-4 的場景描述，主要是以 VRML [11] 的概念作延伸，因此是以 Node、Field 爲組成場景的基本單位，利用這些 Node、Field，場景的組成就像是堆積木般，由不同的 Node 以樹狀結構建構而成，至於場景劇情的控制，MPEG-4 則提供了三種類型的 BIFS command(Replace, Insert, Delete)，透過這些命令來對 Node 或 Field 做取代、插入、刪除的動作，最後，MPEG-4 亦保留 VRML 傳遞事件訊息的方式，以 ROUTE 替不同的 Node 之間傳遞所需的訊息。

MPEG-4 的場景我們亦可稱爲 BIFS Scene，它的概念源自於傳統視訊影片的播放 [8]，傳統視訊影片可以看成是圖片連續播放的結果，如圖 5 所示，影格(frame)與相對應的時間戳記(time stamp)可以看成是視訊影片(video)組成的單位元素，一連串具有時間戳記的影格便構成了視訊影片。BIFS Scene 也是相同概念所構成的，如圖 6 所示，BIFS command 與相對應的時間戳記(time stamp)可以看成是 BIFS Scene 組成的單位元素，一連串具有時間戳記的 BIFS command 便構成了 BIFS Scene。MPEG 組織特別把具有時間戳記的 BIFS command 稱爲 Access Unit [1]，顧名思義，可知道 Access Unit 就是 BIFS Scene 的單位元素。另外，MPEG-4 把壓縮過的場景描述部分，稱爲 BIFS stream。

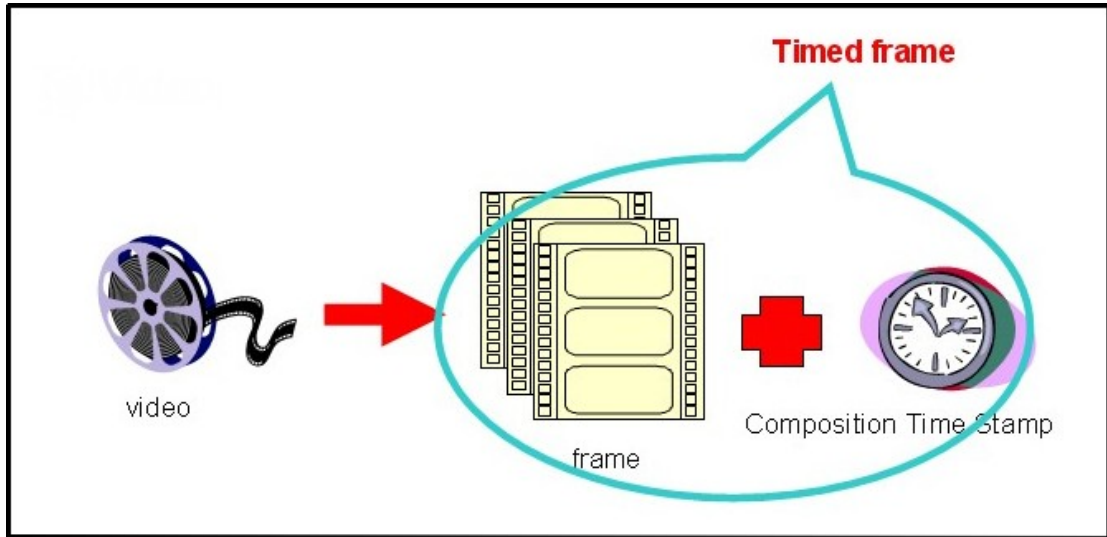


圖 5 傳統視訊影片的組成概念圖

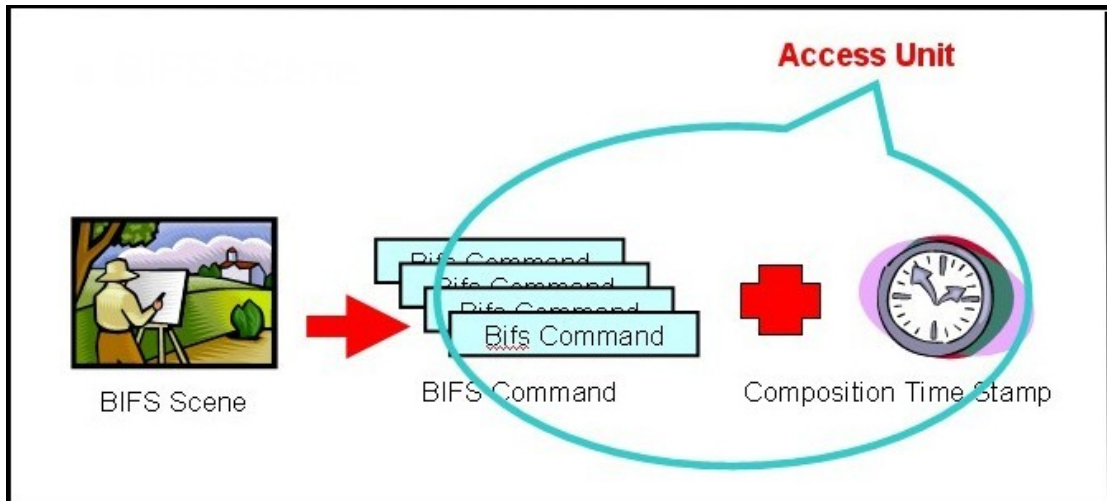


圖 6 BIFS Scene 的組成概念圖

2.1.2 物件描述子架構

在 MPEG-4 場景描述的架構中，除了劇情的描述是以 VRML 的概念作延伸，物件描述子的部分亦是以 VRML 的語法為依據，因此，物件描述子的部分也是由 Node、Field 所組成。物件描述子架構把現實世界對多媒體物件的抽象概念，以 Node 的方式做一一對應，如圖 7 所示，在現實世界的概念中，我們認為一個多媒體物件(Media object)可以由許多不同的串流(stream)所構成，而每個串流在 MPEG-4 的規範下可擁有自己的解碼方式(decoder)及與其他串流同步(synchronize)的規則，至於解碼方式則根據多媒體的類型(type)與組成(coding)的方式作不同的處理。這些對於多媒體物件的認知在物件描述子架構下皆以不同的描述子來表示，例如以 ObjectDescriptor 代表多媒體物件、以 ES_Descriptor 代表串流，在經由

這樣的對應之後，就可以對多媒體物件做詳盡的描述，而對描述子操作就相當於對物件操作，這樣便可以達到控制物件的目的。最後，把這些用來描述物件的描述子壓縮起來，成為與 BIFS stream 相對應的 OD stream(Object Descriptor Stream)。

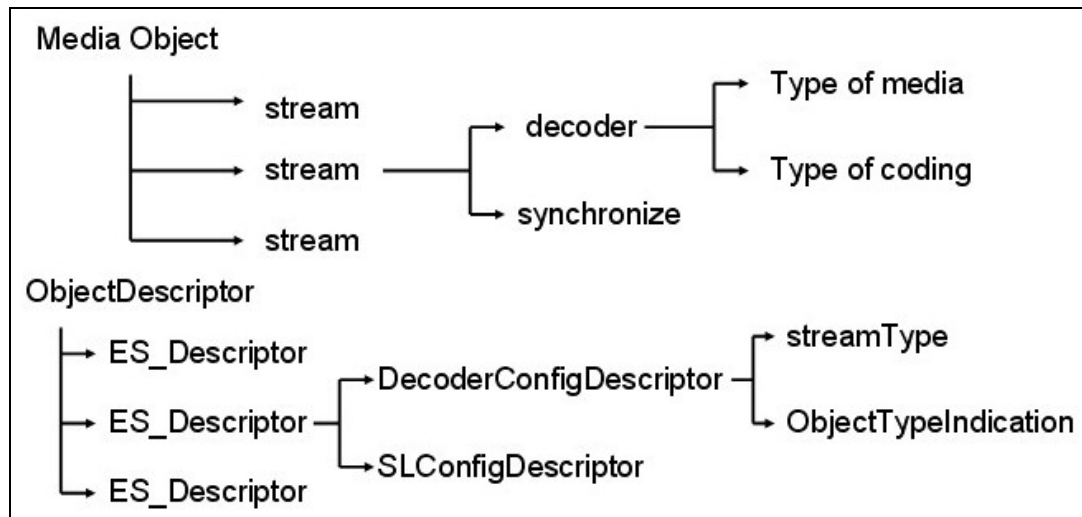


圖 7 多媒體物件與物件描述架構對應圖

相對於 BIFS command 可以控制場景劇情，在物件描述描述子架構中則有 OD command(Object Descriptor Command)可以控制物件。MPEG-4 定義了三種 OD command，分別是 Adding、Updating、Removing，用來增加、更新、移除關於物件方面的訊息。

在物件描述子架構中有一個很特別的描述子，稱為 Initial Object Descriptor (IOD)，它是整個 BIFS scene 的起始點，也就是說要獲得場景的資訊會先從這個物件描述子開始，一般來說，這個物件描述子指向 BIFS stream 與 OD stream，並定義場景外觀的相關訊息，是每個場景不可缺少的物件描述子。

2.1.3 整合場景描述與物件描述子

有了描述劇情的場景描述架構，以及描述物件的物件描述子架構，接下來便可以整合場景描述與物件描述子，產生 MPEG-4 檔案的場景描述語言文件。圖 8 以 BT script 為例，一份 MPEG-4 檔案的場景描述語言文件可以分成幾個區塊，WorldInfo 是用來做註解的區塊，版權宣告、作者資訊、版本資訊、系統需求都可以在此描述。Scene 與下方多個 Access Unit 是場景描述的部分，用來描述場景劇情，表達物件之間的關連性，以上的三個區塊都包含在場景描述架構中。最後的兩個區塊，包含一個 Initial Object Descriptor 與多個 Object Descriptor，就是用來建立物件描述子架構的部分。基本上每個 MPEG-4 檔案的場景描述語言文件都可以做這樣的區塊劃分，除了 Initial Object Descriptor 區塊在 MPEG-4 檔案的場景描

述語言文件一定存在之外，其餘的區塊是可有可無的。

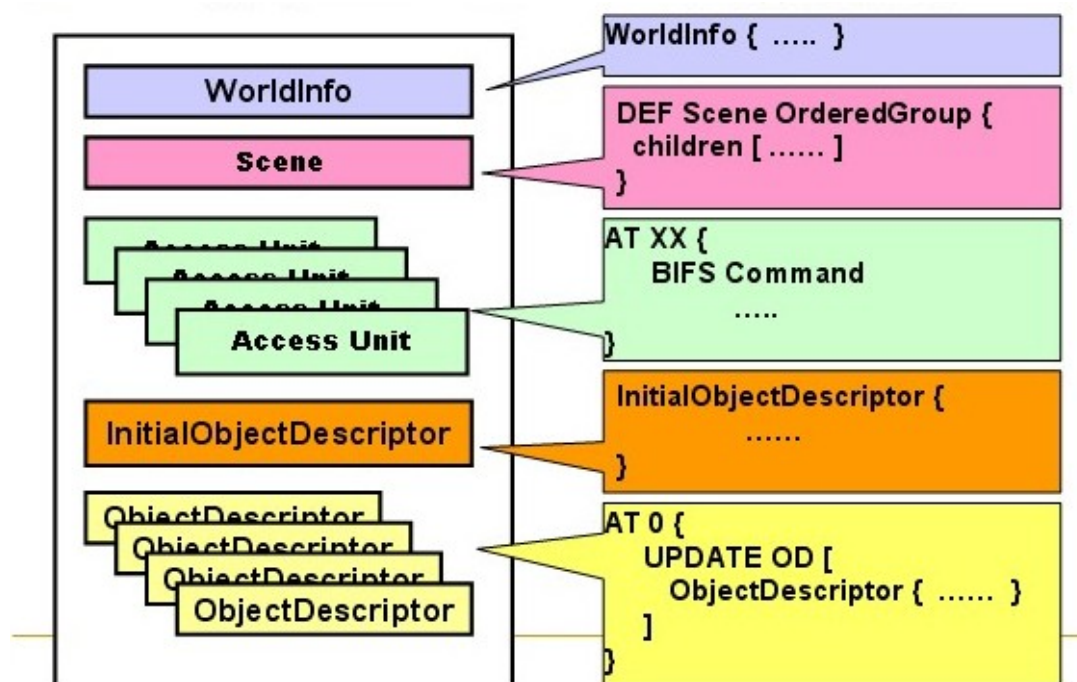


圖 8 BIFS 程式架構—以 BT script 為例

2.2 MPEG-4 場景描述語言

目前 MPEG-4 系統的場景描述語言，大致上有 BIFS，BT script，與 XMT 三種，以下分別作介紹：

2.2.1 BIFS

BIFS 是 Binary Format for Scenes 的縮寫，在 MPEG-4 系統中，是一種 Binary Format 的場景描述語言，MPEG 組織對 BIFS 做以下的定義〔1〕：

“A coded representation of interactive audio-visual scene description information (Binary Format for Scenes – BIFS)”

在 MPEG-4 檔案中，壓縮好的場景描述部分稱為 BIFS stream，可以用來控制場景內物件的動作，管理場景劇情的演出，但是由於 BIFS 是經過壓縮後的二元碼，所以這些二元碼以使用者的角度來看，只是一堆雜亂無章的序列，無法直接告知使用者此場景的相關訊息，屬於較低階的表示方式。

2.2.2 BT script

由於 BIFS 是以二進位的方式編碼，並不適合編輯者直接撰寫來描述場景的控制，因此，ENST 公司〔10〕根據 VRML 的語法，發展一種 BIFS Text(BT)的格式，顧名思義就是 BIFS 的 textual format，方便編輯者以撰寫程式語言的方式，來達到場景控制的目的，另外，BT script 將 BIFS 表達的意義以 Node 的方式作對應，可以很精確的描述 BIFS stream 的意思。

2.2.3 XMT

XMT 分成 XMT-A 與 XMT-O 兩種〔1〕，都是以 XML 為架構〔22〕的 textual format 場景描述語言。MPEG 組織將 XMT-A 訂為 low-level 的 MPEG-4 場景描述語言，與 BT script 相同，可以很精確的表達 BIFS stream 的意義。而 XMT-O 是 high-level 的 MPEG-4 場景描述語言，除了以 XML 語法為架構外，並且由 SMIL 為基礎作延伸，與 XMT-A 不同的地方是，XMT-O 隱藏 BIFS 一部份的複雜性，企圖以較容易的程式撰寫方式來控制場景物件的關係，但是，也因為如此，XMT-O 對場景內物件的控制力比起 XMT-A 或是 BT script 就顯得弱了許多。

2.2.4 分析比較

接下來，我們以一個例子說明 BT script 與 XMT 如何描述場景的物件，圖 10 表示在場景內有一張 JPEG 格式的圖片，若以 BT script 來表達，如圖 11 所示，若以 XMT-A 來表達，如圖 12 所示，若以 XMT-O 來表達，如圖 9 所示。我們可以發現，BT script 與 XMT-A 的語法很接近，都需要分別撰寫場景描述與物件描述子兩部分的程式碼，不同的地方在於 BT script 以 VRML 語法為架構，而 XMT-A 則是以 XML 為架構。另外，XMT-O 則是三種描述語言中最簡易的一種，場景描述程式碼與物件描述子程式碼是寫在同一個區域，複雜性較低。

```
  
  <transformation scale="1.0 1.0"/>  
</img>
```

圖 9 以 XMT-O 語法表示圖片



圖 10 圖片之呈現

//場景描述：

```
Shape {
  appearance Appearance {
    material Material2D { filled true }
    texture ImageTexture { url 11 }
  }
  geometry Bitmap {
    scale 1.0 1.0
  }
}
```

//物件描述子：

```
ObjectDescriptor {
  objectDescriptorID 11
  esdescr [
    ES_Descriptor {
      es_id 11
      decConfigDescr DecoderConfigDescriptor {
        objectTypeIndication 108 streamType 4
      }
      slConfigDescr SLConfigDescriptor { }
      muxInfo muxInfo { fileName picture.jpg }
    }
  ]
}
```

圖 11 以 BT script 語法表示圖片

```

<!--場景描述：-->
<Shape>
  <appearance>
    <Appearance>
      <material><Material2D filled="true"/></material>
      <texture><ImageTexture url="11"/></texture>
    </Appearance>
  </appearance>
  <geometry>
    < Bitmap scale="1.0 1.0"/>
  </geometry>
</Shape>

<!--物件描述子：-->
<ObjectDescriptor objectDescriptorID="11">
  <esdescr>
    <ES_Descriptor es_id="11">
      <decConfigDescr>
        <DecoderConfigDescriptor objectTypeIndication="108" streamType="4"/>
      </decConfigDescr>
      <slConfigDescr><SLConfigDescriptor/></slConfigDescr>
      <muxInfo>
        <muxInfo fileName="picture.jpg "/>
      </muxInfo>
    </ES_Descriptor>
  </esdescr>
</ObjectDescriptor>

```

圖 12 以 XMT-A 語法表示圖片

2.3 現有的 MPEG-4 編輯工具

本節介紹兩個現有的 MPEG-4 編輯工具，分別是 Digimax 的 MAXPEG Author [13]與資策會多媒體實驗室的 E-Learning Authoring Tool [15]。透過本節的介紹，可以知道這些 MPEG-4 編輯工具的優缺點。

🚦 Digimax 的 MAXPEG Author：

MAXPEG Author 具有視覺化的編輯環境，支援 2D 與 3D，可以輸出標準的 MPEG-4 格式檔案，以拖拉的方式編輯物件，而物件的屬性則利用 Scene Tree 視窗來做詳細的設定，另外也提供一個時間軸的視窗來設定事件的觸發，最後，此編輯工具允許使用者以 JavaScript 語言編輯場景的互動關係。

雖然 MAXPEG Author 提供強大的功能，對 MPEG-4 幾乎全面的支援，可是，在編輯場景的時候可能會用到 Scene Tree 視窗或是 JavaScript 語言，Scene Tree 視

窗如圖 13 的標示所顯示，是場景內所有節點的樹狀結構圖，只有瞭解 MPEG-4 場景概念的人，才會明白此樹狀結構的意義。對沒有撰寫程式專業能力的使用者來說，Scene Tree 視窗操作起來並不是那麼容易瞭解，因此利用 Scene Tree 視窗來達到對場景內所有節點詳細設定是很困難的，而 JavaScript 語言的撰寫，更是不利於沒有撰寫程式專業能力的使用者。

因此，雖然 MAXPEG Author 可以編輯出各式各樣的 MPEG-4 檔案，但是此工具並不適合沒有撰寫程式專業能力的使用者，主要還是作為程式設計師在製作 MPEG-4 檔案時的輔助工具較為適當。

