

計畫編號：PG9304-0785

數位廣播電視系統整合之研究

研究報告

委 託 單 位 ： 交 通 部 電 信 總 局
研 究 單 位 ： 財 團 法 人 工 業 技 術 研 究 院
中 華 民 國 九 十 三 年 十 一 月

計畫報告書脊

「數位廣播電視系統整合之研究」 研究報告

委託單位：交通部電信總局
研究單位：財團法人工業技術研究院

出版機關：交通部電信總局

計畫編號：PG9304-0785

數位廣播電視系統整合之研究

研究報告

委 託 單 位 ： 交 通 部 電 信 總 局
研 究 單 位 ： 財 團 法 人 工 業 技 術 研 究 院
中 華 民 國 九 十 三 年 十 一 月

目 錄

摘要 Abstract	2
壹、研究背景	4
貳、數位廣播電視系統國際MHP標準與市場現況	7
參、其他相關數位電視多媒體平台現況	39
肆、我國數位廣播電視頭端系統平台特性研究與整合要素分析	51
伍、廣播電視數位視訊電視網路(Network)及數位頻道服務碼代碼管理(ID)統一 管理規範及註冊機制	56
陸、廣播電視業者之頭端編碼系統與條件式接取系統產生器間之統一介面格式與 軟體協定建議	
柒、我國數位視訊共通平台建置整合之方案研究及建議	66
參考文獻	85
附件	
(一) 技術規格摘錄	
(二) 出國報告	
(三) 會議紀錄	
(四) DVB SSU 與 EuroLoader	
(五) 其他	

摘要

數位電視已於九十二年全區開播，數位視訊平台亟需整合，以免未來民眾須重複購置多台數位機上盒，或須使用多張授權卡導致轉臺時收視之不便。因此，國內當務之急在於推動法規修訂、數位視訊平台之關鍵性共通規範之引進、視訊平台產業技術研發與標準等。

本計畫將進行瞭解我國廣播電視數位系統參數整合要素、各系統間差異性，並提出整合具體方案。研擬無線、有線、衛星廣播電視系統共通平台(MHP) 建置之整合方案。針對所有廣播電視網路各類識別碼之管理，研擬統一管理規範，提出註冊機制，使每個網路有所區分，促使服務頻道在不同網路間有一致之識別。並研擬廣播電視系統端同步加密建議標準。尋求我國數位視訊平台整合發展之適合且較佳之方式。

Abstract

The debut of digital television broadcasting in domestic market has been launched in 2003. The digital video platforms need to be further consolidated, so that the consumers would not have to be forced to purchase many different set-top boxes, or to get many smart cards, in order to receive different programs from various system operators. Therefore, the very first things to do are the renewing of associated regulations, the introduction of key common paradigms for digital video platforms, video platform technology research and development and standardization, etc.

In this project, we are going to explore the integration parameter ingredients of the Digital TV Broadcasting System in domestic market, understanding the differences among various kinds of systems, and to propose a consolidation plan. A common platform for the integration of terrestrial, cable, and satellite broadcasting systems will be proposed. A registration mechanism for an unified management paradigm to identify different networks will be built, and also the simulcrypt encryption standard. Hopefully, this project will benefit the integration of our digital video platforms.

壹、研究背景

2003 年起，愛爾蘭、挪威、法國、丹麥、義大利、芬蘭、瑞典、西班牙、德國、荷蘭等多個國家陸續開播數位電視，所採用的都是歐規 DVB MHP 標準，主要是以有線系統業者為主軸，僅在德國為地面無線電視台。另者日本及中國地面數位電視標準及廣播時程將逐漸明朗化。尤其在中國已有上海交大所屬子公司高清數字科技的研發團隊正進行 MHP 中介軟體 (Middleware) 規範之訂定及研發。我國數位電視發展，在歷經多年努力後，亦於九十二年四月十八日，完成台灣地面數位電視全區開播，正式將台灣數位電視產業推向新的里程碑。預期全球數位電視市場在未來 2 年將邁向高度成長期。

國際數位視訊匯流的發展已然蔚為趨勢，盱衡國際標準日趨成熟，惟國內視訊平台尚缺乏健全之法規、確切之標準及整合之環境，亟待產官學研各界積極投入，殫精竭慮，共謀對策，戮力推動，方期有成。

行政院國家資訊通信發展推動小組鑑於我國視訊產業網路、服務、產品匯流，朝網路寬頻、服務互動、產品互通（共通標準）等整合發展之趨勢，並為因應數位視訊製播分離與共通平台兩項重大變革，於九十二年三月成立 NICI 視訊整合指導小組，由蔡政務委員清彥擔任召集人，下設「應用服務」、「平台傳輸」、「終端設備」及「綜合業務」等分組分工推動。其中「平台傳輸分組」由交通部張次長家祝擔任分組召集人，電信總局吳副局長嘉輝擔任執行秘書，於九十二年業已舉行兩次分組會議。

行政院國家資訊通信發展推動小組（簡稱 NICI 小組）九十二年三月三十一日第八次委員會議暨行政院視訊整合指導小組九十二年八月十五日第二次會議討論台灣數位視訊發展之推動政策與方向，會議結論由平台傳輸分組推動數位視訊平臺傳輸之發展，交通部分工執行「研究評估與規劃廣播電視系統數位共通營運平台之建置」及「推動引進廣播電視系統條件式接取系統同步加密共通國際標準 (simulcrypt)」。

行政院國家資訊通信發展推動小組視訊整合指導小組並於九十三年元月二日第三次會議主席裁示略以：有關數位電視多媒體共通平台（MHP）標準，請電信總局邀集經濟部相關部會研商一致的標準。