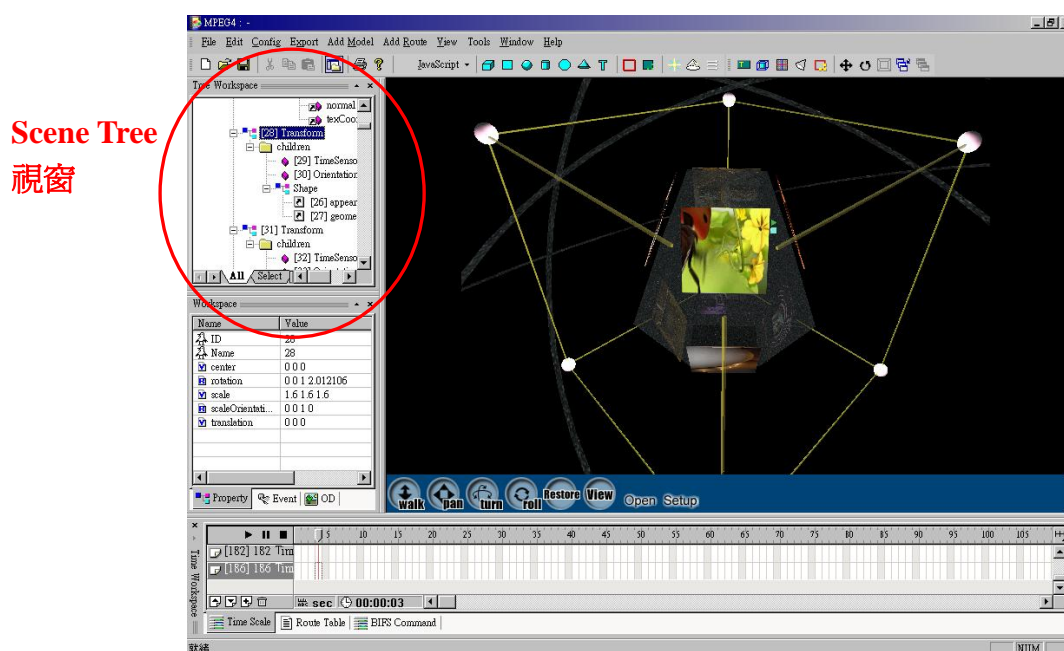


圖 13 Digimax 的 MAXPEG Author

資策會多媒體實驗室的 E-Learning Authoring Tool :

E-Learning Authoring Tool 的編輯功能不如 MAXPEG Author 強大，只能編輯簡單的場景，而物件屬性的設定，在 1.2.1 節中有提到，並沒有隱藏 MPEG-4 場景概念，使用者需瞭解 MPEG-4 節點的用途，才能做正確的設定。

E-Learning Authoring Tool 最大的特點，是提供樣版的方式來做編輯的動作，降低使用者編輯 MPEG-4 場景的困難度。如圖 14 所示，E-Learning Authoring Tool 內建旅遊導覽和英文對話兩種樣版，假如使用者要編輯此兩種類型的 MPEG-4 檔案，利用樣版模式可以迅速的製作出來，但是，如果使用者要編輯的 MPEG-4 檔案並不在這兩種類型的範圍內，E-Learning Authoring Tool 的編輯功能就略顯不足了。

除了對 MPEG-4 支援較少的缺點外，E-Learning Authoring Tool 與 MAXPEG Author 都同樣具有無法隱藏 MPEG-4 場景概念的缺失，如此一來，這兩種 MPEG-4 編輯工具就比較適合程式設計師來使用了。

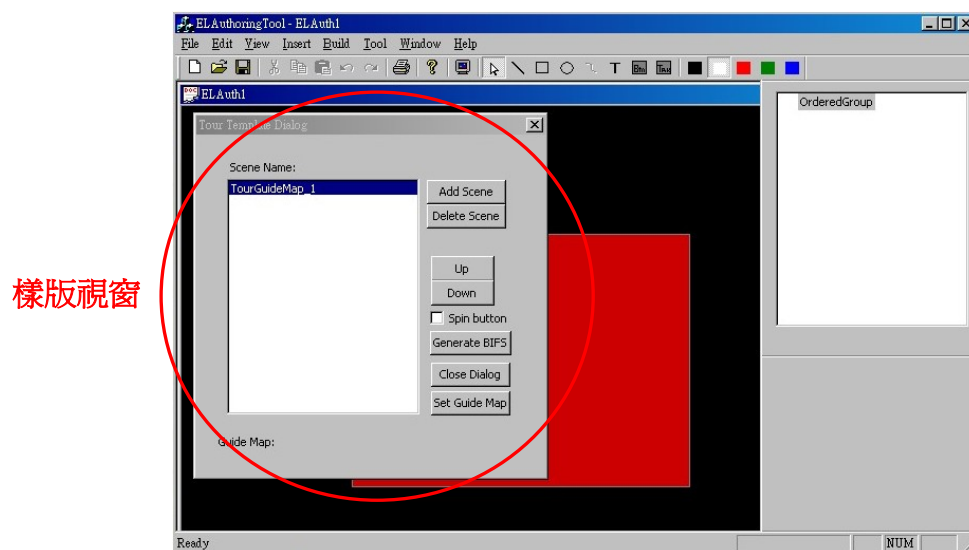
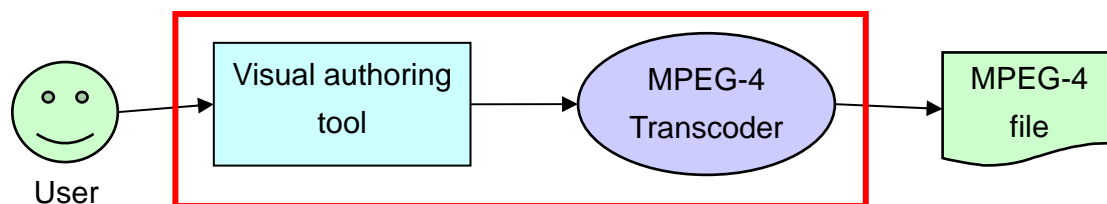


圖 14 資策會多媒體實驗室的 E-Learning Authoring Tool

2.4 解決方案

由於現有的 MPEG-4 編輯工具普遍都有需要瞭解 MPEG-4 場景概念才能編輯 MPEG-4 檔案的缺點，並不適合沒有撰寫程式專業能力的使用者，所以本研究把一個適合於沒有撰寫程式專業能力使用者的視覺化編輯工具，與可以產生 MPEG-4 檔案的轉譯器做結合，因此，MPEG-4 需求者只要操作視覺化編輯工具，即可創作出自己的 MPEG-4 檔案，可以解決現有的 MPEG-4 編輯工具的缺點。

如圖 15 所示，MPEG-4 需求者用視覺化編輯工具做編輯的動作，接下來，編好的多媒體透過轉譯器，產生最終的 MPEG-4 檔案。以使用者的角度來看，使用者只知道自已以視覺化編輯工具做編輯，接著便產生出 MPEG-4 檔案，並不知道中間有轉譯器作為媒介，因此透過這樣的方式，只要視覺化編輯工具的操作方式夠簡單夠直覺，能隱藏 MPEG-4 的場景概念，那麼 MPEG-4 需求者就可以自行製作 MPEG-4 檔案。



透過轉譯器產生 MPEG-4 檔案

圖 15 適合於沒有撰寫程式專業能力使用者的 MPEG-4 製作流程圖

三、系統需求分析

本研究將適合於沒有撰寫程式專業能力使用者的視覺化編輯工具，與可以產生 MPEG-4 檔案的轉譯器做結合，所以在系統需求方面，首先必須找到一個讓沒有撰寫程式專業能力的使用者操作起來很容易上手的視覺化編輯工具，再來就是訂出轉譯器的功能需求，根據這些功能需求來設計轉譯器，接著，為了方便視覺化編輯工具與轉譯器的結合，選擇適當的 MPEG-4 場景描述語言作為產生 MPEG-4 檔案的媒介，最後，根據上述的考量，做一整體的分析。

3.1 視覺化編輯工具的選擇

3.1.1 選擇的依據

在視覺化編輯工具的選擇方面，為了符合 MPEG-4 的特性，並且必須適合於沒有撰寫程式專業能力的使用者，一個視覺化編輯工具必須有下列的特性：

具備互動能力：

互動能力是 MPEG-4 的一大特性，如果視覺化編輯工具無法編輯出互動的效果，那最後產生的 MPEG-4 檔案也會缺少互動的能力，就失去 MPEG-4 的優勢，因此，所選擇的視覺化編輯工具必須具備編輯互動式多媒體的能力。

整合各類型多媒體：

在 MPEG-4 的規範中，不論是文字、圖片、聲音、影片，都可整合至 MPEG-4 檔案中，這些不同類型的多媒體充實了 MPEG-4 場景的豐富性，因此，所選擇的視覺化編輯工具也必須跟 MPEG-4 一樣，都整合各類型的多媒體，才能顯示出 MPEG-4 與眾不同的作用。

可描述劇情：

MPEG-4 跟先前 MPEG 系列(像是 MPEG-1、MPEG-2)不同的地方之一，就是場景內的物件可以用場景描述語言以劇情的方式來描述彼此間的關係，因此劇情的表達是 MPEG-4 重要的一環，我們所選擇的視覺化編輯工具要能有編輯劇情的能力，才能表達 MPEG-4 劇情的部分。

✚ 避免撰寫程式：

不論是互動能力、操作各類型物件、或是劇情的編輯，這個視覺化編輯工具都必須以直覺化、單純容易的方式完成編輯的動作，絕對要避免程式的撰寫，不然就失去使用視覺化編輯工具的意義了。

✚ 操作上適合沒有撰寫程式專業能力的使用者：

所選擇的視覺化編輯工具除了必須與 MPEG-4 的特點相近外，另一個重要的因素就是操作上是否適合沒有撰寫程式專業能力的使用者。由於有沒有撰寫程式專業能力的使用者瞭解 MPEG-4 場景概念十分不容易，所以這個視覺化編輯工具必須能完全隱藏 MPEG-4 場景概念，讓使用者在編輯的過程中，以拖拉點選的方式完成 MPEG-4 場景描述語言的撰寫。

✚ 具有 script：

由於所選擇的視覺化編輯工具在使用者操作完成後，必須將編輯好的多媒體資訊傳遞給轉譯器，因此這個視覺化編輯工具要有對多媒體完整的訊息描述，這樣的描述稱為 script，以 script 為視覺化編輯工具與轉譯器的媒介，才能適當的產生正確的 MPEG-4 檔案。

3.1.2 選擇的結果

由上述的分析可知，本研究所需的視覺化編輯工具最主要必須符合兩個條件，一個是與 MPEG-4 特色相近，另一個是適合於沒有撰寫程式專業能力的使用者。環顧目前市面上的各式多媒體編輯工具，發現智勝國際公司的編輯手，非常適合作為本研究的視覺化編輯工具，智勝國際公司的編輯手〔19〕有以下的特點：

✚ 豐富的編輯：

編輯手採用拖、拉、點、選的視覺化編輯技巧，結合文字、圖片、聲音、影片等素材來編輯製作互動式多媒體，以場景的編輯為單位，提供開場、退場的劇情控制模式，可以讓物件以依序或同時的方式做演出，並提供預先播放的能力。

✚ 人性化的操作模式：

編輯手以直覺式介面設計為理念，選擇背景、設定演員、編輯劇情都有

引導式功能操作，依照介面的提示，便可輕鬆的設計自己的多媒體。

由於編輯手製作出來的多媒體具有互動性，並整合各式多媒體，又有劇情的描述，可以知道與 MPEG-4 的特性非常接近，而拖、拉、點、選的編輯方式，再加上直覺式介面與引導式的操作，這樣人性化的操作模式的確適合於沒有撰寫程式專業能力的使用者。另外，由於編輯手是本實驗室與智勝國際合作開發的，所以軟體的原始碼取得較為容易，基於上述原因，本研究選擇以智勝國際編輯手，作為視覺化編輯工具。



圖 16 智勝國際編輯手

3.2 轉譯器功能需求

在選定視覺化編輯工具之後，接下來的工作就是分析轉譯器所需要的各個功能，一開始我們可以把轉譯器看成圖 17 所表示的功能元件，也就是說，轉譯器以視覺化編輯工具所產生的 script 為輸入，經過轉譯器的轉換後，產生出 MPEG-4 檔案，以這樣的角度來看轉譯器，可以讓我們清楚轉譯器要達成的目標，有助於轉譯器的分析。接下來本研究將根據轉譯器所需的功能及步驟，做進一步的分析。

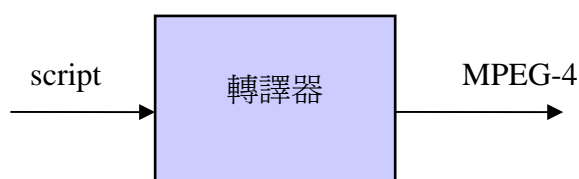


圖 17 轉譯器概觀

Script 分析步驟：

將視覺化編輯工具所產生的 script 丟入轉譯器後，首先轉譯器會分析並認知 script 的語法，瞭解 script 的語意，接著根據這些分析出來的內容，建立後來轉譯器所需要的資料結構，而資料結構的部分，則會包含各個物件的屬性、場景的設定、劇情的演出、以及多媒體物件的存放位置，可以說是替轉譯器準備好 MPEG-4 場景的所有資訊，這個步驟，稱為 Script 分析步驟。

場景處理步驟：

在 script 分析步驟完成後，便選擇一種 MPEG-4 的場景描述語言，作為轉譯器翻譯的目標語言，至於目標語言的選擇，會在 3.3 節中介紹。在對應的過程中，需要知道文字、圖片、聲音、影片各類型物件的寫法，而物件演出劇情、物件互動關係、物件屬性的對應、場景設定的對應，也要以 MPEG-4 場景描述語言，重新撰寫出來。這個步驟是轉譯器的核心步驟，一旦對應出錯會導致 MPEG-4 場景的演出出現意外的錯誤，所以必須特別注意。我們把這個步驟稱為場景處理步驟。

多媒體資料處理步驟：

另外，MPEG-4 標準所支援的多媒體類型，與場景內多媒體物件的格式可能會有不一致的情況發生，因此，這些多媒體的類型要做適當的轉換。例如，圖片的部分轉為 jpeg 格式，聲音的部分轉為 aac 格式，視訊的部分轉為 m4v 格式，這個檔案格式轉換的步驟，稱為多媒體資料處理步驟。

MPEG-4 壓縮步驟：

當場景處理步驟對應出 MPEG-4 場景描述語言的寫法，而多媒體資料處理步驟亦將物件的格式轉換完成，接著便可以找尋適當的 MPEG-4 encoder，整合場景描述與物件實體資料，壓縮成 MPEG-4 檔案。其中要特別注意的

是，此 MPEG-4 encoder 必須支援中文字的顯示，也就是說壓縮出來的 MPEG-4 檔案，在 MPEG-4 播放器播放時要能正確的對中文字解碼，這樣一來，這些 MPEG-4 檔案會更適合於國人的需要。以上壓縮 MPEG-4 的動作，稱為 MPEG-4 壓縮步驟。

使用者介面：

轉譯器除了產生 MPEG-4 檔案的功能外，另外也需要簡易的使用者介面，方便使用者對轉譯器做簡單的控制。例如，使用者可以控制產生出來的 MPEG-4 檔案的存放目錄，並且可以顯示目前 MPEG-4 產生的進度，因此，一個簡易的使用者介面可以增加轉譯器的價值。

經過上述的分析之後，我們可以為各個步驟建立不同的功能模組，用模組化的方式來組成轉譯器，而當轉譯器需要修正調整時，這種模組化的轉譯器可以抽換掉不適合的模組，或是更新模組的功能，因此，如圖 18 所示，轉譯器可分成 script 分析、場景處理、多媒體資料處理、MPEG-4 壓縮、使用者介面，共五大模組。



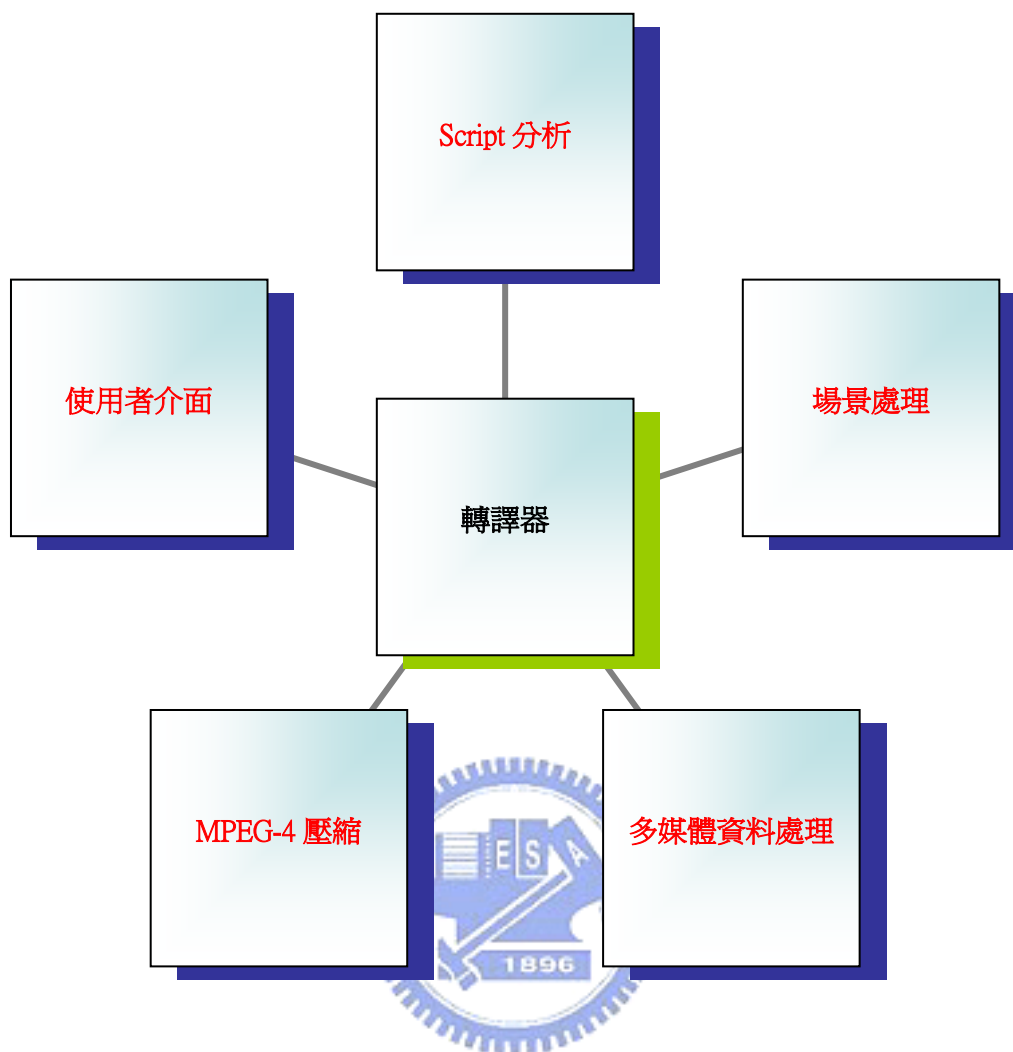


圖 18 轉譯器五大模組

3.3 MPEG-4 場景描述語言的選擇

在轉譯器的功能需求中，為了對應視覺化編輯工具產生的 script，必須選擇一種合適的 MPEG-4 場景描述語言，作為轉譯器的目標語言。目前 MPEG-4 的場景描述語言主要有 BIFS、BT script、以及 XMT，至於選擇的依據其實跟 MPEG-4 encoder 有密不可分的關係，因為通常 MPEG-4 encoder 的輸入就是 MPEG-4 場景描述語言，所以 MPEG-4 encoder 的選擇可以決定使用何種 MPEG-4 場景描述語言。

由於 MPEG-4 標準從 1998 年被 ISO 接受後，到現在已經發展許多年了，而 MPEG-4 encoder 的發展也相當成熟了，一般來說，MPEG-4 encoder 會接受 MPEG-4 場景描述語言與整理好的多媒體實體資料，來產生 MPEG-4 檔案，而不同的 MPEG-4 encoder 依照其設計理念與目標，會有不同的能力與特色，因此所能接受的 MPEG-4 場景描述檔也就不盡相同了。

表 1 MPEG-4 encoder 比較

組織	Encoder	特點
資策會多媒體實驗室	BifsEnc	Command - line，支援中文字
GPAC	mp4box	Command - line，有其他功能
IBM	XMTBatch	以 XMT-O 為輸入，轉換效率較差

表 1 是幾個 MPEG-4 encoder 的比較，IBM 的 XMTBatch [16] 以 XMT-O 為場景描述語言，設計目標是希望程式設計師撰寫 XMT-O 這個較簡易的場景描述語言，來控制 MPEG-4 場景；GPAC [8] 是一個發展 MPEG-4 標準的組織，它的 mp4box 除了基本產生 MPEG-4 的能力外，也提供額外處理 MPEG-4 檔案的能力，它使用的場景描述語言同時支援 BT script 與 XMT-A；資策會多媒體實驗室 [15] 的 BifsEnc 是從 ENST 的 mp4tool 所延伸出來的，以 BT script 為場景描述語言，最大的特色就是加入對中文的支援。

本研究所選擇的 MPEG-4 encoder 是資策會多媒體實驗室的 BifsEnc，主要原因是它對中文顯示的支援，這樣的選擇可以使得本研究製作出來的 MPEG-4 檔案更適合於國人。另一方面，本研究所需要的轉譯器目標語言，也就是 MPEG-4 場景描述語言的選擇就是 BT script，以符合資策會多媒體實驗室的 BifsEnc 的需求。

3.4 整體分析

經過上述的系統需求分析後，本研究以智勝國際的編輯手作為視覺化編輯工具，並且以編輯手的 script [2] 做為轉譯器與視覺化編輯工具連接的橋樑，而在轉譯器方面，分成 script 分析、場景處理、多媒體資料處理、MPEG-4 壓縮、使用者介面五個模組，再以 BT script 作為轉譯器的目標語言及 MPEG-4 encoder 的輸入，透過這樣的方式，整個系統的輪廓就越來越清晰了。

圖 19 表達出經過系統需求分析之後，MPEG-4 需求者製作 MPEG-4 檔案的流程。另外特別注意的是，一旦視覺化編輯工具選定後，整個系統設計的重點就落在轉譯器上。

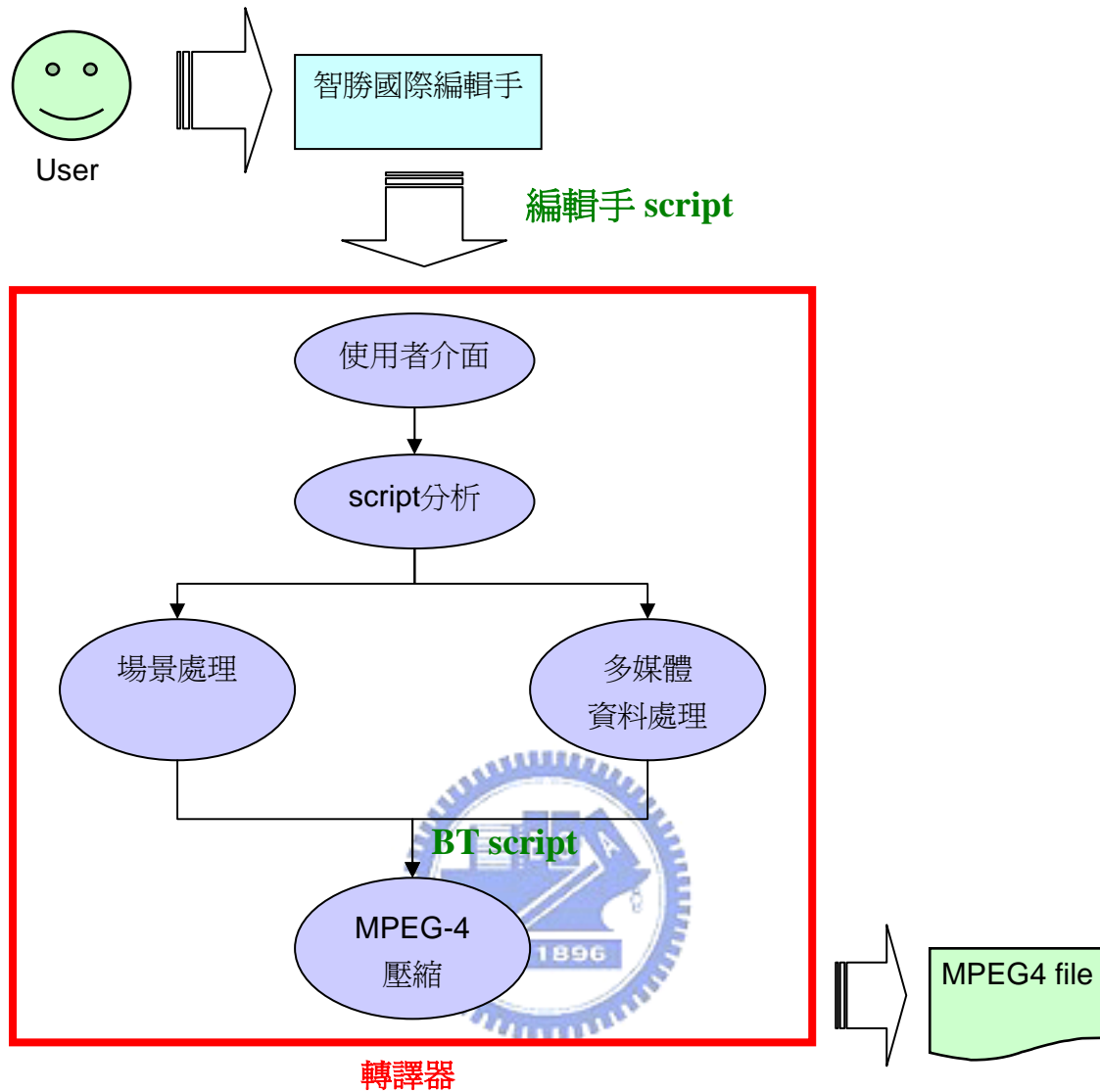


圖 19 系統需求分析後的 MPEG-4 製作流程

四、系統設計與實作

在系統的設計方面，首先我們以整體大架構的方式來看系統，以瞭解整個系統的運作流程、各個元件連結的方法、與系統所能提供的能力，接下來，由於轉譯器是系統設計的重點，因此會對轉譯器中的各個模組做詳盡的說明，並且將各個模組的設計概念與實作方法一一做介紹。

4.1 系統架構

4.1.1 MPEG-4 編輯工具與 MPEG-4 播放器

本研究的目標是實作一個適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具，爲了增加本研究所實作的 MPEG-4 編輯工具之價值，我們讓此編輯工具輸出兩種不同大小的 MPEG-4 檔案，其中一種是正常的大小，即是使用者在編輯工具上所見的大小，適合在 PC 上的播放器做播放，另一種則是 PC 版本的縮小版，目的是爲了在 PDA 上播放。例如，若使用者在編輯工具上設定場景大小爲 800x600，那麼輸出爲 PC 版本則大小仍爲 800x600，若輸出爲 PDA 版本則根據某個比例做縮小的動作。

一般來說，只要是符合 MPEG-4 標準的播放器就可以播放 MPEG-4 檔案，在 PC 的環境中，目前已經可以看到許多成熟的 MPEG-4 播放器，幾乎支援 MPEG-4 完整規格，然而在 PDA 的環境中，卻鮮少看到對 MPEG-4 的支援。有鑑於此，本研究編輯工具所輸出的 PDA 版本 MPEG-4 檔案，就需要一個執行於 PDA 環境的 MPEG-4 播放器，來做展示播放的工作，這個在 PDA 上的 MPEG-4 播放器則由實驗室的葉京荃同學負責研究與製作〔21〕，彼此再互相配合。

圖 20 顯示出本研究的 MPEG-4 編輯工具與 MPEG-4 播放器的關係，首先，MPEG-4 需求者利用本研究所實作的 MPEG-4 編輯工具，創作出 MPEG-4 檔案，接著根據使用者的需要，可以輸出爲 PC 大小的版本，並在 PC 上用 MPEG-4 播放器播放，也可以輸出爲適合 PDA 螢幕大小的格式，並在 PDA 上用葉京荃同學所製作的 MPEG-4 播放器播放。

爲了讓製作出來的 MPEG-4 檔案有最佳的展示效果，本研究在 PC 上所採用的 MPEG-4 播放器是資策會多媒體實驗室的 MPEG-4 播放器，此播放器能和轉譯器中的 MPEG-4 encoder 互相配合，展示效果較佳。

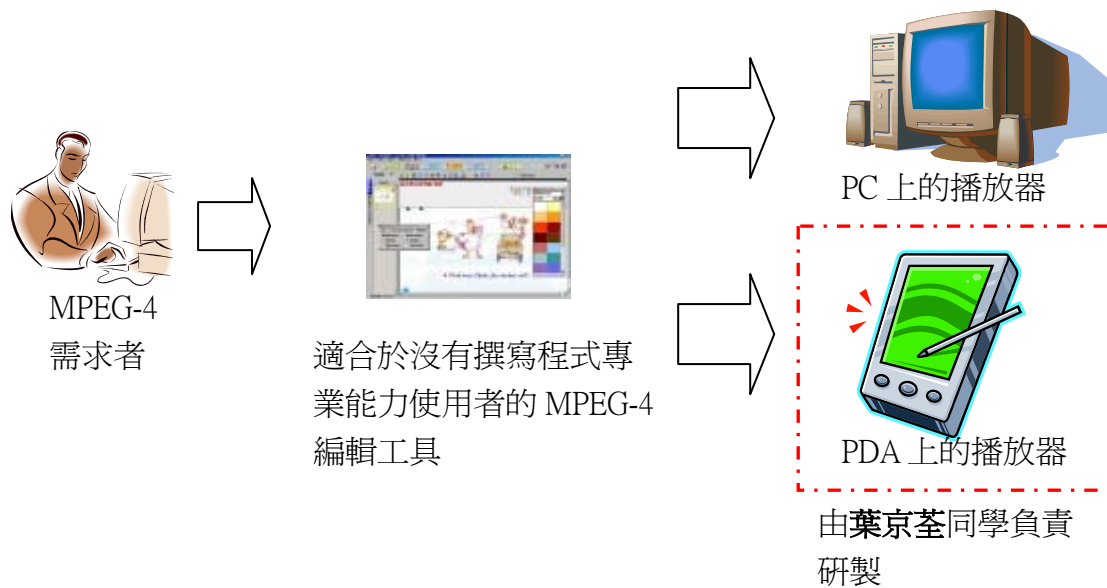


圖 20 MPEG-4 編輯工具與 MPEG-4 播放器

4.1.2 視覺化編輯工具與轉譯器的結合

瞭解本研究的 MPEG-4 編輯工具與 MPEG-4 播放器的關係後，接下來再針對本研究的 MPEG-4 編輯工具做分析。由第三章的系統需求分析我們知道，此適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具由視覺化編輯工具與轉譯器組成，因此視覺化編輯工具與轉譯器的結合方式便是此段所要介紹的。

視覺化編輯工具除了輸出 script 給轉譯器作為輸入外，視覺化編輯工具還會在使用者編輯完成，確定要轉成 MPEG-4 檔案的時候，呼叫轉譯器的執行檔，把使用者編輯完成的互動式多媒體，丟給轉譯器處理，而轉譯器經由一連串的轉譯過程與各個模組的分工，以 MPEG-4 格式產生出最終的互動式多媒體。

在視覺化編輯工具呼叫轉譯器的過程中，本研究以轉譯器的使用者介面模組作為視覺化編輯工具與轉譯器之間的媒介。使用者介面模組會取得使用者的操作指示，讓轉譯器可以適當轉譯 MPEG-4 檔案。整個過程如圖 21 所示。

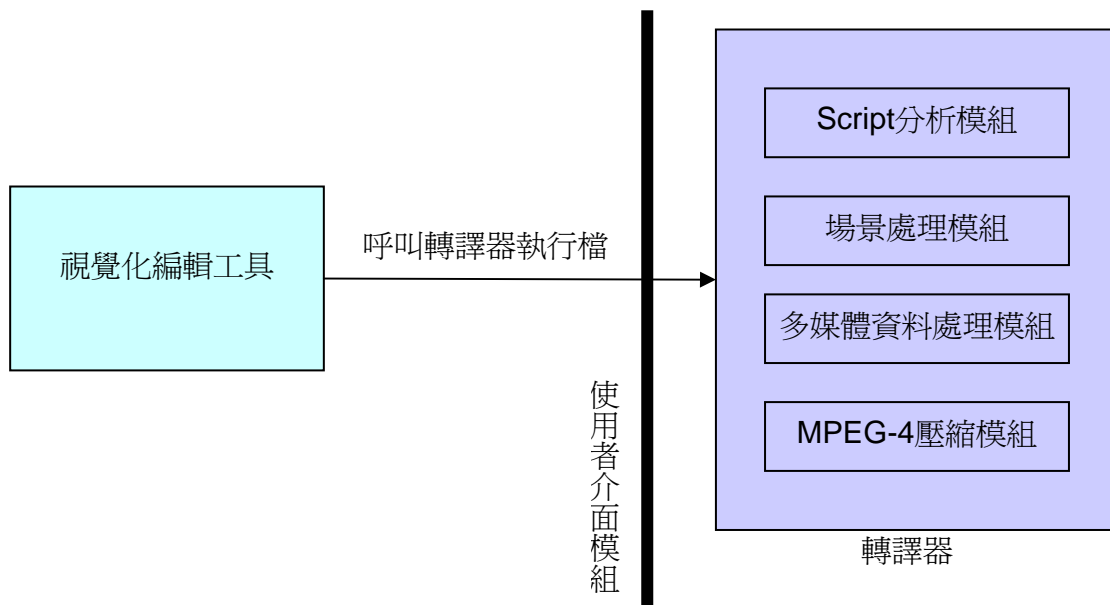


圖 21 視覺化編輯工具與轉譯器的結合

4.1.3 轉譯器模組概觀

當視覺化編輯工具呼叫轉譯器後，轉譯器接收視覺化編輯工具的 script 與使用者模組所收到的指示，分派工作給轉譯器中適當的模組來處理。根據第三章的分析，我們知道轉譯器的模組有 script 分析模組、場景處理模組、多媒體資料處理模組、MPEG-4 壓縮模組、與使用者介面模組，其中我們再把場景處理模組根據處理的訊息劃分成幾個子模組，分別是場景訊息子模組、靜態顯示子模組、互動劇情子模組、進階設定子模組、XML to BT 子模組，圖 22 表示出轉譯器模組的樹狀關係，各模組的實作細節會在 4.2 節做介紹。

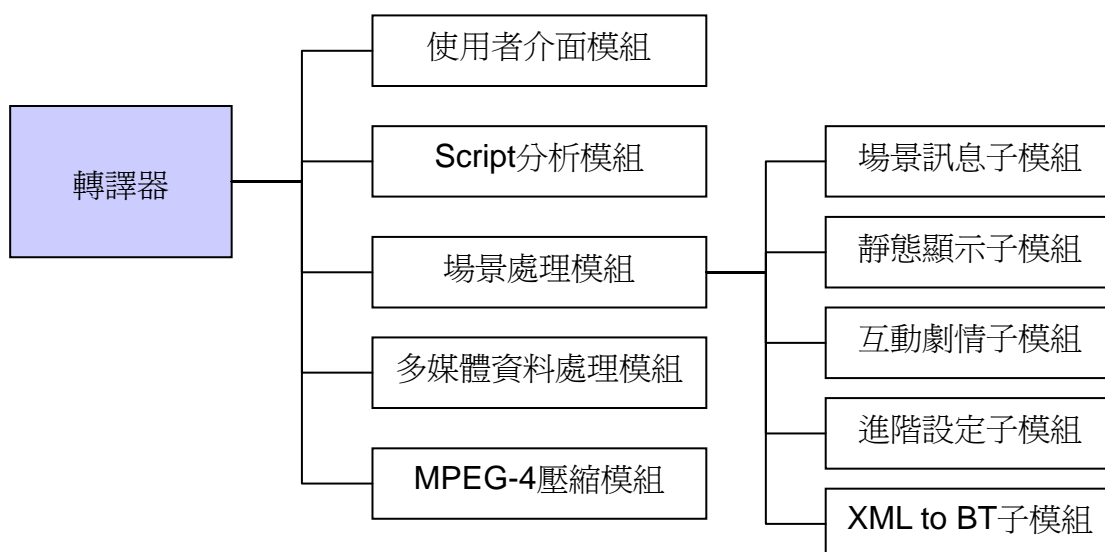


圖 22 轉譯器模組樹狀關係

4.1.4 轉譯器系統程式流程

圖 23 顯示系統程式的流程，我們把視覺化編輯工具輸入的 script 稱為 story，顧名思義以”故事”代表使用者編輯好的互動式多媒體，一個 story 可以由一個或多個場景(scene)組成。

至於轉譯器的系統程式流程，首先，story 經過使用者介面模組，在 Script 分析模組中會解析出此 story 的各個場景(scene)，依序的將場景丟給場景處理模組去產生相對應的 BT script，而此場景的多媒體資料就丟給多媒體資料處理模組做檔案格式的轉換，接著，MPEG-4 壓縮模組再整合上述資訊壓縮出一個場景的 MPEG-4 檔案，此時轉譯器會檢查此 story 是否還有其它的場景，若有，則把此場景的相關訊息丟給 Script 分析模組，再由上述相同流程產生出另一個場景的 MPEG-4 檔案，直到此 story 所有的場景都產生出相對應的 MPEG-4 檔案後，轉譯器才算完成工作。因此，一個 story 經過轉譯器的運作，會產生一個至多個的 MPEG-4 檔案，而檔案的多寡就根據 story 的場景數目不同而有相對應的變化。