交通部電信總局為配合未來行政院之數位電視政策暨數位電視共通平臺標準之相關政策，於九十三年元月規劃在平台傳輸分組下設置「數位電視多媒體共通平台技術標準諮詢委員會」，同月奉平臺傳輸分組分組召集人交通部張次長家祝准予成立，前揭委員會由交通部電信總局吳副局長嘉輝擔任召集人，邀集產官學研各界與會共同商議。

交通部電信總局於九十三年二月十八日舉行「數位電視多媒體共通平台技術標準諮詢委員會」第一次會議，召集人吳副局長嘉輝致詞略以：...盱衡平台傳輸及未來經營管理，將涉及條件式接取（Conditional Access）電子商務、安全保密等系統技術，爰於本分組下設置「數位電視多媒體共通平台技術標準諮詢委員會」。冀透過此機制先行諮詢專家意見並完成協調性工作。如此一方面電信總局配合電信國家型計畫辦公室組織編列三年預算之執行，另一方面結合電信技術中心廣電技術組等機制，共議策訂數位視訊整合之階段性執行方案，乃本諮詢委

員會成立之宗旨。

交通部電信總局為配合行政院 NICI 小組之數位電視政策，爰據以規劃本「數位視訊整合－數位廣播電視系統整合之研究」，初步研究數位視訊平臺整合之相關標準，以推動廣播電視系統數位電視之發展。本案由電信總局於九十三年第一季經過公告等作業後審議決標，委託財團法人工業技術研究院電腦與通訊工業研究所研究辦理，研究期間自九十三年四月至九十三年十一月。

工業技術研究院電通研究所之研究方法及進行步驟

財團法人工業技術研究院電腦與通訊工業研究所在本案之研究方法及進行步驟方面，配合「數位電視多媒體共通平台技術標準諮詢委員會」會議，研提會議討論項目、資料與技術標準內容。蒐集國際 MHP 標準制訂流程、參與國家、國際市場現況、主要架構、內容與功能等資料，進行國際數位電視多媒體平台相關標準現況分析（含差異性、相似性、互通性），瞭解我國數位廣播電視系統平台建置現況（含我國無線、有線、衛星、DAB 及 MOD 等現有數位系統），頭端系統平台特性、差異性與相似性及其他有關整合特性之分析研究，平台整合要素分析研究，藉以提供「數位電視多媒體共通平台技術標準諮詢委員會」產、官、學、研各界專家參考，建立數位電視多媒體共同平台相關技術標準之共識，俾有利於共通營運平台之建置，及廣播電視系統條件式接取系統同步加密共通標準（simulcrypt）之推動。

具體之進行步驟為：

* 配合「諮詢委員會」每次會議，研提會議討論項目、資料與技術標準內容

將結合「台灣數位電視產業聯盟」相關之技術發展與標準制訂工作小組，召開會議與各界專家學者討論共通平台技術與標準項目，所得結果提報「諮詢委員會」會議。

* 蒐集國際 MHP 標準制訂流程、市場現況、主要架構、內容與功能等資料

參與國際 DVB 標準會議，並結合「台灣數位電視產業聯盟」相關之國際交流與市場情報工作小組，收集 MHP 標準相關資料，瞭解國際上市場現況。

* 蒐集國內外數位電視多媒體平台相關標準現況（含差異性、相似性、互通性）

結合「台灣數位電視產業聯盟」相關之國際交流與市場情報工作小組，收集國內外數位電視多媒體平台相關標準資料，瞭解其差異性、相似性、互通性，但是因資料取得管道有限，可能無法收集到齊全的內容。

* 我國數位廣播電視頭端系統平台特性、差異性與相似性及其他有關整合特性之完整分析研究（含我國無線、有線、衛星及 MOD 等系統）

收集相關公學會意見，了解各系統平台特性、差異性與相似性，歸納分析相關資料，做成比較表。可能的困難在於有些資料牽涉商業機密，無法取得完整資料。

* 我國數位廣播電視系統平台整合要素分析研究暨整合具體方案研究（含無線、有線、衛星、MOD 等系統）

根據前述分析所得資料，研提整合方案。可能的困難在於有些資料牽涉商業機密，無法取得完整資料。

* 我國數位廣播電視（含無線、有線、衛星、MOD）系統平台整合之整體規劃（含 93 年至 95

年階段執行工作項目規劃、處理建議項目等)

根據前述分析所得資料，進行系統平台整合之整體規劃。可能的困難在於有些資料牽涉商業機密，無法取得完整資料。

* 無線廣播電視數位系統共通平台建置整合之方案研究及建議規範(含 MHP 系統服務規格標準草案，依功能分級)

瞭解業者有關 MHP 應用服務時程及終端業者產品開發時程，依據此資料提出系統共通平台建置整合之方案及建議規範。可能的困難在於有些資料牽涉商業機密，無法取得完整資料。

* 完成有線、衛星廣播電視數位系統共通平台建置整合之方案研究報告及建議規範(含 MHP 系統服務規格標準草案，依功能分級)

瞭解業者有關 MHP 應用服務時程及終端業者產品開發時程，依據此資料提出系統共通平台建置整合之方案及建議規範。可能的困難在於有些資料牽涉商業機密，無法取得完整資料。

* 完成數位視訊系統平台中介軟體功能及應用之研究

瞭解數位視訊系統平台中介軟體規格，彙整提出相關功能及應用之研究報告。

* 完成廣播電視數位視訊電視網路(Network)及數位頻道服務碼代碼管理(ID)統一管理規範及註冊機制

透過與相關公學會討論，建立廣播電視數位視訊電視網路(Network)及數位頻道服務碼代碼管理(ID)統一管理規範，並協助業者向國際標準組織 DVB 提出 Network ID 註冊登錄。

* 提出廣播電視業者之頭端編碼系統(ENCODING SYSTEM / MPEG-2 MUX)與條件式接取系統產生器間之統一介面格式與軟體協定建議草案

透過與相關公學會討論，訂定廣播電視業者之頭端編碼系統(ENCODING SYSTEM / MPEG-2 MUX)與條件式接取系統產生器間之統一介面格式與軟體協定草案。可能的困難在於有些資料牽涉商業機密，無法取得完整資料。

* 提出我國無線、有線、衛星、MOD 及 DAB 數位視訊共通平台建置整合之研究報告及建議規範(含 MHP 系統服務規格標準草案，依功能分級)

根據前述分析所得資料，彙總提出我國數位視訊共通平台建置整合之研究報告及建議規範。可能的困難在於無線、有線、衛星、MOD 及 DAB 數位視訊系統平台之營運模式各不相同，不易取得共識，達到共通平台建置整合之目標。

貳、數位廣播電視系統國際 MHP 標準與市場現況

MHP 原本是歐洲 DVB (Digital Video Broadcasting) 組織，所制訂的互動數位電視的 middleware 標準，原本只適用於歐規數位電視傳輸標準 (DVB-T、DVB-C、DVB-S) 之上。在其他國家或地區，所採用的則是其他的互動數位電視的 middleware 標準。例如，在美國 ATSC 系統上，原本採用的 middleware 是 ATSC DASE (DTV Application Software Environment) 標準；在日本，所採用的 middleware 則是 BML (Broadcast Markup Language)，是由日本電波業會 ARIB，所制訂的標準 (ARIB STD-B24 – Data Coding and Transmission Specification for Digital Broadcasting)。但是，因為以下的兩個原因，目前全球制訂數位電視標準的主要標準組織，在互動數位電視的 middleware 標準上，都開始向歐規的 MHP 靠攏。第一個原因是互動數位內容程式的部分，在不同的標準間，轉換上的難度比 video 與 audio 內容的轉換來得高，所以，在數位廣播電視界，確實存在著一種對全球互動數位電視 middleware 標準的高度需求。第二個原因是歐洲 DVB 組織的 MHP 標準，在技術上相對比其他的數位電視 middleware 標準成熟，而且 DVB 也一直存在著一種將 DVB 相關的標準，推展到全世界的強烈野心。所以，在 2002 年 1 月，美國的 CableLabs 決定採用 MHP 作為其 OpenCable 標準上的 middleware 標準之後，MHP 就開始走上全球的互動數位電視 middleware 標準的盟主之路了。

GEM (Globally Executable MHP) 就是在前述的需求之下，由 DVB 所制訂的全球數位電視 middleware 標準。全球新一代的互動數位電視標準，許多都是以 GEM 為核心所制訂的衍生標準，包括：北美 cable 數位電視系統 - OpenCable 上的 middleware 標準 OCAP (OpenCable Application Platform)，美國 ATSC 系統上新一代的 middleware 標準 ACAP (Advanced Common Application Platform)、以及日本的數位電視系統上，新的 middleware 標準 ARIB-AE (ARIB Application Engine)。而且，為了讓 GEM 變成名符其實的全球標準，DVB 也已經將 GEM 標準，呈交給 ITU (International Telecommunication Union)。此標準已於 2003 年 4 月被接受，成為 ITU-T 的標準 – ITU-T Recommendation J.202。

在報告的本部分裡，我們首先會介紹歐規 DVB MHP 標準的制訂流程，以及 MHP 的系統架構與功能。接下來，我們會介紹 GEM 與 ITU-T J.202，以及 MHP 目前的全球採用狀況。由於成本是 MHP 是否能夠真的被普及化的一項關鍵，因此，我們會分析 MHP-based 的 set-top box 的成本變化的趨勢。此外，我們還會對 MHP 在互動數位電視市場上的優勢與劣勢，作一番分析 (SWOT 分析)。最後，我們會介紹 MHP 在全球的衍生標準 (GEM-based 的標準)，包括 OCAP、ACAP、以及 ARIB-AE，並簡單說明一下中國目前 DTV middleware 的發展現況。

2.1 歐洲 DVB MHP 標準簡介

MHP 的前身是歐盟委員會 (European Commission) 旗下的 ISIS program 的一個計劃 - UNITEL (Universal Set-Top Box) project。這個計劃開始於 1996 年，它所探討的是如何開發一個共同平台，讓使用者可以很方便的存取各種媒體服務。接著，在 1997 年，MHP 正式成為 DVB 的一個工作項目。同時，在 DVB 裡，成立了兩個新的工作小組：

- DVB-MHP commercial module: 負責定義使用者和市場對互動數位電視的需求。
- DVB-TAM (Technical issues Associated with MHP): 負責制訂符合 MHP commercial module 所提出的需求的 API (Application Programming Interface) 標準。