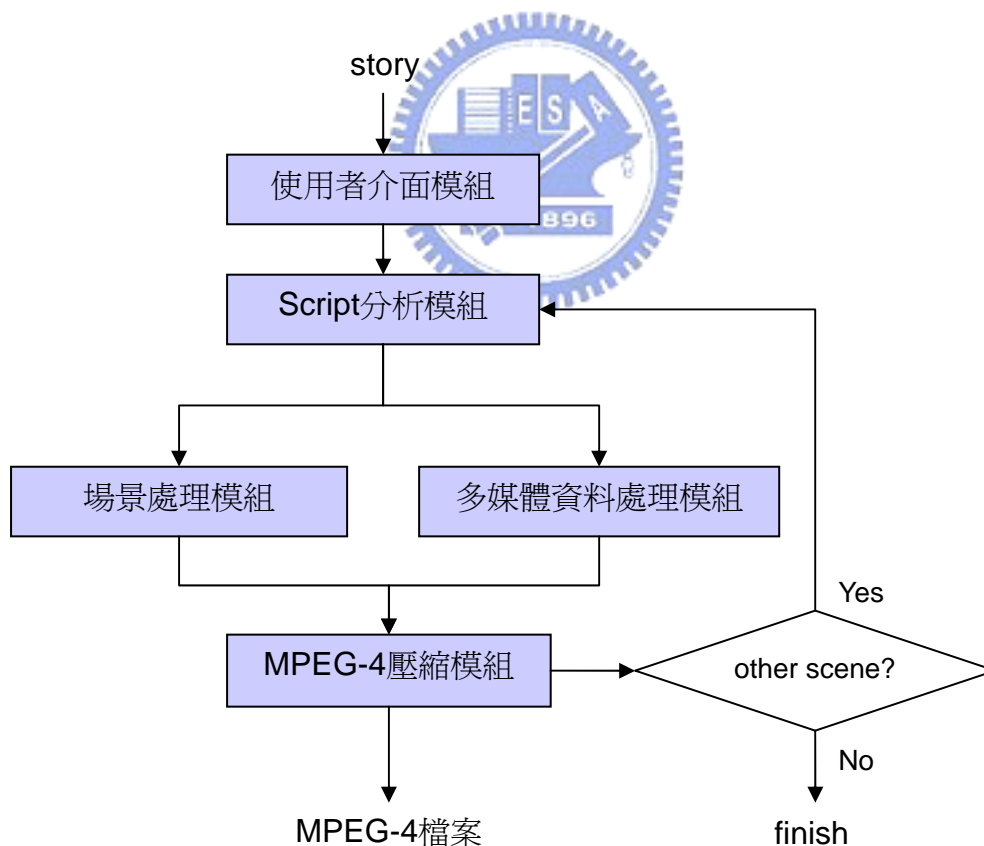


圖 23 轉譯器系統程式流程

4.2 轉譯器設計

本節除了闡述轉譯器的設計理念外，也實作出轉譯器，而轉譯器實作的環境如下：

作業系統：Microsoft Windows XP Professional Edition (Build 2600 : Service Pack 2)

開發工具：Borland C++ Builder 6.0 Enterprise Edition [12]

4.2.1 使用者介面模組

使用者介面模組主要的功能有二，第一個是讓使用者對轉譯器下指令，第二個功能是顯示出 MPEG-4 檔案目前的轉換進度。使用者對轉譯器所下的指令，如圖 24 所示，主要是 MPEG-4 檔案的輸出目錄和輸出類型，輸出目錄是 MPEG-4 檔案產生的目錄所在，而輸出類型則可以決定 MPEG-4 檔案的場景大小，有 PC 與 PDA 兩種輸出方式，至於此兩種輸出方式的不同，在 4.1 節已說明過。

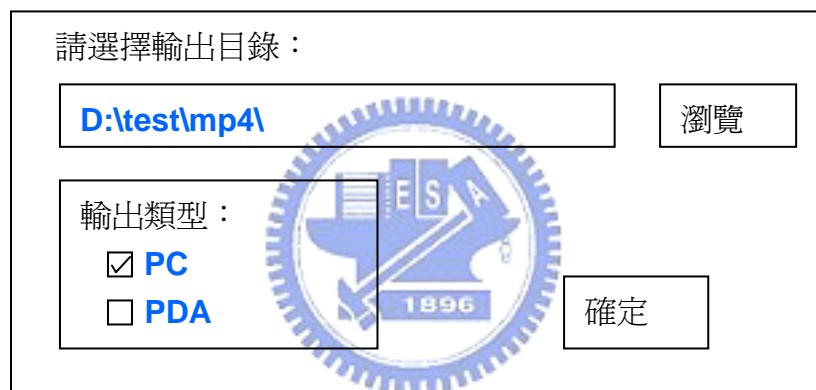


圖 24 使用者介面 — 選擇輸出目錄與輸出類型

在選定輸出目錄與輸出類型後，使用者按下確定之後會出現圖 25 的 MPEG-4 轉換進度顯示，目前進度以場景為單位，是指 story 中正在轉換的場景之進度，而整體進度則是以 story 為單位，是指整個 story 轉換的完成度。

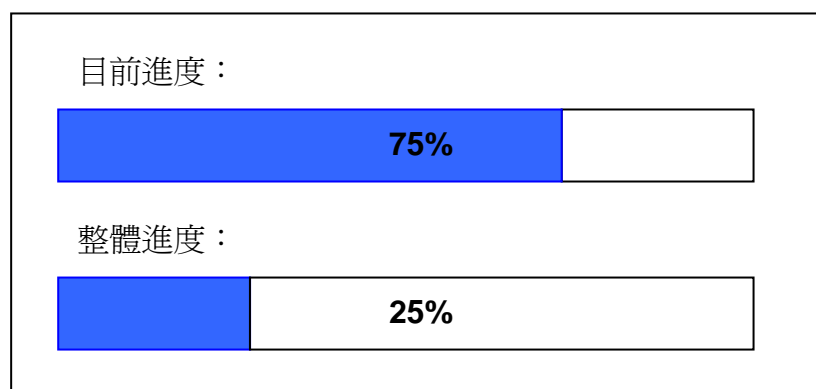


圖 25 使用者介面 — 顯示轉換進度

此轉換進度的顯示介面是以 call back function 的方式實作，也就是說，在使用者介面模組中，設定一個 function pointer，並讓此 function pointer 指向執行轉譯器轉換工作的函式，因此，在使用者介面中執行 function pointer 所指的函式，就可達到一邊顯示進度，一邊執行轉譯的目的。

4.2.2 Script 分析模組

Script 分析模組主要的工作是分析視覺化編輯工具所輸入的 script 資訊，也就是編輯手的 script，圖 26 顯示出分析的流程，首先，編輯手的 script 會經過 yyparse() 的分析，一方面檢查 script 是否符合編輯手 script 的格式與語法，另一方面也在 parse 的時候建立樹狀結構，接著，如果編輯手的 script 一切無誤的話，在 walk tree 的階段會去尋訪由 yyparse() 所建立的樹狀結構，並將 script 內的各項資訊從樹狀結構中取出，存在程式的變數中。

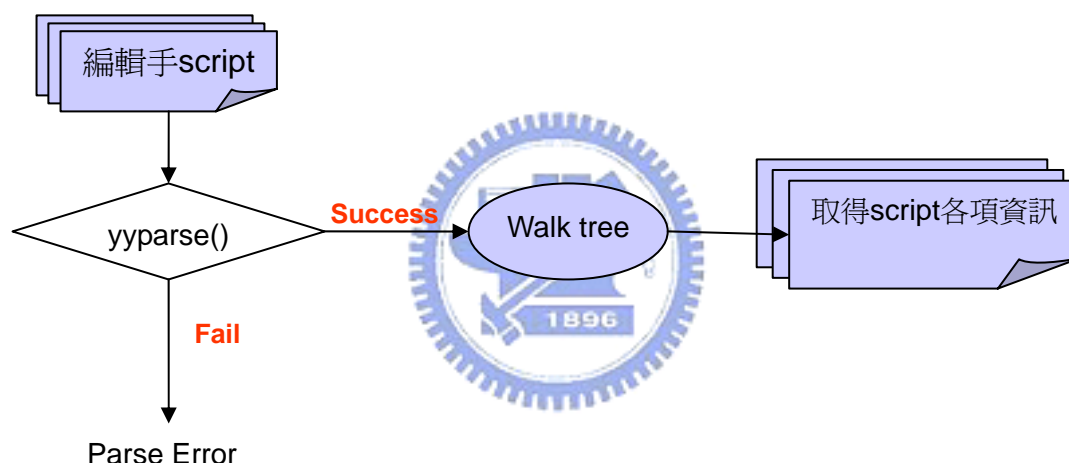


圖 26 編輯手 Script 分析流程

取得的 script 各項資訊，大致上可以分成三類不同的訊息：

🚦 物件訊息：

物件訊息指的是場景內各類型多媒體的相關資訊，所謂各類型多媒體，就是文字、圖片、聲音、影片這些組成場景的物件，而物件訊息包括物件的名稱、內容、位置、大小、路徑、縮放、演出方式等各種的訊息，另外，場景大小的資訊也一併放入此訊息中。

🚦 劇情訊息：

劇情訊息指的是場景內各物件演出的相對關係，有開場劇情與互動劇情兩類，劇情訊息是由依序演出、同時演出不同時消失、同時演出同時消失等

等不同演出方式所構成的。

多媒體訊息：

多媒體訊息類似 MPEG-4 的物件描述子架構，用來指向物件實體的位置，訊息內容包含物件的格式、物件存放的位置。

以上三種訊息，是由 script 分析模組負責取得，其中物件訊息與劇情訊息會傳遞給場景處理模組，而多媒體訊息則交給多媒體資料處理模組，如圖 27 所示。

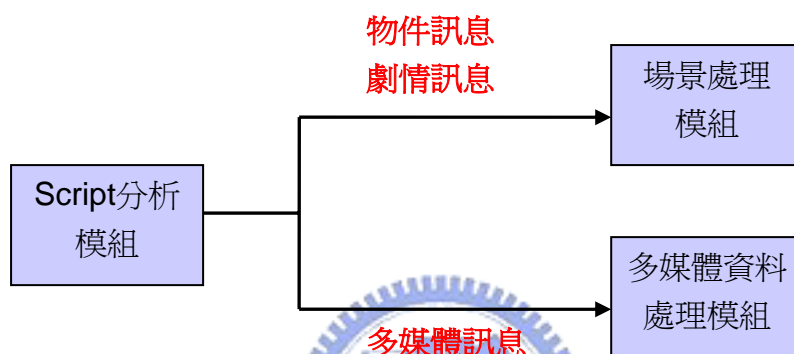


圖 27 各類訊息的流向

4.2.3 場景處理模組

場景處理模組接受 Script 分析模組所產生的物件訊息及劇情訊息，並轉為相對應的 BT script。由於物件訊息及劇情訊息中各部分所對應的 BT script 都不同，所以場景處理模組再分為場景訊息子模組、靜態顯示子模組、互動劇情子模組、進階設定子模組、XML to BT 子模組，分別對應不同部分的 BT script。

下列是五個子模組所負責的工作內容：

- 場景訊息子模組：負責場景外觀與初始化的工作。
- 靜態顯示子模組：負責描繪場景中各類物件，如文字、圖片、聲音、影片。
- 互動劇情子模組：處理劇情與互動的部分。
- 進階設定子模組：設定物件演出結束依據、物件深度、物件演出前後消失或出現。
- XML to BT 子模組：統整上述四個子模組所產生的 BT script，並寫入檔案。

如圖 28 所示，物件訊息與劇情訊息首先經過場景處理子模組與靜態顯示子模組，場景處理子模組與靜態顯示子模組會分別取得所需要的訊息，接下來，程式會判斷是否有需要產生互動劇情，再以互動劇情子模組來處理，而進階設定的判斷也與互動劇情的判斷相同，若有進階設定才需要進階設定子模組來運作，最後，再通過 XML to BT 子模組的整合，產生出與編輯手 script 相對應的 BT script，整個流程才算結束。

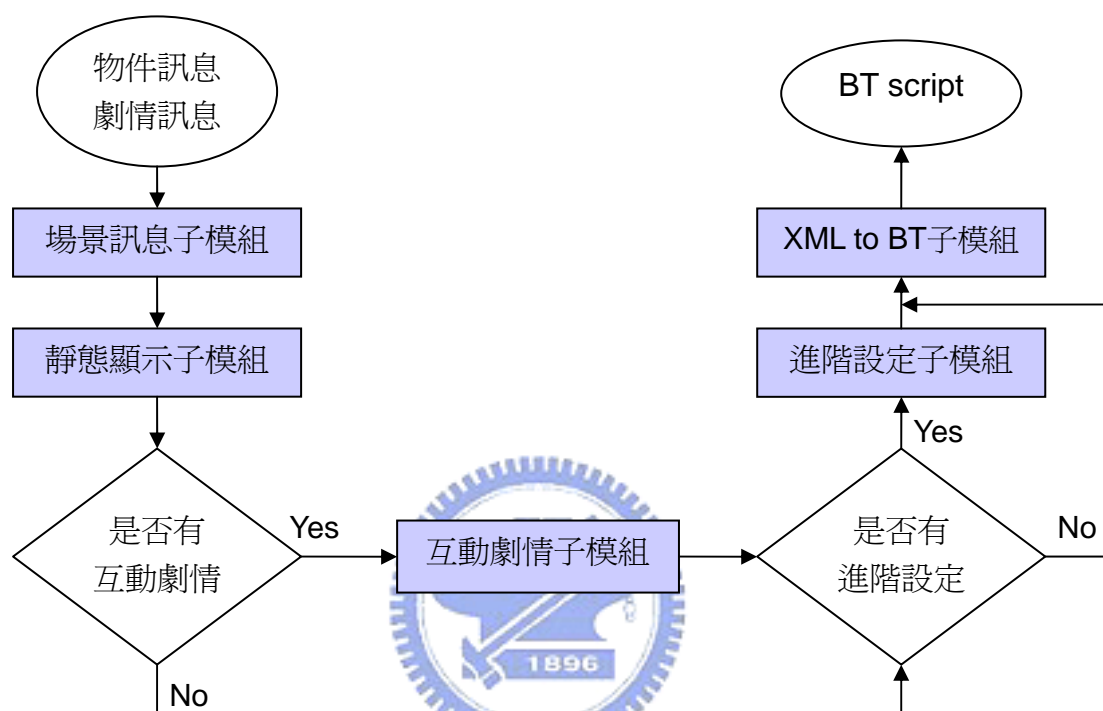
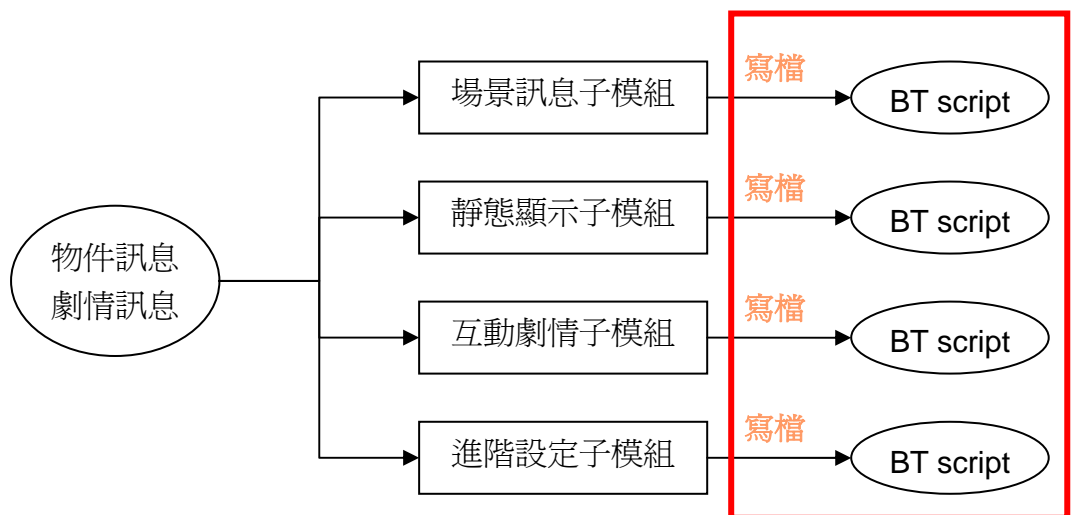


圖 28 場景處理模組流程

五個子模組會產生相對應的 BT script 程式碼，如果這些程式碼一旦產生就寫入檔案中，會造成一些額外的負擔。如圖 29 所示，各個子模組產生 BT script 後立即寫入檔案，首先，必須搜尋 BT script 應寫入的位置，以插入程式碼的方式填入檔案中，這樣一來，會導致插入位置之後的所有程式碼必須跟著移動，這並不是個有效率的方法。另外，各個子模組都做寫入的動作會造成多次寫檔，我們知道在電腦系統中寫檔是比較耗時的工作，因此如果能減少寫檔的次數，那麼轉譯的速度就會跟著加快。

因此，本研究利用 Borland C++ Builder 6.0 所提供的 XML 元件，來暫存子模組所產生的 BT script，這樣可以減少寫檔的次數，而 BT script 放置在 XML 元件所佔的記憶體空間中，在寫入時的效率也會比直接寫檔來的高。



寫檔時必須搜尋BT script應寫入的位置，且多次寫檔降低效率

圖 29 子模組多次寫檔造成效率低落

Borland C++ Builder 6.0 的 XML 元件以 `_di_IXMLDocument`、`_di_IXMLNode`、`_di_IXMLNodeList` 三種類別來組成，並且以 XML 的格式將資料儲存在節點之中，因此，在本研究中，各個子模組所產生的 BT script，會先以 XML 的格式存在 XML 元件中，如圖 30 所示，最後再以 XML to BT 子模組轉換 XML 格式成 VRML 格式，再做寫檔的動作。