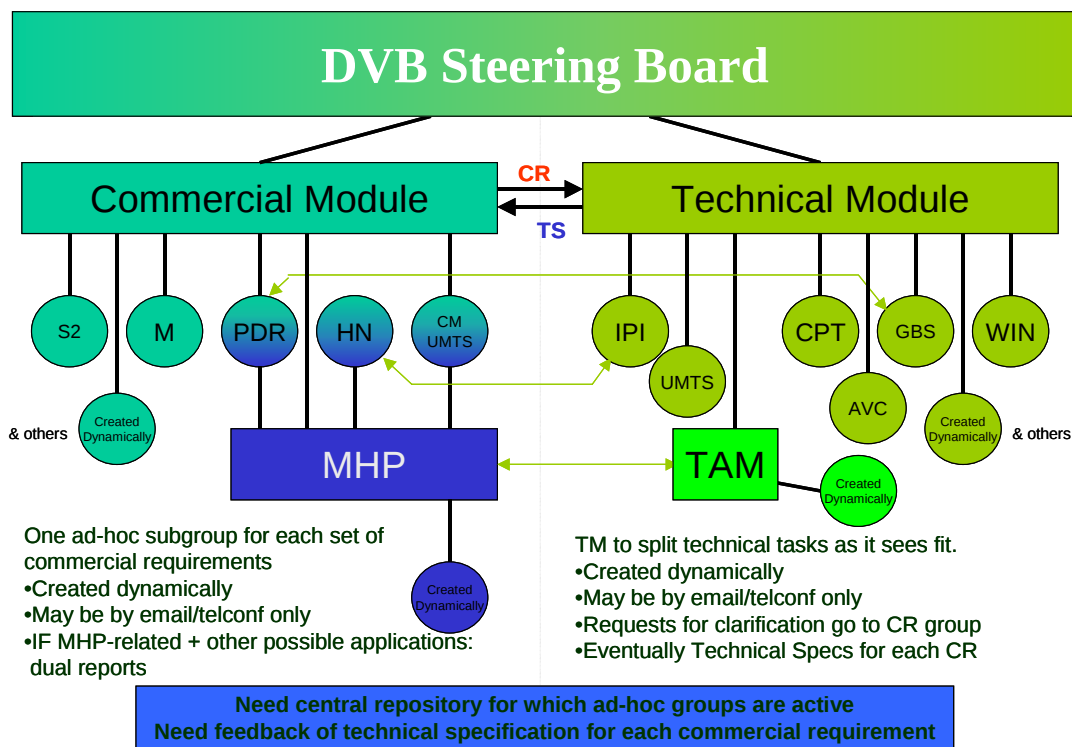


圖 2-1、DVB 內與 MHP 相關的組織圖

圖 2-1 是 DVB 的組織圖，組織圖中的最上層是 DVB 指導委員會 (DVB Steering Board)。MHP 工作小組與 TAM 工作小組，分別附屬於 DVB 的 commercial module 與 technical module 之下。由屬於 commercial module 的 MHP 工作小組向附屬於 technical module 的 TAM 工作小組提出 commercial requirement (CR，目前最新的版本是 DVB Document A062);接著，再由 TAM 工作小組根據 commercial requirement 制訂出相關的 technical specification (TS)。

DVB project 制訂一項新的標準時，必須經由以下的步驟：

1. Commercial Requirements
2. Technical Requirements
3. Requests for Detailed specifications
4. Framework (Basic Architecture)
5. First Working Draft
6. Working Draft
7. Draft Standard
8. International Standard

DVB 組織制訂任何新的標準時，一定會先考慮清楚這個標準的商業與市場需求，之後，才會制訂相關的技術標準。這也是 DVB 所制訂的標準，會在全球市場上產生相當大的影響力的原因。另外，DVB 在內部研擬出最後的標準草案 (draft standard) 之後，會再將標準草案轉呈給其他的標準制訂組織，變成所謂的國際標準 (international standard)。例如，MHP 最終被制訂成 ETSI (European Telecommunications Standards Institutes) 的標準，而 GEM 最終被制訂成 ITU (International Telecommunication Union) 的標準。

DVB 制訂 MHP 標準的目的，是希望同時結合廣播、電信、與資訊科技，將 DVB 原本的單向廣播標準，提升成為一種互動數位電視的標準。在 MHP 出現之前，廣播電視業已經出現了許多種各家廠商專屬的 middleware，主要應用在付費電視的環境下 (cable 和 satellite)。制訂 MHP 標準的另一個目的，是希望能提出一個開放式的 middleware 標準，來漸漸取代這些專屬標準。過去採用專屬標準的互動數位電視是所謂的垂直市場，供應鍊的每個環節缺乏競爭，所以會有頭端與客戶端設備昂貴、以及缺乏互動數位內容等問題。但因為 MHP 是開放式標準，它所形成的會是一個互動數位電視的水平市場。在互動數位電視供應鍊的每個環節上，都有因為有競爭而使得數位內容和設備的價格降低。另外，也因為市場規模擴大了，數位內容種類也會變得更多。

MHP 標準的最初版本 – 1.0 版，在 2000 年 1 月被發表。它的修正版 - MHP 1.0.1 版，發表的時間是一年後 – 2001 年的 4 月。MHP 1.0.1 版修正了一千多個，由 MHP 實作與測試單位的所提出的問題。接著，DVB 在 2001 年的 6 月，發表了 MHP 的 1.1 版，這個版本的相較於原本的 MHP 1.0 版，增加了許多新的功能，包括：

- 加入了 Smart Card API
- 定義了 DVB-HTML 作為 MHP 的選擇性功能
- 加入了 “internet access profile”
- 加入了其他的應用程式功能，包括：從 return channel 下載應用程式、支援將應用程式儲存在 set-top box 或 IDTV 的能力、支援 ‘plug-in’ interface、以及提供了更完善的國際化功能...等。

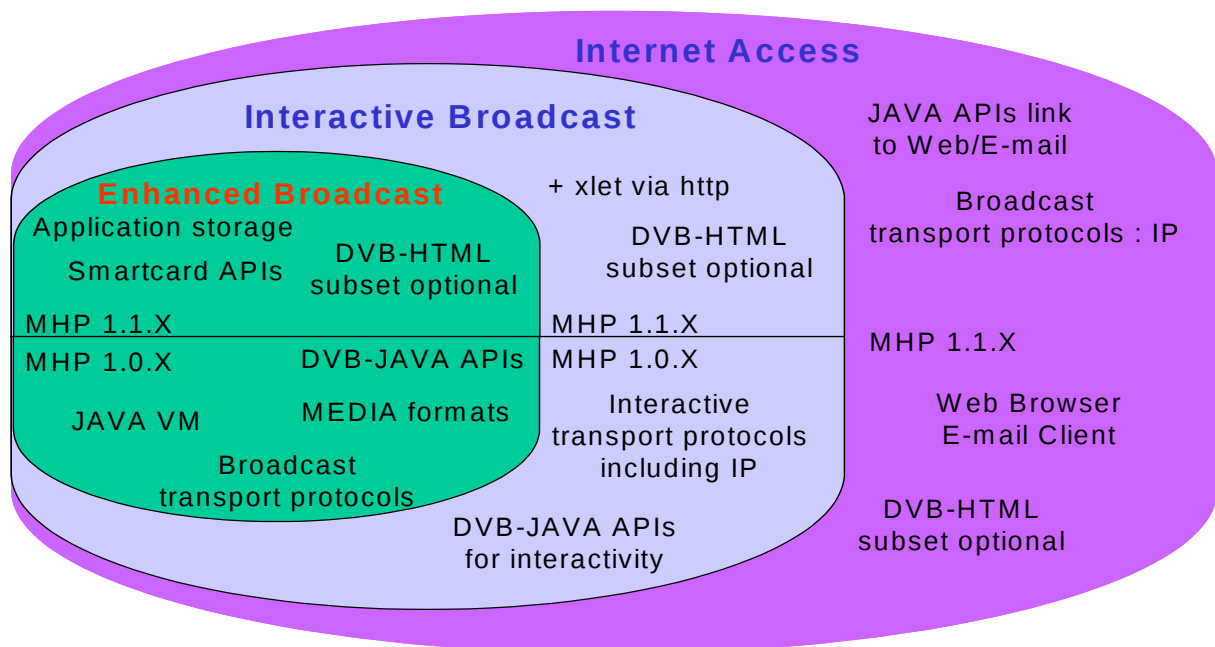


圖 2-2、MHP 所支援的 profile

目前，MHP 最新的版本是 MHP 1.0.3 (2003 年 3 月) 與 MHP 1.1.1 (2003 年 4 月)。請參考圖 2-2，MHP 定義了三種不同的 profile，包括 enhanced broadcast、interactive broadcast、以及 Internet

access。MHP 1.0 標準只支援前兩種 profile，MHP 1.1 標準則支援了所有的 profile。Enhanced broadcast 適用於 set-top box 沒有 return channel 的情況；interactive broadcast 則適用於有 return channel path 的情況。MHP 在 return channel 上支援的是 TCP/IP family 的 protocol，包括 TCP 與 UDP。至於 Internet access profile，主要的目的是定義 MHP 的應用程式，如何跟 Internet client 彼此雙向互動的能力。

圖 2-3 是 MHP 的系統架構圖。MHP 是一種 Java-based 的 middleware，所以需要 Java virtual machine。由於 MHP 標準制訂的原則是，盡量採用現有最佳的標準當基礎，所以在 MHP 裡，採用了 Sun 的 personal Java core libraries、Java TV API，以及 HAVi API、與 DAVIC API。至於 MHP 內部新增的功能，則被歸類為 DVB specific API。Java TV API 主要負責控制 video 與 audio stream 的 playback；HAVi API 負責的是使用者界面的功能；而 DAVIC API 負責的主要是控制 MPEG 處理的一些相關參數。Application manager 是 MHP 的控制中心，它控制了 MHP 上執行的應用程式的 life cycle。此外，它也負責管理 MHP 上執行的應用程式的資源使用以及顯示方式。至於圖中的 Plug-in，主要的目的是讓舊有的互動數位電視的應用程式 (如 OpenTV 的應用程式)，也可以在 MHP 的 set-top box 上被執行。