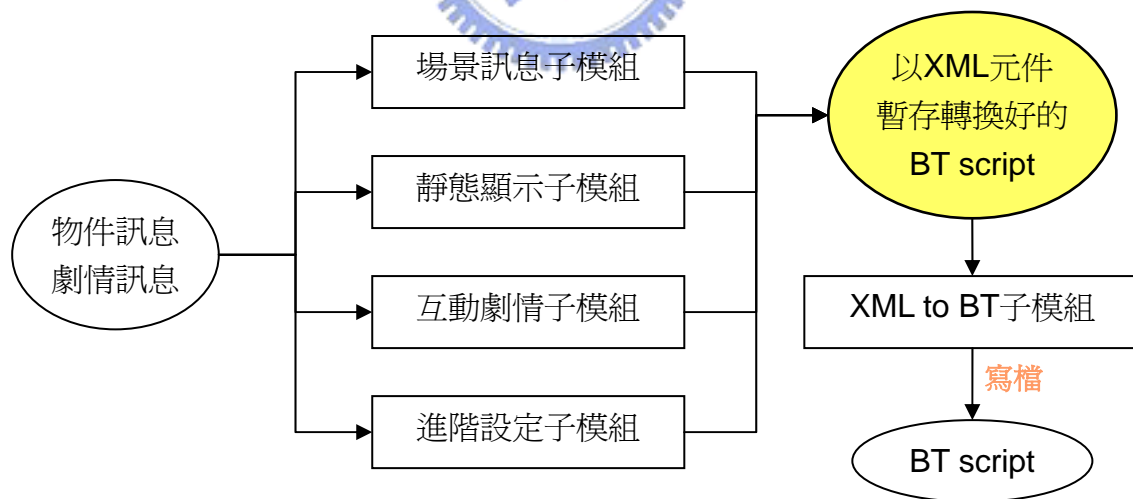


圖 30 以 XML 元件提高效率

4.2.4 多媒體資料處理模組

相對於場景處理模組產生顯示物件、互動劇情的 BT script，多媒體資料處理模組負責把實體資料做格式的轉換。實體資料有聲音、影片、圖片三大類型，為

了符合我們在 3.3 節選擇的 MPEG-4 encoder 所支援的格式，這些從視覺化編輯工具取得的實體資料可能會有格式不符的情況產生，因此多媒體資料處理模組會檢查所有的實體資料，將其分門別類，方便格式的轉換。

以下說明各類型實體資料的轉換過程：

聲音：

在聲音部分，如圖 31 所示，wav 與 mp3 轉為 MPEG-4 標準所規範的 aac 格式，接著，為了方便 MPEG-4 encoder 的作業，再將一個 aac 格式的聲音檔轉為 info、media、Nhnt 共三個檔案，其中 media 是實體資料的部分，info 與 Nhnt 則類似標頭檔，用來說明 media 的內容與規格。

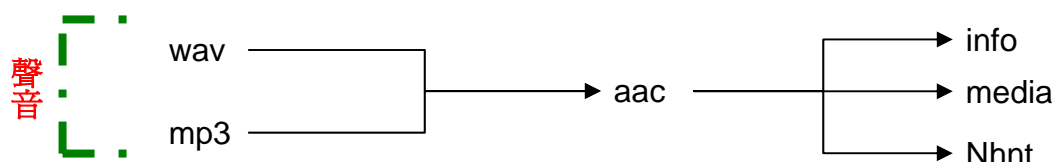


圖 31 聲音格式轉換流程

影片：

一般來說，影片檔案都會同時有視訊與聲音兩部分，聲音部分會按照上一段的方式轉成三個檔案，視訊部分則如圖 32 所示，mpeg 與 avi 格式會轉換成 m4v 的格式，一樣是 MPEG-4 標準所規範的，而一個 m4v 檔與 aac 相同，為了方便 MPEG-4 encoder 的作業，會轉成 info、media、Nhnt 共三個檔案。所以，一個結合聲音與視訊的影片檔案，經過多媒體資料處理模組的格式轉換後，會變成六個檔案。

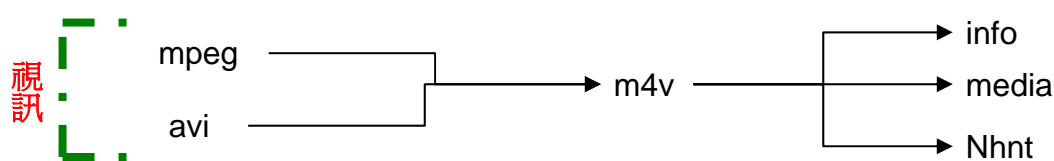


圖 32 視訊格式轉換流程

圖片：

圖片的轉換與聲音、影片的轉換相形之下單純的多，不管實體資料是 gif 還是 eir，都會轉成 jpeg 格式，讓 MPEG-4 encoder 做整合的動作，轉換流程如圖 33 所示。

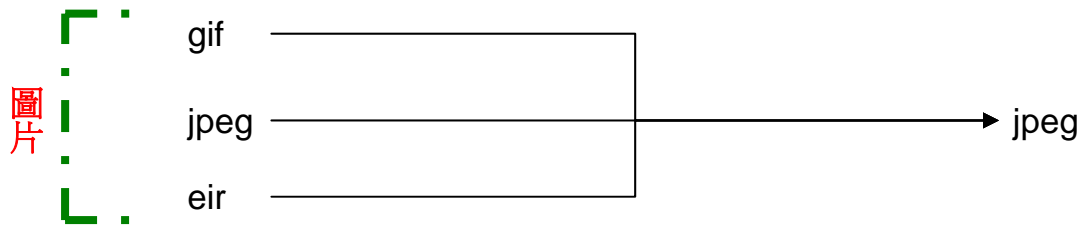


圖 33 圖片格式轉換流程

4.2.5 MPEG-4 壓縮模組

MPEG-4 壓縮模組主要的工作是將實體資料與 BT script 做整合，產生 MPEG-4 檔案。在 3.3 節中，我們選擇資策會多媒體實驗室的 BifsEnc 做為 MPEG-4 encoder，原因是它支援中文字的顯示，它的壓縮流程如圖 34 所示，場景處理模組產生出來的 BT script，還有多媒體資料處理模組轉換完成的實體資料，經過 MPEG-4 encoder 的壓縮後，表示一個場景的 MPEG-4 檔案就完成了。

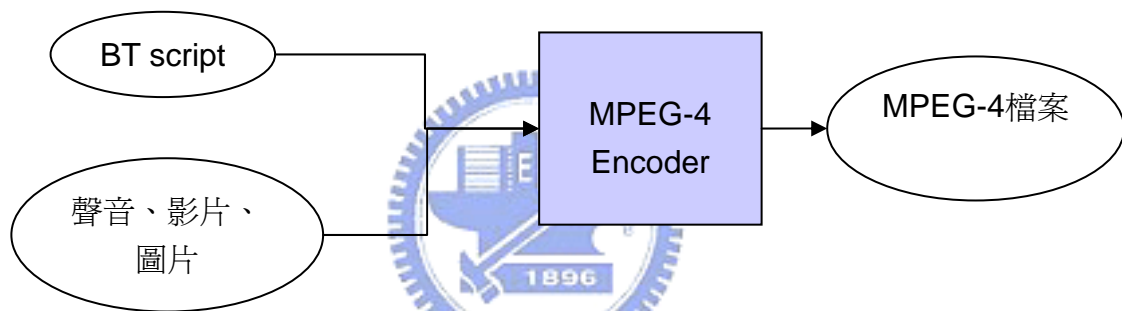


圖 34 MPEG-4 壓縮流程

4.3 轉譯器設計 – 場景處理模組

在 4.2.3 節中，我們已經看過場景處理模組的整體概觀，而在本節中，會針對場景處理模組的五個子模組，場景訊息子模組、靜態顯示子模組、互動劇情子模組、進階設定子模組、XML to BT 子模組做更進一步的介紹。

4.3.1 場景訊息子模組

場景訊息子模組的主要工作有兩項，分別是場景外觀大小的計算與初始化 BT script 的架構。在計算場景外觀大小的時候，會牽涉到使用者在使用者介面所勾選的 MPEG-4 檔案輸出類型。在程式的實作中，有一個紀錄場景縮放比例的變數，稱為 Ratio，所有與場景外觀互相關連的值，例如物件的路徑，都會根據 Ratio 的值做適當的調整。當 MPEG-4 檔案輸出類型為 PC 的時候，Ratio 變數的值為 1，此時不會影響到場景外觀或任何物件原始的值，可是，當 MPEG-4 檔案輸出類型為 PDA 的時候，Ratio 變數的值就變為場景外觀原始大小與 PDA 螢幕大小的比

值，接著就以此值對場景內的物件做相對應的調整。

圖 35 以場景外觀大小(Layout)為例，說明 Ratio 變數對場景的影響。當 MPEG-4 檔案輸出類型為 PDA 時，Ratio 變數會造成 Layout 的改變，因此也造成 Layout 部分的 BT script 程式碼表示方式會與 PC 輸出類型有所不同。

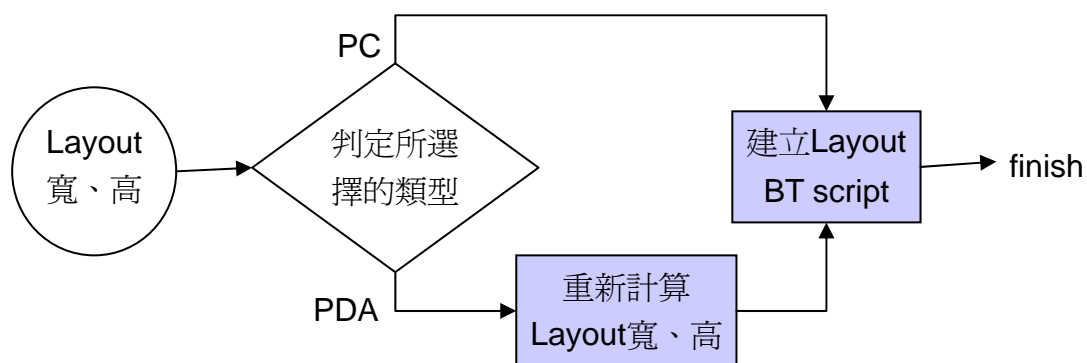


圖 35 Ratio 變數對場景的影響

場景訊息子模組的另一個重要任務，就是對 BT script 的架構，進行初始化的動作。初始化的動作會建立最基本的場景框架，以及 Initial Object Descriptor(IOD) 的 BT script 程式碼，在 IOD 的區塊中，又分成 BIFS stream 與 OD stream 的程式碼部分，而上一段所談到的場景外觀大小(Layout)，也包含在 BIFS stream 程式碼中，初始化完成的 BT script 架構如圖 36 所示。

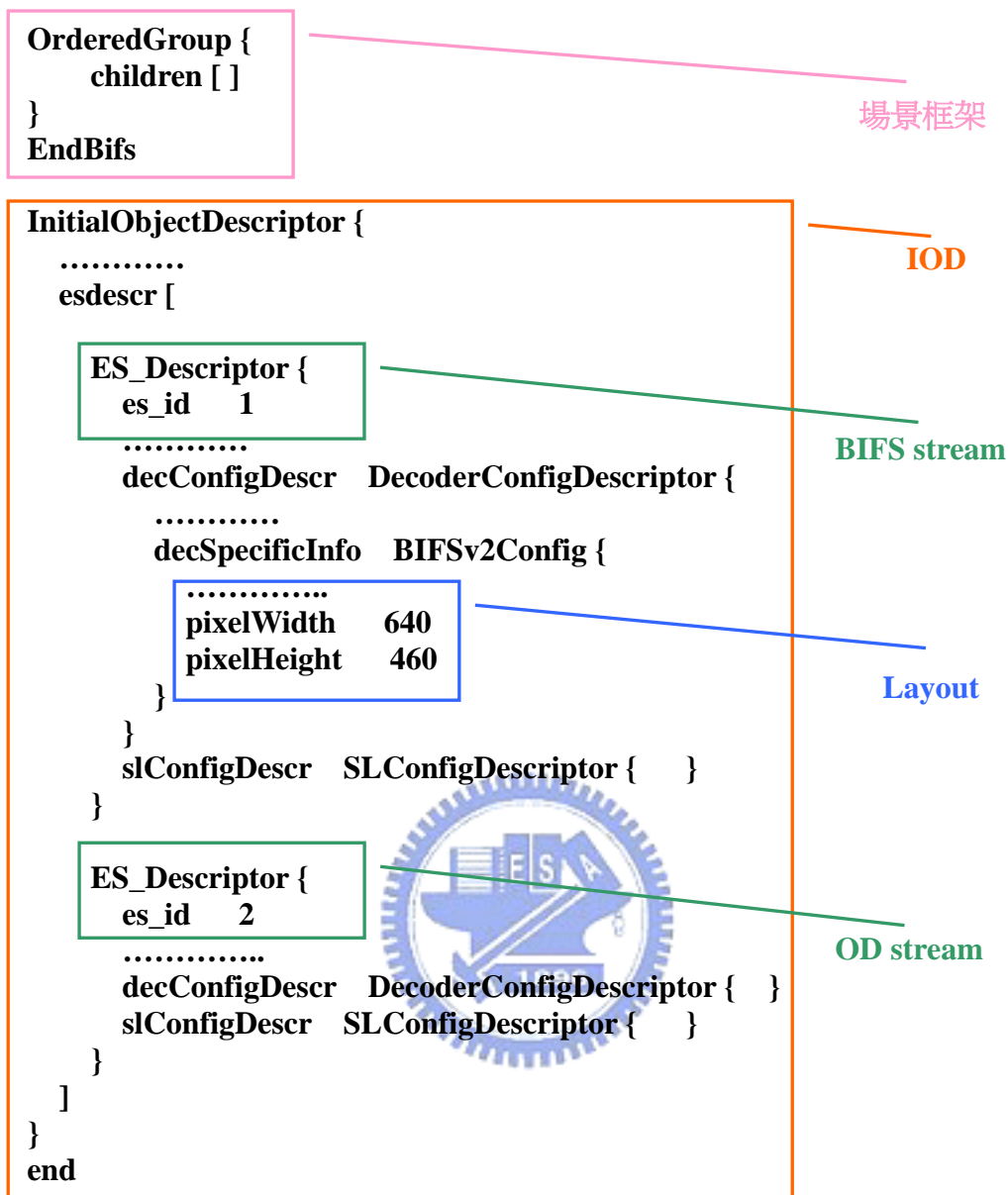


圖 36 初始化 BT script 架構

4.3.2 靜態顯示子模組

靜態顯示子模組負責編寫文字、圖片、視訊、聲音所對應的 BT script 程式碼，這些程式碼包括場景部分和物件描述子部分，也就是在場景中顯示物件的程式碼與描述物件實體資料的程式碼，除此之外，同時並編寫與這些物件演出相關的 BT script，例如路徑、縮放。

靜態顯示子模組的流程如圖 37 所示，首先，物件訊息會經由一個類型判定的程序，分析出物件的類型，會有文字、圖片、影片三種類型，並且建立相對應 BT script 程式碼，接下來，文字與圖片物件可以有附加聲音的功能，而影片物件

通常除了視訊以外還會有聲音，因此會有一個判定是否有聲音的程序，並建立相對應聲音的 BT script，再來，是有關物件演出的部分，一樣會有程序檢查路徑與縮放的功能，並建立相對應 BT script 程式碼。

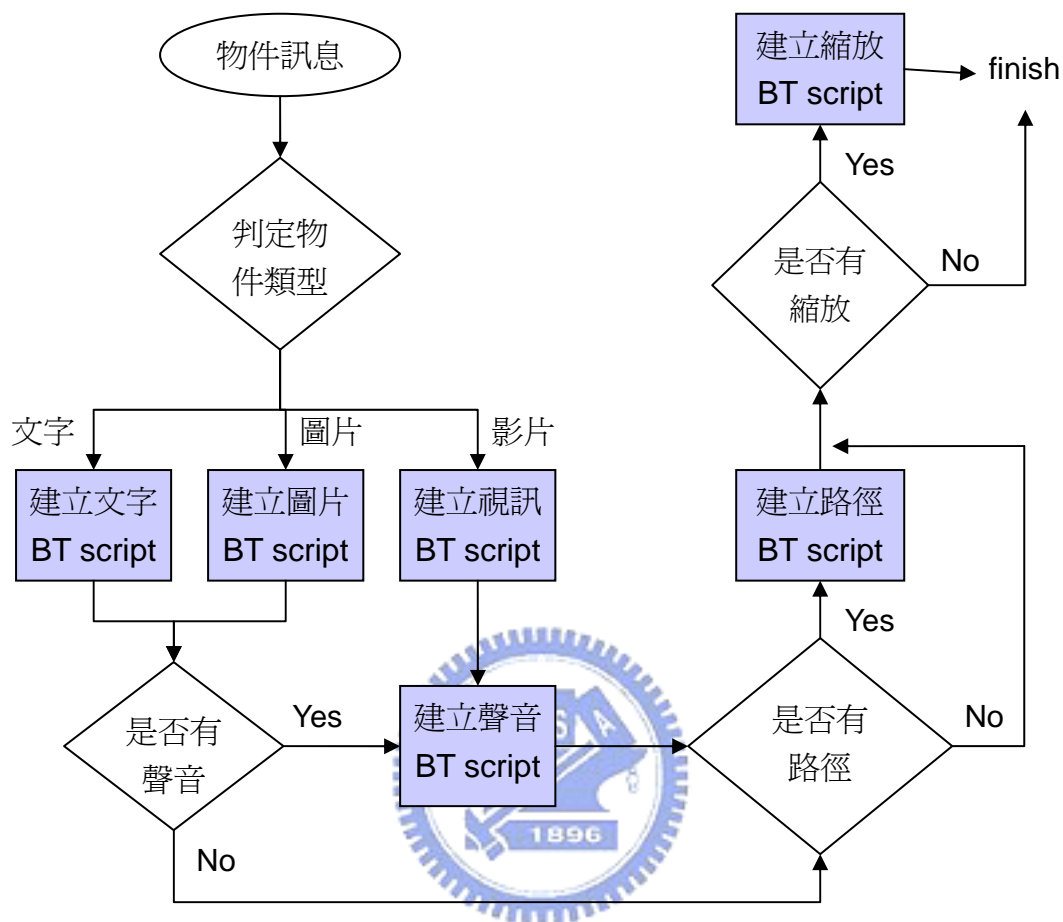


圖 37 靜態顯示子模組流程

不論物件類型是文字、圖片還是影片，他們都具有相同的程式架構，不過在某些細節上，仍然有相異的部分。場景部分的 BT script 可區分成數個程式區塊，如圖 38 所示，物件的 BT script 程式碼會有內容區塊、感應器區塊、路徑區塊、縮放區塊、聲音區塊這幾種相同的區塊，而影片物件則會多出視訊區塊，這些區塊會由一個 Group 節點包起來，形成文字、圖片、影片的 BT script 程式碼，也就是說，圖 37 所產生出來的 BT script 程式碼，會依據物件類型的不同，放置在適當的區塊中。以下分別介紹各個區塊。

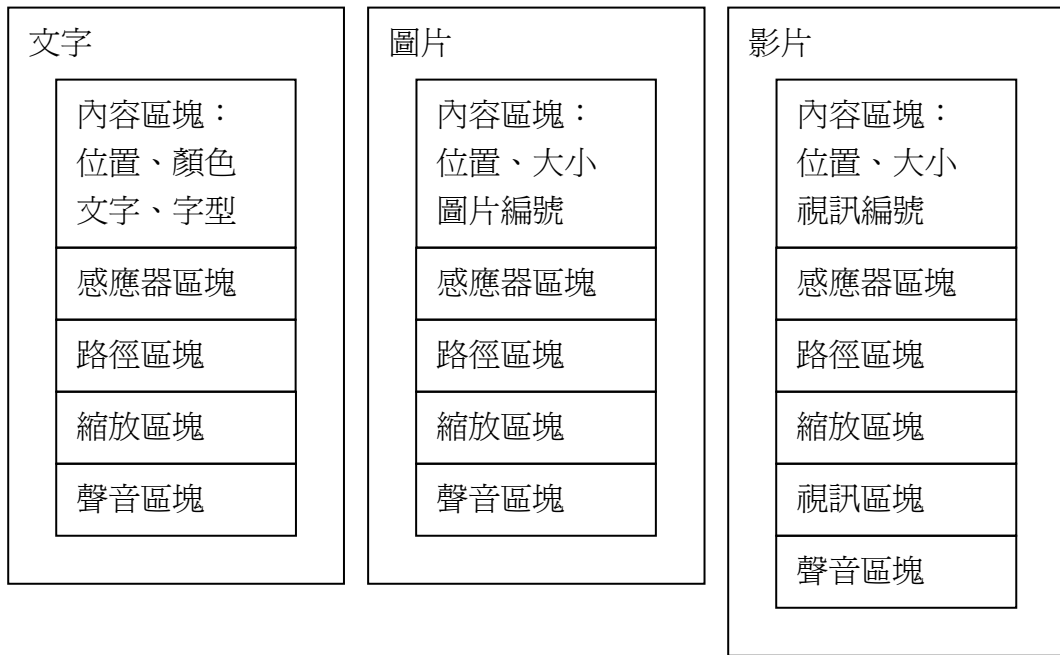


圖 38 三種不同類型物件的 BT script 程式架構

✚ 內容區塊：

由 Transform2D 節點所構成，文字物件會記錄出現位置、文字顏色、文字內容與字型等訊息，圖片物件則記錄出現位置、圖片大小與圖片編號等訊息，而影片物件除了記錄視訊編號外，其它的訊息與圖片物件相同。

✚ 感應器區塊：

主要由 TimeSensor 節點與 TouchSensor 節點構成兩個感應器，TimeSensor 感應器記錄此物件演出時所花費的時間，TouchSensor 感應器則使得此物件具有被滑鼠按下而觸發的能力。

✚ 路徑區塊：

以 PositionInterpolator2D 節點構成，在此區塊中會將物件路徑所行經的各點，以及行經各點的時間記錄下來，方便在物件演出時，做為路徑進行的依據。

✚ 縮放區塊：

與路徑區塊相似，同樣以 PositionInterpolator2D 節點構成，記錄的資訊也與路徑區塊類似，變成是紀錄各點的縮放比例。

聲音區塊：

以 Sound2D 節點和 MediaControl 節點組成，記錄聲音的編號，以及此聲音播放控制的方式，像是播放速度，播放位置之類的訊息。

視訊區塊：

此區塊專為影片物件所提供，以 MediaControl 節點來控制影片物件的視訊部分，基本上它的控制行為會與影片中聲音的控制是一致的，如此一來，視訊與聲音才會有同時播放的效果。

靜態顯示子模組另一個重要的工作就是建立各類型物件的物件描述子 BT script 程式碼。在文字、圖片和影片三種物件類型中，擁有實體資料的物件就需要建立物件描述子 BT script 程式碼，例如，一個擁有聲音的文字物件，雖然文字部分不需要物件描述子，但在聲音的部分，則需要物件描述子來指向實體資料的位置，由此可知，只要物件擁有圖片、聲音、視訊的資料，就必須為其建立物件描述子 BT script 程式碼，以對應實體資料的位置。



圖 39 物件描述子 BT script 程式架構

圖 39 是物件描述子 BT script 的程式架構，首先，整個物件描述子會有一個 Object Descriptor 編號，用來區別不同的物件描述子，此編號也可作為場景部分程式碼引用的依據。接著，設定 DecoderConfigDescriptor 節點的 ObjectTypeIndication 與 streamType 兩個 Field，可以讓 MPEG-4 系統明白此描述子是指向何種類型的物件，若是聲音的話這兩的值分別設定為 64 和 5，JPEG 圖片則是 108 和 4，視訊則為 32 和 4。最後，實體資料路徑會記錄此多媒體檔案的真實位置，通常是電腦系統中某個檔案的路徑。

4.3.3 互動劇情子模組

互動劇情子模組負責處理劇情訊息，如圖 40 所示，由 **Script** 分析模組所取得的劇情訊息，會由一段至多段劇情所構成，而每段劇情又可分為數個段落，每個段落則由不同數目的物件負責演出的工作。圖 41 是編輯手用來描述劇情的例子，我們假設圖 40 的劇情 0 正是這個例子，這個例子的意義是，劇情一開始先由物件 0 演出，等到物件 0 演出完畢後，再由物件 1 與物件 2 同時演出，但是兩者是各自演完後結束，接下來物件 3 與物件 4 雖然也是同時演出，但是先演完的會等待後者，直到兩者都演完後，才一起做結束的動作，而圖 41 最後一行的意義則是由目前的場景跳躍至場景 SceneXX。由圖 40 劇情 0 與圖 41 比較可以發現，圖 41 的每一行其實代表一個段落。

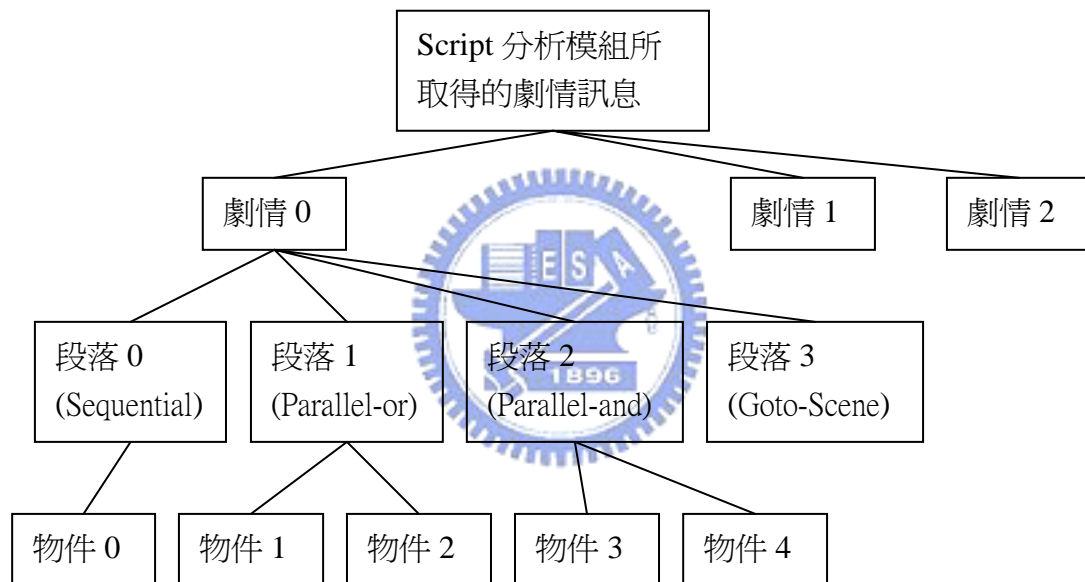


圖 40 劇情訊息樹狀圖

```
Actor000.play();
parallel( Actor001.play(), Actor002.play() );
parallel( __parameter.parallel_and(), Actor003.play(), Actor004.play() );
EBook.GotoScene( "SceneXX" );
```

圖 41 編輯手描述劇情的例子

表 2 劇情的處理方式

劇情	物件 0		物件 1	物件 2	物件 3	物件 4	SceneXX	
	Sequential		Parallel-or		Parallel-and		Goto-Scene	

我們把圖 41 的四個段落分爲四個不同的類型，如圖 40 劇情 0 所示，分別是依序演出(Sequential)、同時演出不同時結束(Parallel-or)、同時演出同時結束(Parallel-and)、換幕(Goto-Scene)，本研究對於段落的處理，就是以這四種類型爲依據，並且以表 2 的方式記錄各個段落演出的物件。至於劇情的觸發方式則有兩種，一種是使用者觸發，如按下滑鼠左鍵，另一種是特殊事件觸發，如開場、退場這種特殊的時間點，這兩種方式都會引起劇情的演出，不論劇情是由哪一種方式觸發，都會以表 2 記錄之。

一段劇情轉爲 BT script 程式時，會成爲一個 Access Unit，也就是說，如果由 Script 分析模組所取得的劇情訊息具有多段劇情的話，轉成 BT script 就會有多個 Access Unit。圖 42 顯示一段劇情的 BT script 程式碼概念圖，分成多個功能區塊和數個段落，AT XXX 代表此劇情呈現的時間，並且將 BT script 程式碼包含在裡面。圖 42 的各個功能區塊都是由 conditional 節點所構成，而演出的順序基本上是由下往上一個接一個觸發，一開始，觸發區塊會觸發初始化區塊和段落 0，初始化區塊會將劇情內所有的物件做初始化的動作，像是物件顯示位置、文字或貼圖，都會在此區塊完成。而每一個段落，根據段落類型(Sequential、Parallel-or、Parallel-and)的不同，會由一個或多個物件做演出，並在段落演出結束後，觸發下一段落的開始。當劇情的每一段落都演出完畢後，最後會觸發一個結束區塊，來對應劇情演出結束的動作，像是關掉劇情最後一段的聲音，就會在此區塊完成。

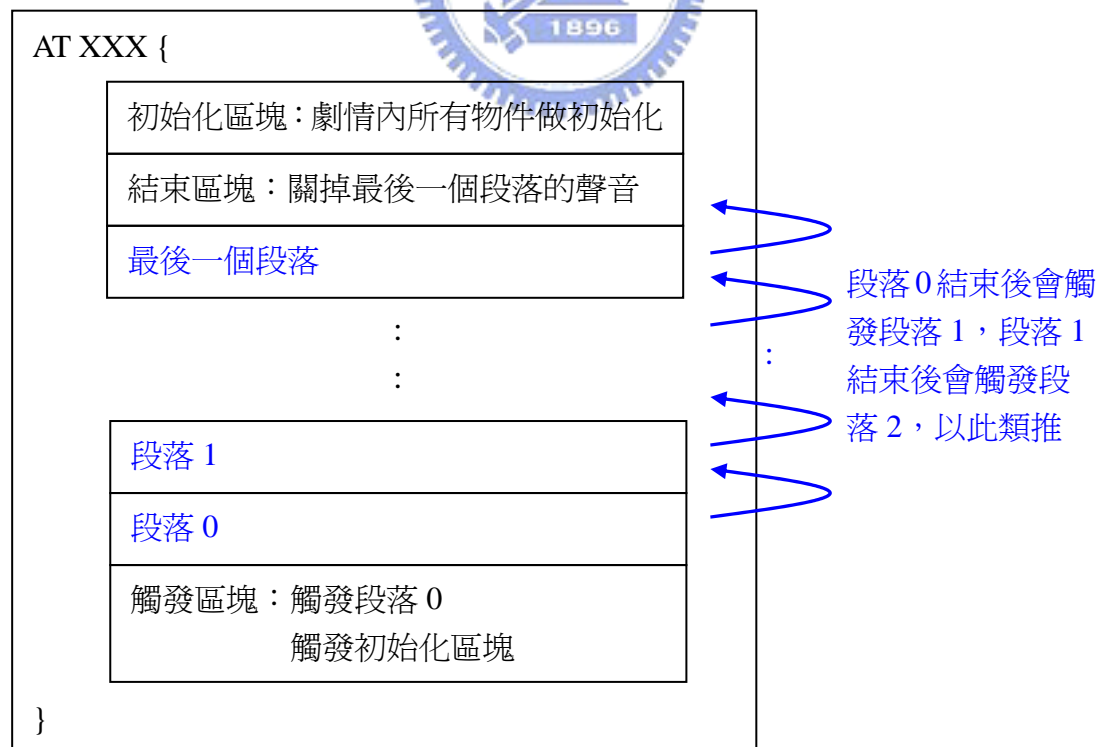


圖 42 劇情的 BT script 程式碼概念圖

互動劇情子模組除了處理劇情的 Access Unit 之外，還會產生播放各個物件的 BT script，稱之為物件播放程式碼。在上一段的說明中，我們知道劇情的每一段落會由一個或多個物件的演出來構成，而一個物件的演出就是靠一份物件播放程式碼來達成演出的效果。因此，考慮段落的類型，若段落是 Sequential 類型，則演出的物件只有一個，所以此段落只需要一份物件播放程式碼，反之，若為 Parallel 類型，那麼參與此段落演出的物件就不只一個，所以此段落需要的物件播放程式碼也就不只一份了。

另外，這裡所謂的物件播放程式碼，與 4.3.2 節中靜態顯示子模組產生的物件演出程式碼有所區別。物件播放程式碼是用來”驅動”物件演出程式碼，也就是說，互動劇情子模組產生的物件播放程式碼，會驅動靜態顯示子模組產生的物件路徑，因此我們就可以看到物件在場景畫面上的移動軌跡，也就是”播放”的效果。

圖 43 是物件播放的 BT script 程式碼概念圖，物件的播放是構成每一個段落的元素，而物件播放的程式碼是以 conditional 節點構成，首先，內部有一個 TimeSensor 節點作為此物件播放的時鐘，紀錄演出時間的長度，接下來，若上一個段落有關閉聲音的需要，在 TimeSensor 節點下會有關閉聲音的程式碼區塊。再來，會有路徑驅動、縮放驅動、視訊播放、聲音播放各種動作的程式碼區塊，這些程式碼就是讓物件播放的核心，最後，此物件可能有觸發其他劇情段落演出的能力，若有此需要，便會加入觸發其他劇情段落的 BT script 程式碼。

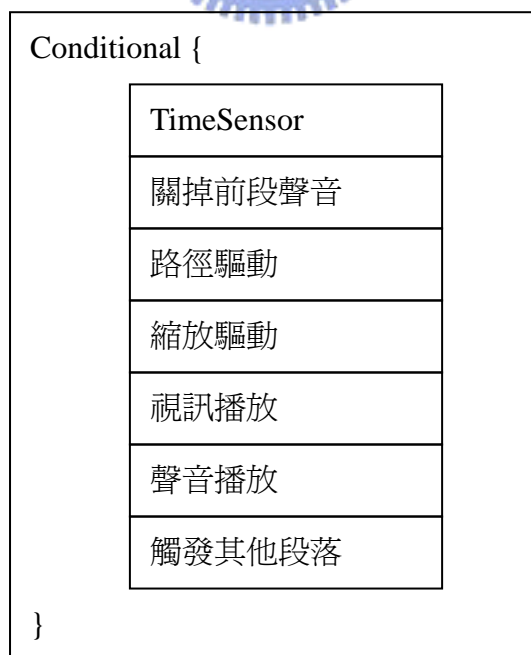


圖 43 物件播放 BT script 程式碼概念圖

總結來說，互動劇情子模組產生與劇情相關的 BT script 程式碼，並且，相對於靜態顯示子模組是設定不會動的靜態物件，互動劇情子模組是根據劇情的訊息，驅動劇情內各物件演出的動作。

4.3.4 進階設定子模組

進階設定子模組主要產生三種功能的 BT script 程式碼，分別是物件演出前後消失或出現、物件演出結束的依據、以及物件深度的設定。

物件演出前後消失或出現

經過靜態顯示子模組和互動劇情子模組產生 BT script 程式碼後，已經可以呈現出物件演出的樣子了，可是，物件不管是演出前或是演出後，都持續地出現在場景中，並沒有其它的行為，變化性略顯不足。因此，進階設定子模組針對這個缺點，額外產生一些 BT script 程式碼，來對物件演出前後的行為做一些變化。

由於我們選擇的視覺化編輯工具，也就是智勝國際編輯手，對物件演出前後的行為已經有某些規範，例如，在編輯手中可以針對物件演出之前，設定此物件是消失還是出現，而演出之後也一樣可以設定消失或出現，因此，本研究的進階設定子模組提供同樣的功能，以 BT script 程式碼表達物件演出前後消失或出現的行為。

我們在互動劇情子模組產生的物件播放 BT script 程式碼中，加入演出前初始化區塊、演出後消失區塊，並在劇情的 BT script 程式碼中，修改初始化區塊，增加處理演出前消失的程式碼，來表現物件演出前後消失的行為，如圖 44、圖 45 所示。演出前初始化區塊是幫具有演出前消失性質的物件重新貼圖，這樣此物件在演出的過程中才不是消失的狀態，演出後消失區塊則以 conditional 節點構成，使得物件演出後消失的程式碼可以包含在此區塊中，最後，至於在劇情的初始化區塊所新增的程式碼，則是達成演出前消失的功能。