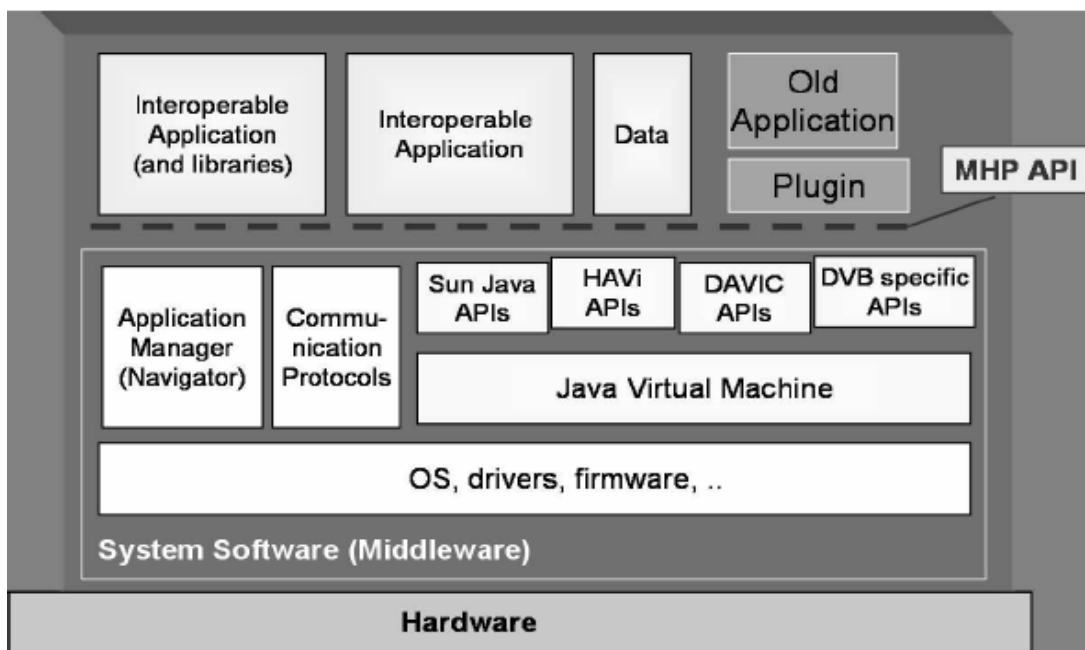


圖 2-3、MHP 的系統架構圖

為了確保各家廠商所推出的 MHP middleware 的相容性，如圖 2-4 所示，DVB 訂定了一套 MHP conformance test 的程序。DVB 本身並不開發 conformance test 的程式，只成立了一個 MHP Expert Group (MEG)，負責收集與管理由其它的公司或組織所 submit 上來的 test 程式。MEG 會將整理完成的測試程式組 (test suite) 提交給 ETSI，由 ETSI 負責轉交給需要 test suite 的廠商。目前最新版的 MHP test suite 是 1.0.2 b，下一版 1.0.3 a 預計會在 2004 年 10 月推出。MHP 的相容性測試是一種所謂的“self-certification”，由廠商自行負責測試的工作。有需要的廠商，可以付 1000 歐元的管理費向 ETSI 拿到 MHP 的 test suite。在完成相容性測試之後，廠商如果

想在產品上打上 MHP logo，則還需要再付 DVB Project Office 10000 歐元的費用 (這是第一筆初始的費用，之後每年還要付年費)，當作 ETSI 為廠商維護 MHP logo 的費用。

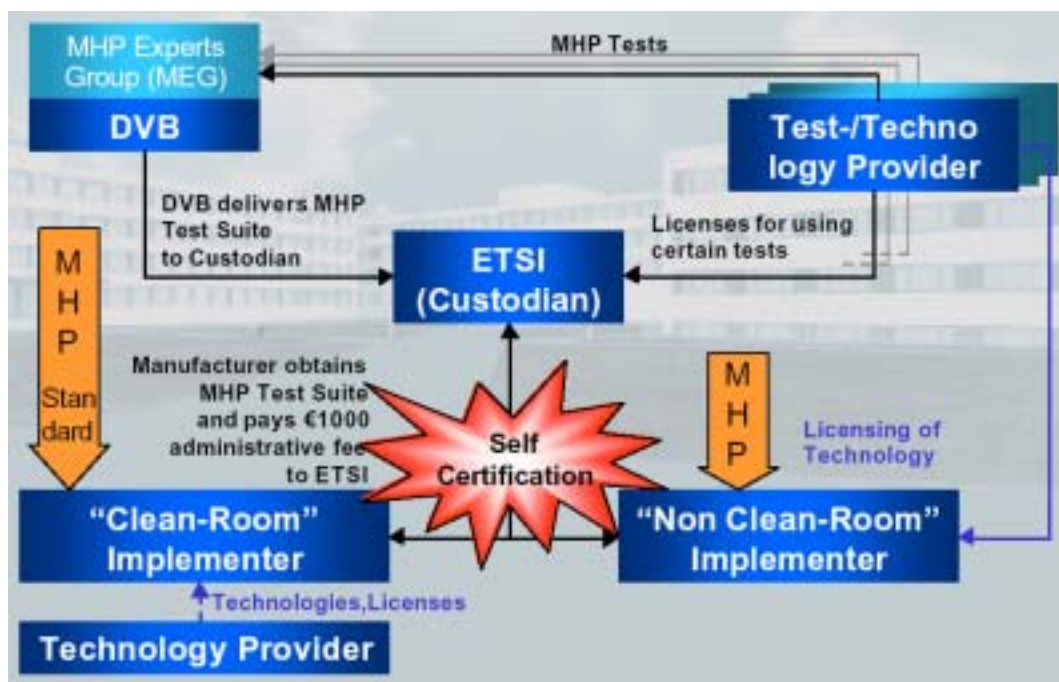


圖 2-4、如何拿到 MHP 的 logo

表 2-1 列出了目前全球主要的 MHP solution provider，相關的產品包括 MHP 的頭端設備、MHP 的 authoring tool、MHP middleware、與 MHP 的應用程式。其中，IRT 的 MHP middleware 是最初在 PC 上開發，用來驗證 MHP 標準可行性的 reference implementation。Osmosys 的 MHP middleware，則是第一個宣稱通過 MHP 相容性測試的 middleware；Panasonic 的 MHP set-top box 是第一個宣稱通過 MHP 相容性測試的產品。目前，工研院電通所 V100，在 MHP middleware、MHP authoring tool，以及 MHP conformance test 相關的核心技術上都有佈局，希望將來能在國內大力推廣 MHP 的相關核心技術。

MHP 未來將往以下幾個重心發展：

- 如何更進一步的整合 IP world 與 DVB world
- 支援 Personal Video Recoder (PVR)
- 如何和家庭網路 (home network) 整合
- 如何和寬頻 IP 網路上的服務整合
- 支援 MPEG-2 之外的 audio/video streaming format
- 提供 MHP mobile 應用程式另一個新的 profile