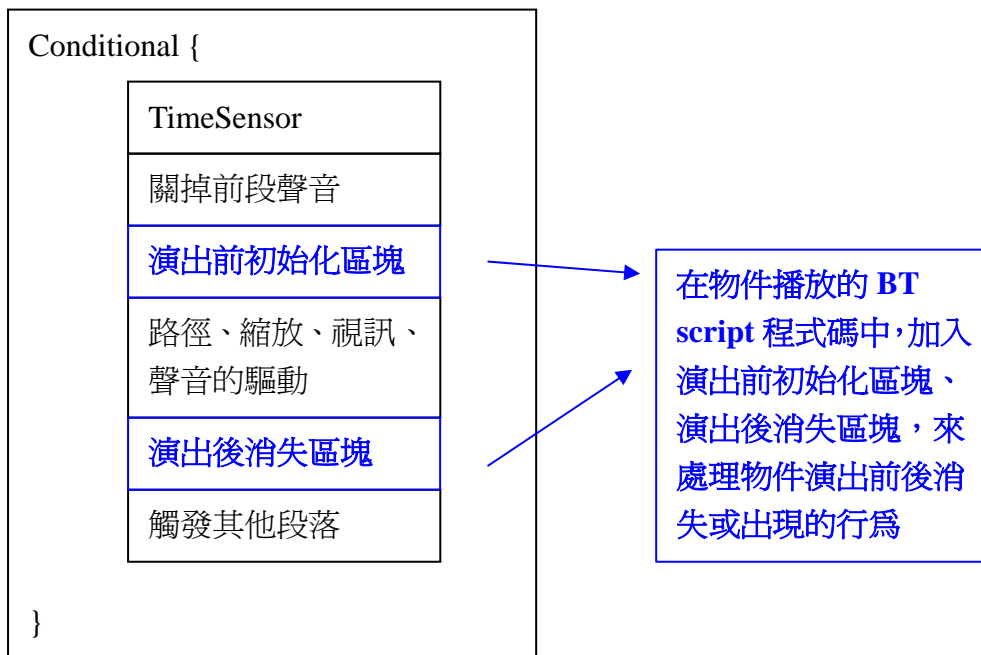


圖 44 加入演出前初始化區塊、演出後消失區塊

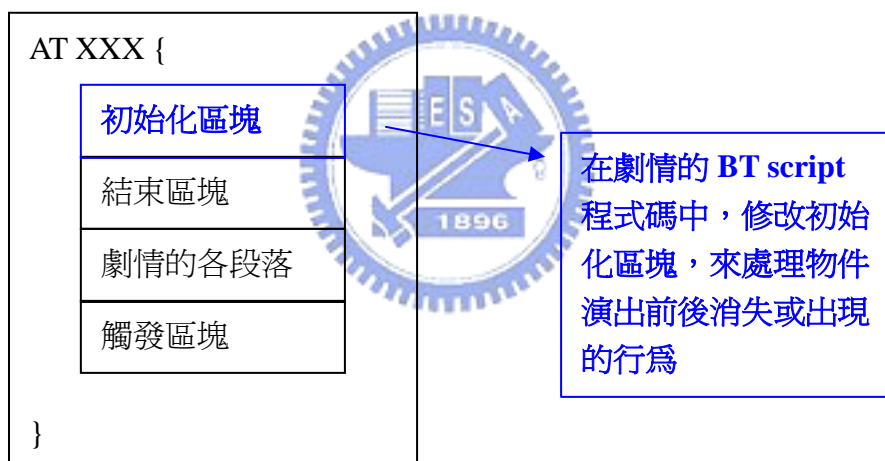


圖 45 修改初始化區塊的 BT script 程式碼

物件演出結束的依據

在編輯手中有三種物件演出結束的依據，分別是物件的路徑演完結束、物件的聲音演完結束、以及路徑與聲音皆演完結束。在本研究產生的 BT script 架構中，是以物件路徑演完與否作為演出結束的依據，為了讓此架構能模擬出聲音演完結束、路徑與聲音皆演完結束的功能，因此，我們比較物件的路徑與聲音的演出長度，並且根據使用者所選擇的演出結束依據，替物件的路徑做增加路徑點或減少路徑點的動作，如此一來，便可在不改變原本的架構下，達到模擬聲音演完結束、路徑與聲音皆演完結束的功能。

物件深度的設定

場景中所有物件都有各自深度的關係，所謂物件的深度，就是指物件之間若在場景中顯示位置重疊時，哪個物件在上哪個物件在下的關係。在 BT script 語法中，寫在越後面的物件程式碼其深度是較淺的，因此，爲了忠實呈現在編輯手中物件之間深度的關係，進階設定子模組會對靜態顯示子模組產生的場景部分 BT script 做排序的動作，將較淺的物件移至程式碼的後半部，將較深的物件移至程式碼的前半部，如此一來，便可呈現場景原本的物件深度關係。

4.3.5 XML to BT 子模組

我們在 4.2.3 節中有提到需要 XML to BT 子模組的原因，在此節則以一例子更清楚的顯示轉換前後的不同。

轉譯器產生的所有程式碼會暫存在 XML 元件中，其外觀類似圖 46 左半部的樣子，是以 XML 的格式暫存，而一旦轉成 BT script 後，就變成圖 46 右半部的 VRML 格式了。

由於 XML to BT 子模組的功能是轉換 XML 格式與 VRML 格式，與對應視覺化編輯工具的能力而產生 BT script 程式碼無關，因此，此子模組在實作上以 dll 的方式撰寫，若有其他的專案需要 XML 格式與 VRML 格式轉換的功能時，便可以直接呼叫 dll 檔，達到元件重新再利用的目的。

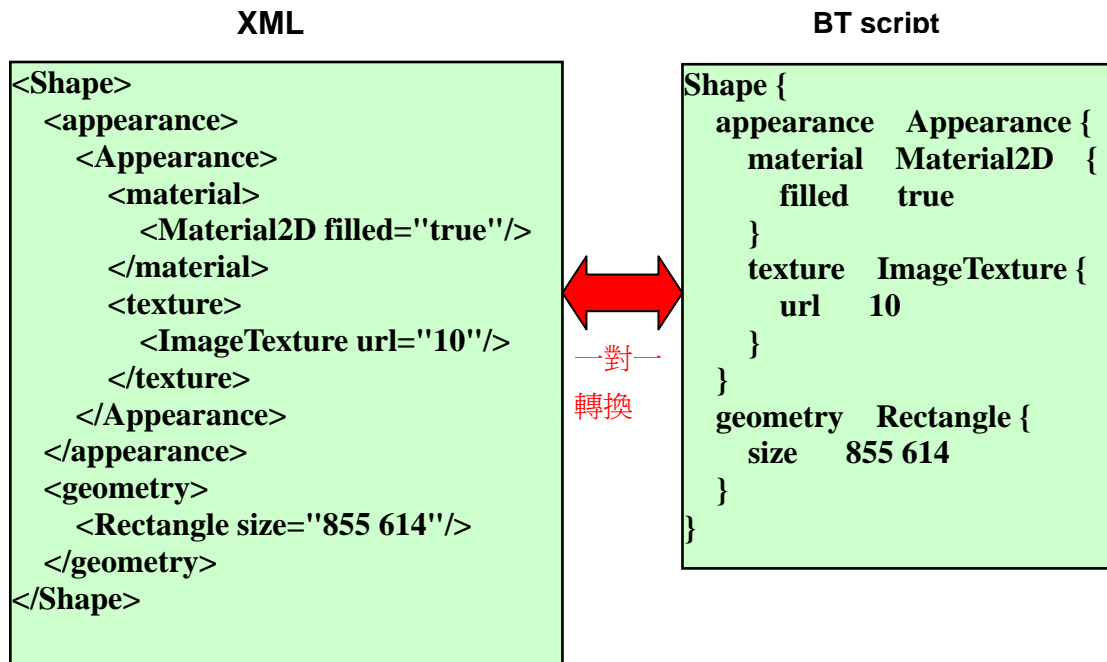


圖 46 XML 格式轉換成 VRML 格式的 BT script



五、系統展示

5.1 操作流程

本研究製作一個適合於沒有撰寫程式專業能力使用者的 MPEG-4 編輯工具，其操作的流程與系統的外觀，將在本章做一個介紹。首先，使用者利用智勝國際編輯手，編輯出場景的內容，其操作順序如圖 47 至圖 51，編輯完成後再利用轉譯器可得到適合於 PC 環境與 PDA 環境兩種不同大小的 MPEG-4 檔案，如圖 52 至圖 56。

下面我們舉出一個編輯的範例，假設使用者欲編輯一個 story，並將此 story 轉換成 MPEG-4 檔案格式。

(1) 在編輯手的教材管理介面中，選取欲編輯的 story，或是新增一個空白的 story。

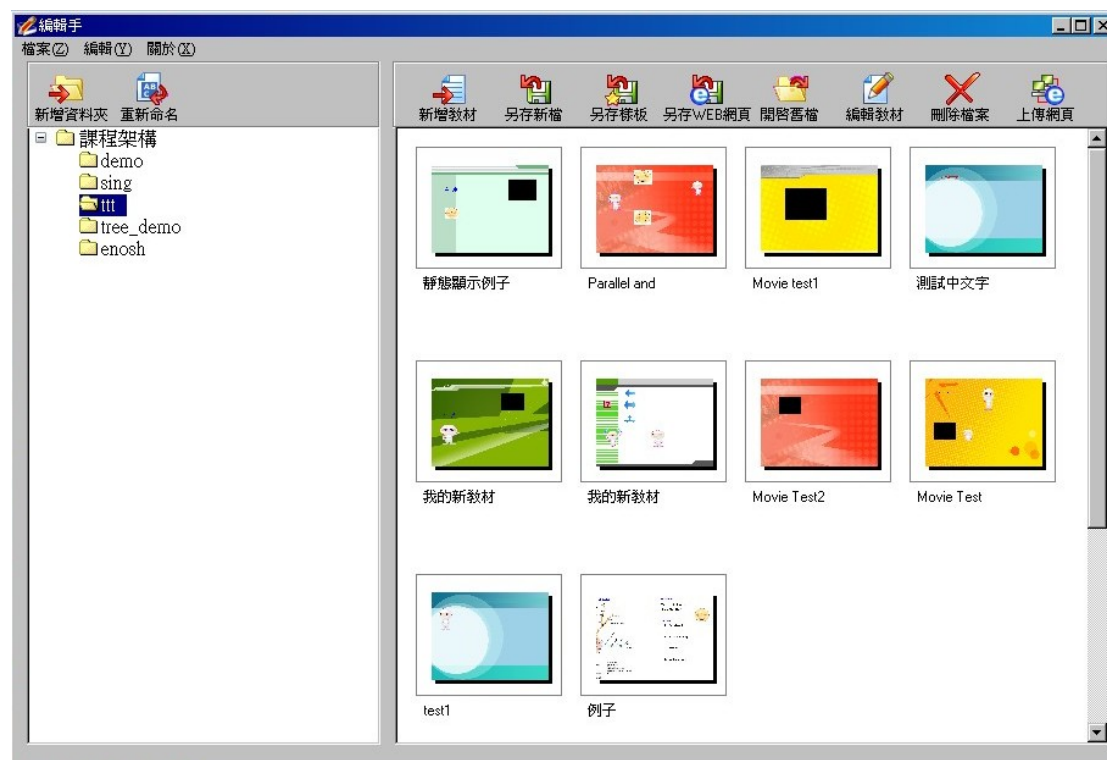


圖 47 編輯手—教材管理

(2) 選定 story 之後，進入場景設定的步驟，在場景設定時，可以設定場景的背景圖片、背景音樂，會以圖 48 右邊的素材視窗來輔助使用者的選取。

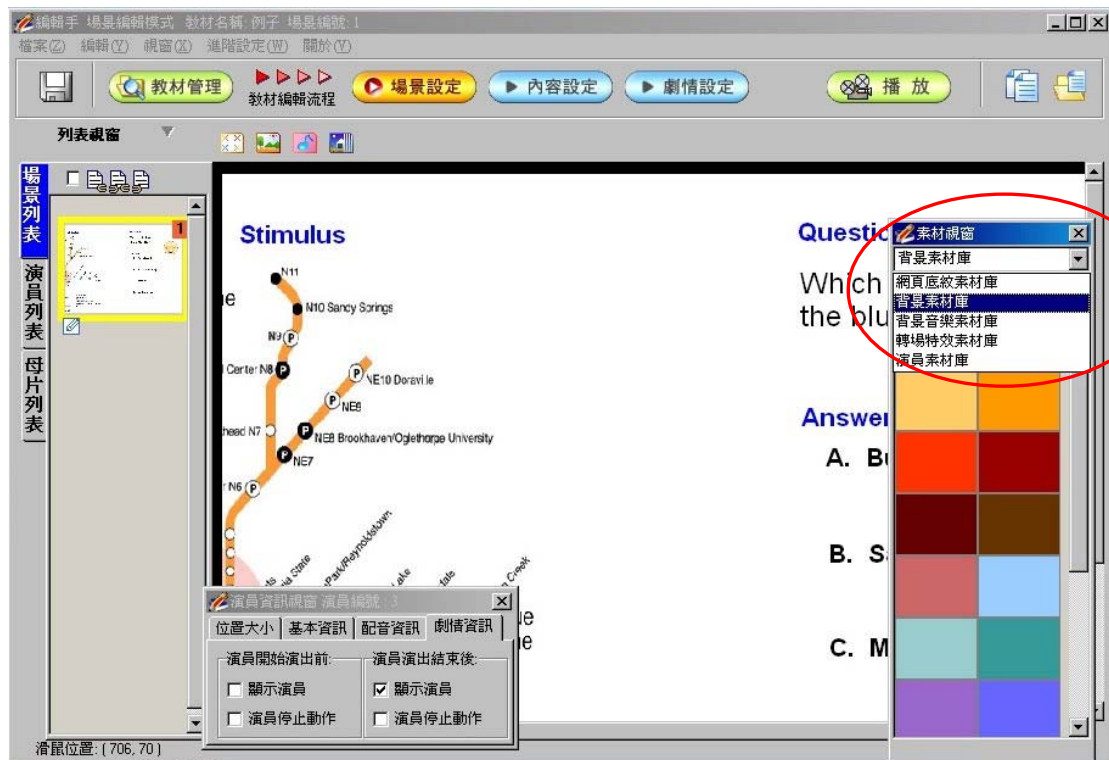


圖 48 編輯手一場景設定

(3) 接著，開始設定場景的內容，此時素材視窗會提供動畫、文字、影片等各式各樣的物件，並且以圖 49 上方的工具列進行編輯的動作，此外，演員資訊視窗可以對物件做進階設定，例如演出前後消失或出現的設定就可以在此選擇，最後，圖 49 左方的演員列表可以顯示場景內所有物件的內容，方便使用者選取。

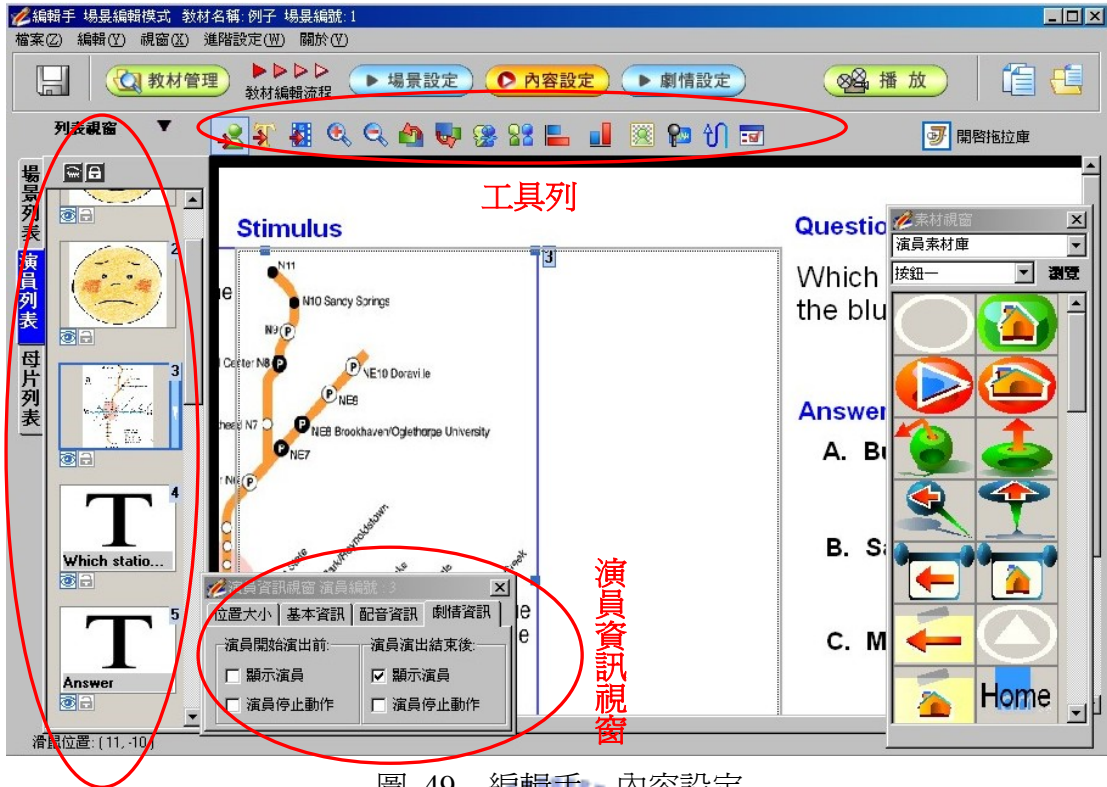


圖 49 編輯手—內容設定

(4) 再來就是劇情的設定。劇情設定可以幫助使用者完成場景劇情的編輯，圖 50 上方的圖示就是劇情編輯的區塊，提供劇情觸發方式的選擇，以及每個劇情段落類型的設定等功能。

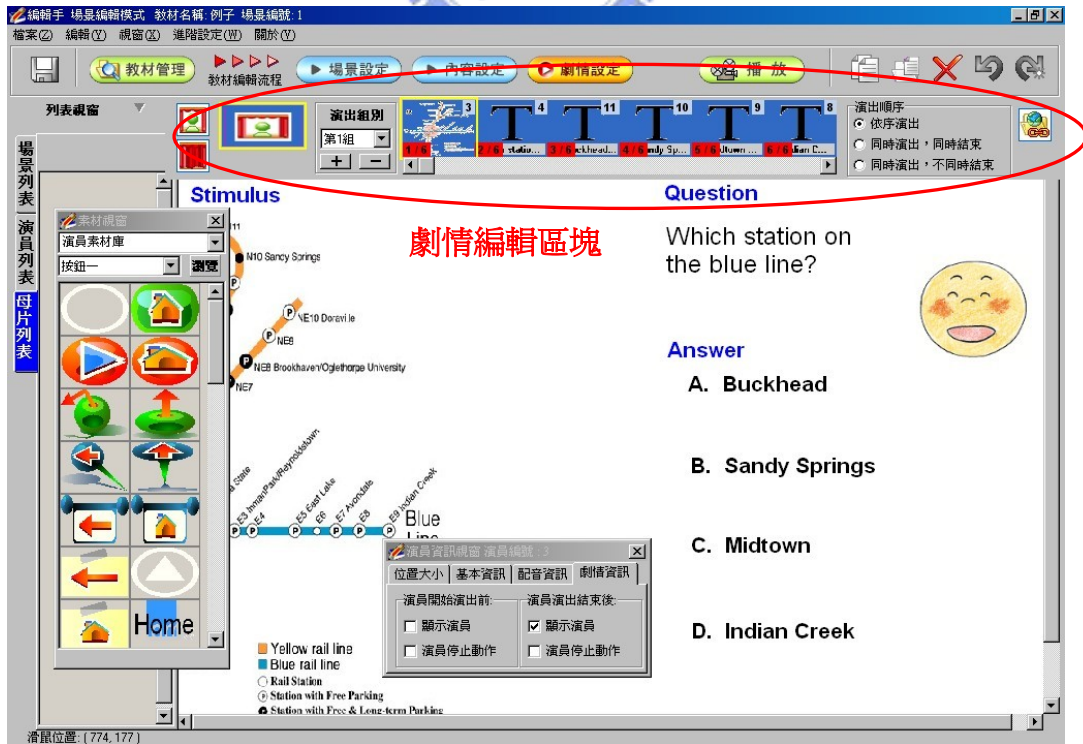


圖 50 編輯手—劇情設定

(5) 圖 51 是一個由編輯手製作完成的例子。

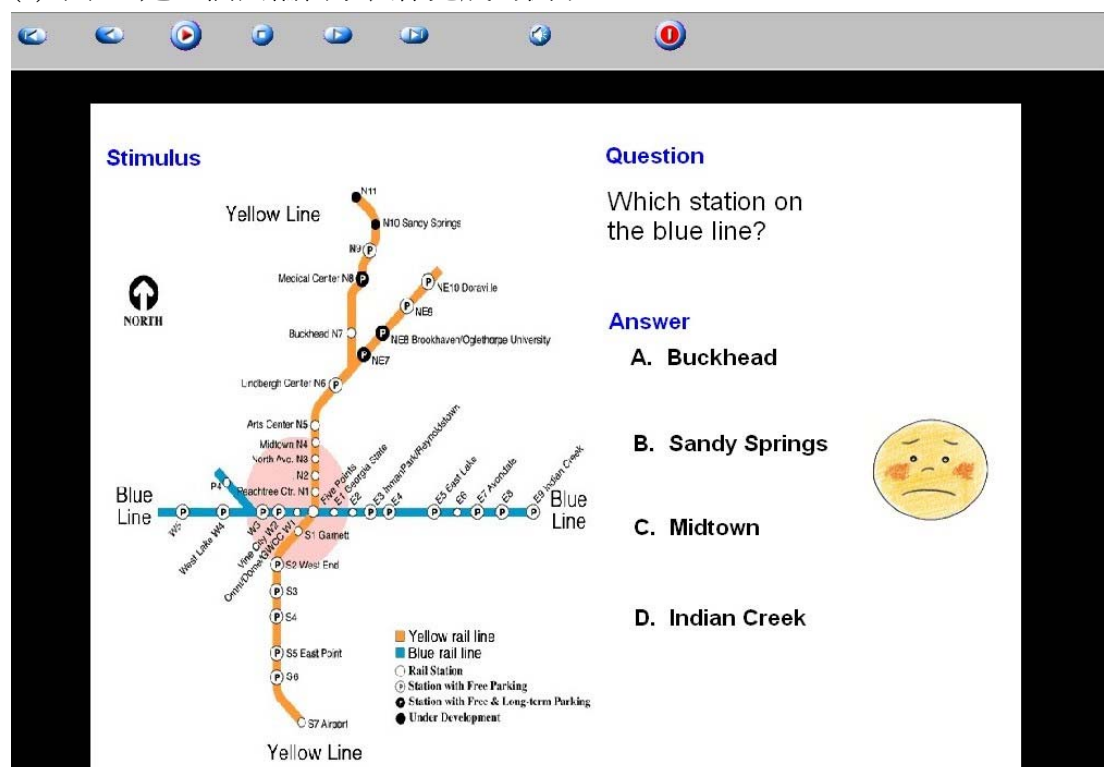


圖 51 編輯手一編輯完成的結果

(6) 當使用者編輯完成後，回到教材管理的介面中，按下圖 52 上方的轉換按鈕，即可呼叫轉譯器，來進行轉換成 MPEG-4 檔案的動作。



圖 52 編輯手一存成 MPEG-4 檔案

- (7) 轉譯器首先會出現使用者介面讓使用者進行輸入及選擇，圖 53 中間是讓使用者輸入 MPEG-4 檔案產生的目錄位置，下方則讓使用者選定輸出的類型，有 PC 版本與 PDA 版本可選擇。

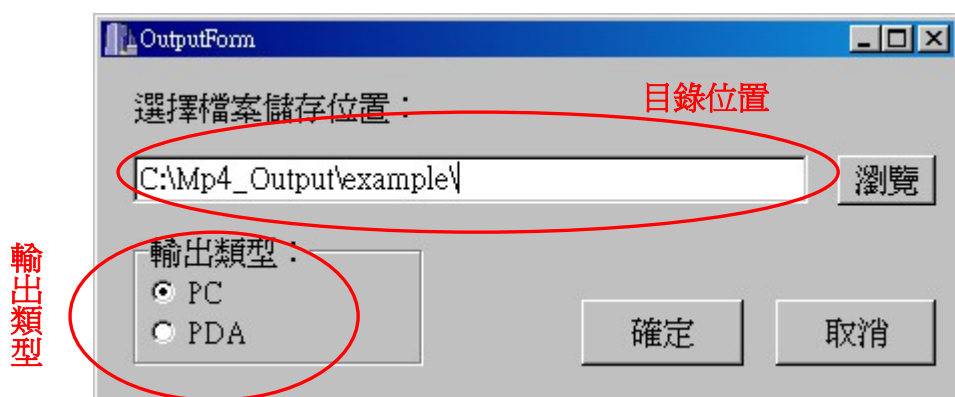


圖 53 轉譯器—使用者介面

- (8) 在轉換的過程中，會出現顯示進度的橫條，如圖 54 所示，並且在轉換完畢後，出現詢問的視窗，詢問使用者是否啓動 MPEG-4 播放器。



圖 54 轉譯器—轉換進度之顯示

(9) 圖 55 是圖 51 的例子在 PC 環境的 MPEG-4 播放器播放的樣子。

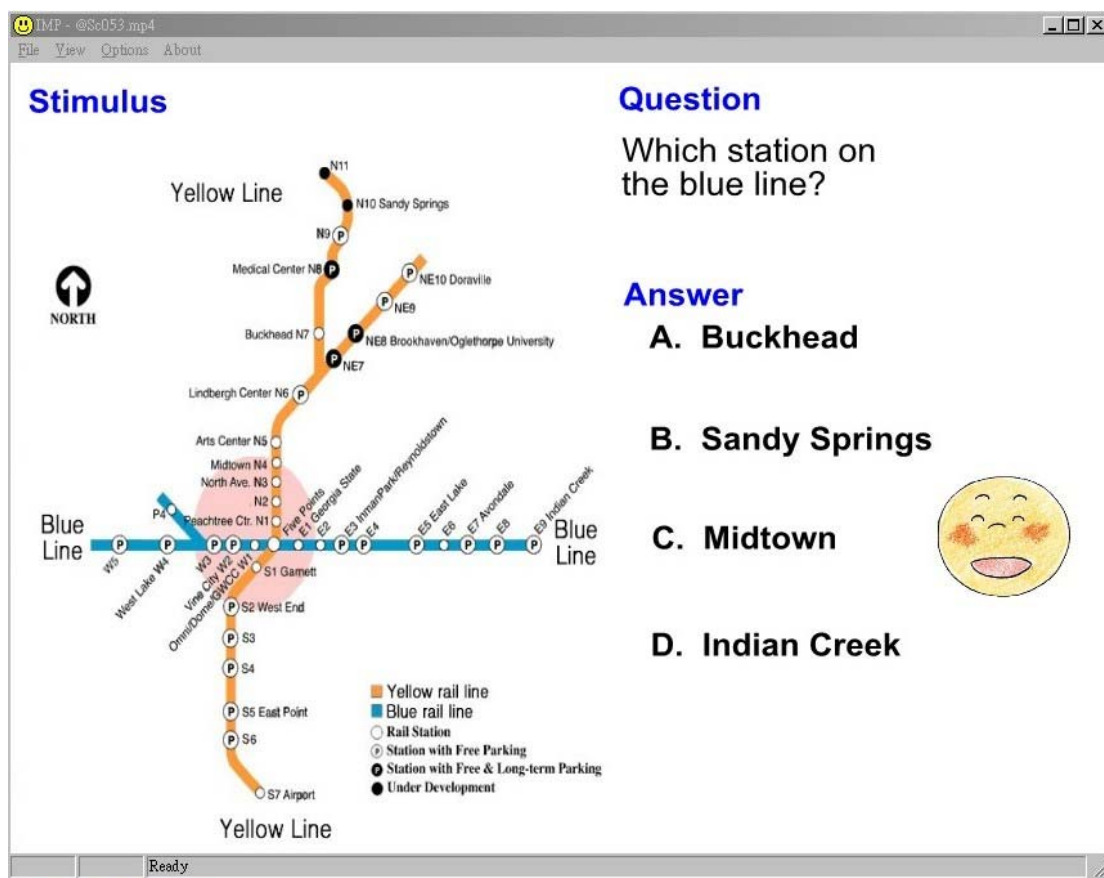


圖 55 PC 版本在播放器上之結果

(10) 圖 56 是圖 51 的例子以 PDA 版本輸出的樣子。

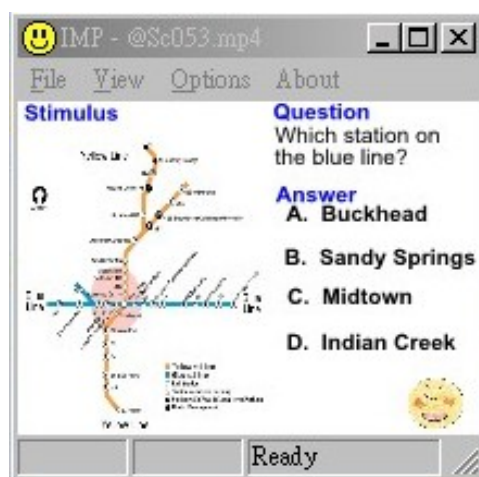


圖 56 PDA 版本在播放器上之結果

5.2 大小比較

將互動式多媒體轉為 MPEG-4 檔案，可以降低互動式多媒體的檔案大小。表 3 顯示出編輯手的多媒體原始大小，以及轉換成 MPEG-4 後的檔案大小差異。第一個例子”瀑布”是一個英文試題的多媒體，由圖片文字及聲音所構成，是一個較常見的多媒體形式，第二個例子”影片”主要是測試影片方面的支援，由兩個短片所組成，第三個例子是節錄自英文歌教唱例子的第一幕，場景主要的構成元素是聲音。

表 3 編輯手 story 與轉換後的 MPEG-4 大小之比較

	編輯手 story	MPEG-4
瀑布	3.14MB	395KB
影片	5.58MB	3.32MB
教唱(第 1 幕)	8.87MB	906KB

表 4 編輯手 story 與轉換後的 MPEG-4 大小之比較---直方圖

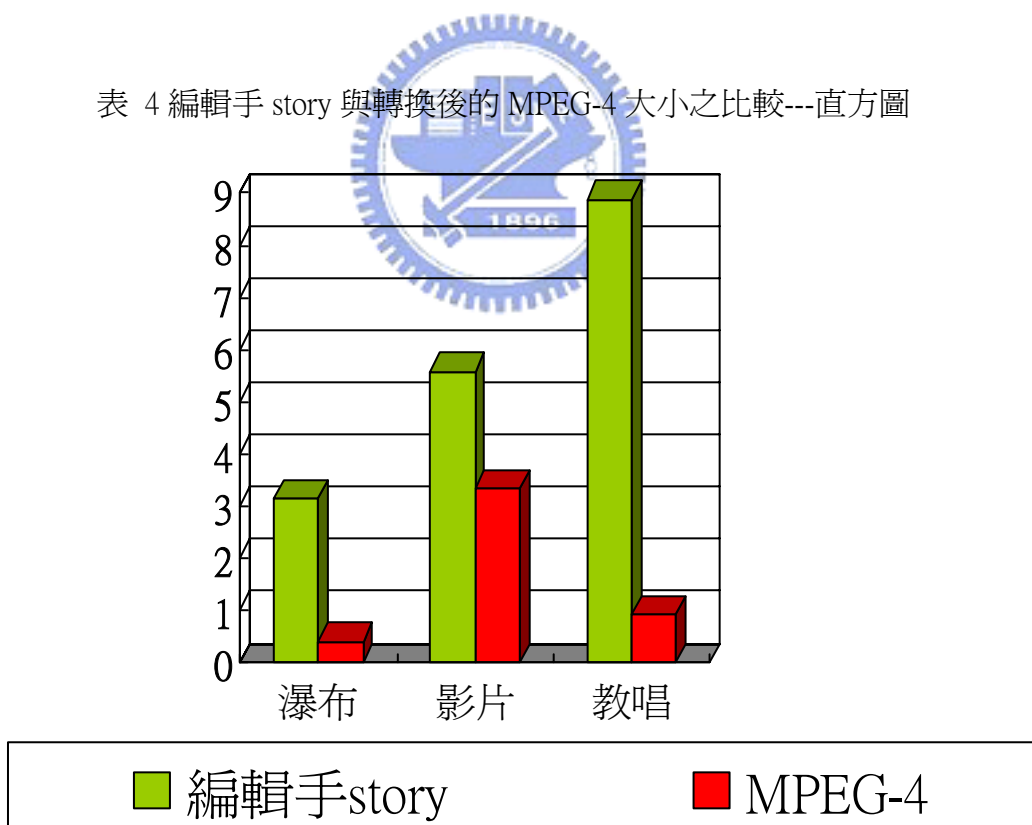


表 4 以直方圖的方式表現兩者檔案大小的差異性。我們可以發現三個例子在經過轉換後，產生的 MPEG-4 檔案比起編輯手原始的檔案，都有明顯降低檔案大小的效果。

六、結論

6.1 本研究特點

本研究針對 MPEG-4 需求者不易製作 MPEG-4 檔案的缺點，結合視覺化編輯工具與轉譯器，達到降低 MPEG-4 檔案製作門檻的目的。因此，本研究具有下列的特點：

✚ 適合沒有撰寫程式專業能力的使用者：

由於本研究在視覺化編輯工具的挑選上特別注意，以方便沒有撰寫程式專業能力的使用者製作 MPEG-4 檔案為原則，可以說免除了 MPEG-4 需求者找程式設計師代工製作 MPEG-4 的問題，使得製作 MPEG-4 檔案再也不是程式設計師的專利。

✚ 不需撰寫程式：

本研究解決製作 MPEG-4 檔案必須瞭解 MPEG-4 場景概念的缺點，透過視覺化編輯工具的編輯，以這些拖拉點選的操作來隱藏 MPEG-4 的場景概念，因此使用者不需撰寫程式便可製作 MPEG-4 檔案。

✚ 所見即所得：

利用視覺化編輯工具來編輯 MPEG-4 檔案，可以達到所見即所得的能力，MPEG-4 需求者再也不需要等待程式設計師的製作完成，在編輯的時候就可以看到 MPEG-4 檔案呈現的效果，方便 MPEG-4 需求者的修改。

✚ 容易且快速：

利用本研究提供的方法，可以簡化 MPEG-4 檔案的製作流程，以視覺化編輯工具及轉譯器作為 MPEG-4 的編輯工具，能兼具易於編輯和支援 MPEG-4 格式的目的，因此 MPEG-4 需求者可以更容易且更快速的製作出 MPEG-4 檔案。

6.2 未來發展

編輯手的限制：

本研究的視覺化編輯工具是智勝國際公司的編輯手，以下是一些 MPEG-4 所具備的特點，而編輯手尚未支援的項目，這些尚未支援的項目會限制本研究 MPEG-4 編輯工具的能力，因此未來可以針對這些項目做加強。

■ 3D 的支援：

MPEG-4 標準有對 3D 的支援，本研究因編輯手不具備 3D 的功能，因此並沒有對 3D 功能加以支援。若未來可以增加編輯手 3D 的編輯環境，那麼就可以對轉譯器加入 3D 的支援，來達到完整支援 MPEG-4 標準的目的。

■ 合成物件的顯示：

編輯手著重的是各類型多媒體的整合，所以圖片、影片、聲音這些自然物件在編輯手中獲得相當好的支援。但是像矩形、三角形、圓形這些合成物件，在編輯手中獲得的支援不如自然物件，因此，顯示合成物件的能力亦是編輯手應加強的部分。

■ 合成物件變換顏色：

MPEG-4 提供合成物件變換顏色的功能，利用互動或其他方式，可以觸發合成物件顏色的變換。這個變換顏色的效果，必須與顯示合成物件的能力互相配合，才能達到相輔相成的作用。

■ 物件的旋轉：

在 MPEG-4 中，定義了一些提供旋轉功能的節點，這些節點可以讓物件沿著某個軸心做旋轉的動作，並具有觸發物件旋轉的功能。若編輯手支援物件旋轉的能力，便可以用這些節點展現出相同的效果。

■ 物件的拖曳：

所謂物件的拖曳，就是使用者可以利用滑鼠，拖曳場景內的物件，進而改變物件的位置。MPEG-4 提供拖曳物件的功能，這樣一來，可以提高場景的豐富性。

轉譯器的調整：

視覺化編輯工具所編輯出來的場景是轉換成 MPEG-4 檔案的源頭，當視覺化編輯工具增加新的能力後，轉譯器如果維持不變，那麼新增的功能就不會在 MPEG-4 檔案上看到，因此，若視覺化編輯工具增加新的能力，則轉譯器要做相對應的調整，來顯示新的能力。



參考文獻

- [1] ISO/IEC 14496-1:2002 , Coding Of Audio-Visual Objects: Systems , March 2002
- [2] 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
- [3] Mihai Burlacu & Sonja Kangas, "MPEG-4 Technology Strategy Analysis", Helsinki University of Technology, 2.4.2003
- [4] Meng-Jyi Shieh, "Design and Implementation of an MPEG-4 System", N.T.U Taiwan, 2003
- [5] Chen-Kuo Chiang, "Stream Management and object Descriptor Framework Design and Implementation for MPEG4 System", N.T.U Taiwan, 2000
- [6] Julien Signès & Yuval Fisher & Alexandros Eleftheriadis, "MPEG-4's Binary Format for Scene Description"
- [7] Fernando Pereira, "MPEG-4: Why, What, How and When?"
- [8] GACP , <http://gpac.sourceforge.net/index.php>
- [9] Doom9's Forum , <http://forum.doom9.org/index.php>
- [10] MPEG-4 @ ENST ,
<http://www.comelec.enst.fr/~dufourd/mpeg-4/index.html>
- [11] 嚴子翔, VRML 虛擬實境網頁語言, 知城數位科技股份有限公司, 2001 年
- [12] 陳燦煌, C++ Builder 6 徹底研究, 博碩文化股份有限公司, 2002 年
- [13] 太極digimax , <http://www.digimax.com.tw/MPEG4/>
- [14] ETRI ,
http://www.etri.re.kr/e_etri/intro/newtech/etri21c_32_2003.html
- [15] 資策會多媒體實驗室 , <http://www.iii.org.tw/special/>

- [16] IBM MPEG-4 Technologies ,
<http://www.research.ibm.com/mpeg4/indexjs.htm>
- [17] 3C整合政策推行小組 ,
http://3c.nii.org.tw/3c/silicon/silicon_index.htm
- [18] 李昶慶 , 影音世代關鍵技術－MPEG4 ,
<http://www.eedesign.com.tw/article/Document/dc845.htm>
- [19] 智勝國際創意網站 , <http://www.caidiy.com.tw/caidefault.htm>
- [20] Delphi K.Top討論區 , <http://delphi.ktop.com.tw/>
- [21] 葉京荃, " 支援視覺化編輯工具所產生之互動式多媒體內容在 PDA 環境下的 MPEG-4 播放器製作" ,N.C.T.U Taiwan, 2005
- [22] Extensible Markup Language , <http://www.w3.org/XML/>