表 2-1、目前全球主要的 MHP Solution Provider

Company	MHP Headend	Authoring Tool	MHP Middleware	MHP Application
Alticast (Korea)	Yes	Yes	Yes	Yes
Canal+ (France)	Yes	Yes	Yes (with IRT)	Yes
Panasonic (Japan)		Yes	Yes	
Philips (Netherlands)			Yes	
Sofia (Finland)		Yes		Yes
IRT (Germany)			Yes	
Osmosys (Swiss)	Yes	Yes (with Aircode)	Yes	Yes (with Sofia)
SONY (UK)		Yes		Yes
Thales (USA)	Yes	Yes	Yes	Yes

2.2 GEM (Globally Executable MHP)

GEM 規格制定的目的，是為了讓其他的組織，如 CableLabs、ATSC、與 ARIB，在以 MHP 為基礎制定規格時，能達到以下的三個目的：

- 由不同組織所制定的，以 GEM 為基礎的規格，彼此間的 interoperability 能夠盡量最大化。
- 因為在 GEM 中盡量使用了原本 MHP 的元件，可以使得整個互動數位電視的產業鏈，上面的每個環節變得更具有經濟規模。
- 同時能兼顧區域性商業與技術上的需求與限制。

GEM 本身並不是一個完整獨立的標準，也不能直接拿來實作。它只是其他的組織，在制定以 MHP 為基礎的標準時，所參考的一個框架 (framework) (GEM-based 的規格通常被稱作是 GEM terminal specification)。另外，GEM 的規格在設計時，最重要的考量就是想辦法讓應用程式開發者，很容易的可以開發出，能在不同規格的 GEM-based 的數位電視接收器上，都可以被執行的應用程式。GEM 將 MHP 的標準內容分為兩大類，第一類是所謂的 DVB technology 的部份，第二類則是 market-specific 的部份。第一類的 MHP 標準內容，是所有以 GEM 為基礎的標準所共通的。GEM 定義了一組共通的 API、API 實際的語意、以及共同支援的 content 的格式，供想撰寫可以在全球 GEM-based 的 middleware 平台上執行的開發者使用。第二類的 MHP 標準內容，則是可以隨著 GEM-based 標準各自的獨特需求替換的。這種替換可以成立的前提是，只要被替換的 MHP 標準內容，和替換它的技術在功能上相同 (functional equivalents) 就

可以。至於被替換的標準和替換它的技術，是不是在功能上真的是相同的，就需要由訂定 GEM-based 標準的組織與 DVB 彼此協調確認了。表 2-2 是 GEM 1.0.1 規格裡的 “Table 8: Functional equivalents”，列出了在 MHP 原本的規格中，屬於 market-specific 的部份。這些 functional equivalents 還可以再被分為兩大類，第一類是強制性的 (mandatory)，所有的 GEM-based 標準都必須支援。第二類則是選擇性的，GEM-based 標準可以依據本身的需求，來決定要不要支援這些 functional equivalents。

Name	GEM Clause(s)	MHP Definition	Enhanced Broadcast Profile 1	Interactive Broadcast Profile 1
Arch	5, "Basic Architecture"	MHP [1], clause 5, "Basic Architecture"	<u>Q</u>	<u>Q</u>
Carousel	6.2.5, "Object carousel" See also 11.7.2, "Application discovery and launching APIs"	MHP [1], clause 6.2.5, "DSMCC User-to-user Object Carousel"	<u>M</u>	<u>M</u>
IP MPE	6.2.6, "Protocol for delivery of IP multicast over the broadcast channel"	MHP [1], clause 6.2.6, "DVB Multiprotocol Encapsulation"	<u>Q</u>	<u>Q</u>
SI	6.2.9, "Service information"	MHP [1], clause 6.2.9, "DVB Service Information"	<u>M</u>	<u>M</u>
	11.6.1, "Signaling-bound service information API"	MHP [1], clause 11.6.1, "DVB Service Information API"		
	Annex O, "Integration of the JavaTV SI API"	MHP [1], clause O, "Integration of the JavaTV SI API"		
Broadcast IP signalling	6.2.10, "IP signalling"	MHP [1], clause 6.2.10, "IP signalling"	<u>Q</u>	<u>Q</u>
Audio	7.2.1, "Audio"	MHP [1], clause 7.2.1, "Audio"	<u>M</u>	<u>M</u>
Video	7.2.2, "Video"	MHP [1], clause 7.2.2, "Video"	<u>M</u>	<u>M</u>
Subtitles	7.2.3, "Subtitles"	MHP [1], clause 7.2.3, "Subtitles"	<u>Q</u>	<u>Q</u>
	11.4, "Presentation APIs, classes related to subtitles"	MHP [1], clause 11.4.2.5.1, classes related to subtitles		

表 2-2、GEM 所定義的 functional equivalents

以下我們列舉了一些 GEM 與 MHP 規格上，重要的相異處：

- GEM-based 的規格因為使用的是和 DVB 不同的 SI (Service Information) 格式，所以，不需要支援 DVB-SI API 和 MHP 原本的 protocol-independent SI API。但 GEM-based 規格必須提供自己的 protocol-dependent 與 protocol-independent SI API。
- GEM-based 規格可以不使用 MHP 的 object carousel，但必須支援功能上類似的機制。
- 對 application signaling 這項功能而言，GEM 並沒有定義 application signaling 的 API，與該如何做 signaling 的細節。GEM 只定義了在 API 層面，以及該怎麼支援這些 API 的底層功能上的需求。
- GEM-based 規格可以某種程度上地延伸 MHP 原本的 security framework。GEM-based 規格可以更改怎麼判斷一個程式是不是 “trusted application” 的規則；也可以更改如何指定 permission request file 的方法。另外，因為 GEM 只定義了如何和 CA (certificate authority) server 互動的需求，GEM-based 規格可以根據這些需求，來定義實際的 CA message 和 CA server 的配置方式。

- 因為 GEM 並沒有定義 locator 的格式，所以，在撰寫 portable 的 GEM 應用程式時，要注意別將處理 locator 的程式碼寫死。
- GEM-based 規格不一定需要支援 CA API。
- GEM-based 規格不一定需要支援 IP over the broadcast channel (DVB Multiprotocol Encapsulation)。
- GEM-based 規格不一定需要支援 DVB subtitle。

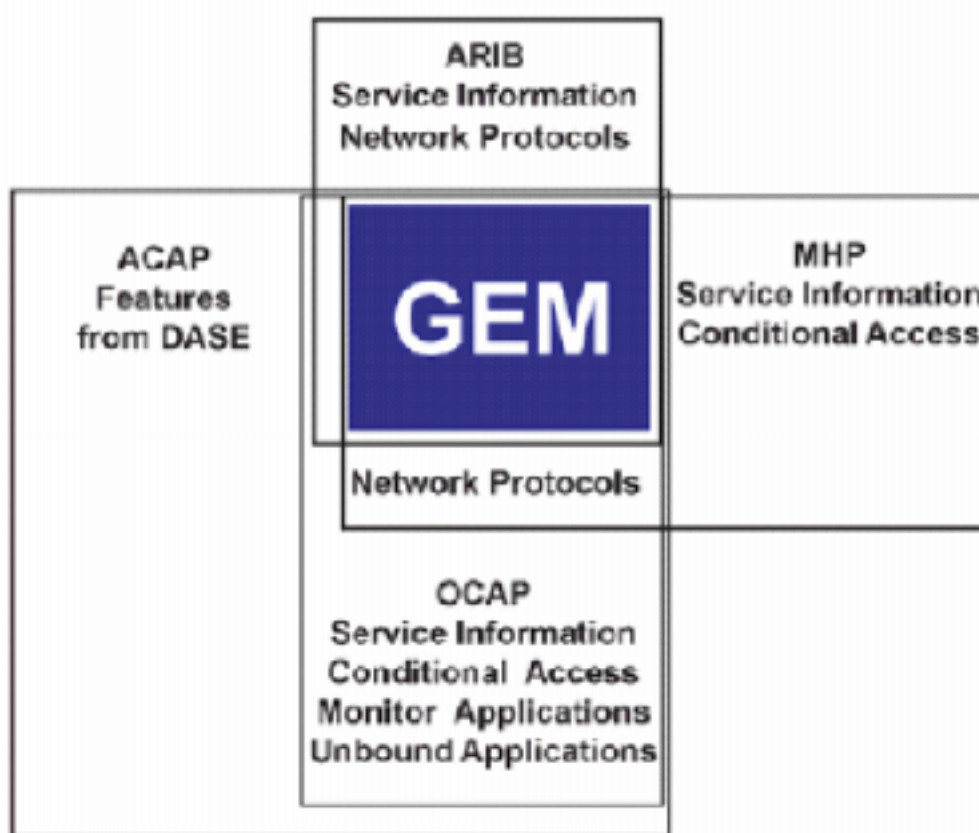


圖 2-5、GEM 與 MHP、OCAP、ARIB-AE 與 ACAP 之間的關係圖

除了前述的規格說明之外，GEM 的規格書裡，還詳細地列出了使用它的其他規格。以下，我們大致簡單地描述一下，GEM 與 OCAP ACAP 與 ARIB-AE 之間的差異 (請參考圖 2-5)。OCAP (Open Cable Application Platform) 是由北美的 cable 業者的規格制定組織 – Cable Labs 所制定的，它是 MHP 為了適應美國 cable 產業的技術與市場環境，所衍生出來的規格。就技術面來說，一些沒有在美國 cable 環境下使用的 DVB 相關技術與規格，在 OCAP 裡，已經都被更換成其他功能相同的元件。另外，就經營層面 (business level) 來說，OCAP 對 MHP 做了一些額外的擴充。OCAP 增加了所謂的 “monitor application”，它可以讓 cable 的網路營運商，控制與

改變 OCAP 接收器運作時的一些 policy。另外，OCAP 還加入了所謂的 unbound application，在 OCAP 接收器上，它可以在任何時間被執行；和 OCAP 接收器目前正在收看哪一個電視頻道完全無關。例如，email client 與 video-on-demand client 都屬於 unbound application 的一種。

ATSC 的 ACAP (Advanced Common Application Platform) 也同樣是以 MHP 為基礎的標準。ACAP 和 MHP 主要的差異在，它重新定義了在 GEM 裡所指定的一些 function equivalent，這些差異包括了：一、ACAP 使用的 Object Carousel，是原本 MHP 的 DSMCC Object Carousel，稍經修改後的版本。二、強制 ACAP 的接收器，一定要有 return channel。三、支援 OCAP 規格裡所定義的 unbound application。除此之外，ACAP 還增加了一個實作規格時的選項，稱為 ACAP-X。ACAP-X 是 DASE 宣告性應用程式環境 (declarative application environment) 的一個修改過的子集合，採用的技術是 HTML 和 ECMAScript。將來，ACAP-X 可能會朝著和 DVB-HTML 與 ARIB 的 BML 技術上融合的方向前進。因為 ACAP 在制定時，參用了許多已經成熟的標準，例如 GEM、MHP 與 OCAP。對想實作 ACAP 的公司或組織來說，也必須留心 GEM 和 OCAP 所使用到的 patent pool。

ARIB STD-B23 – Application Engine Platform for Digital Broadcasting (ARIB-AE) 是由日本 ARIB 所制定的標準，目的是將 GEM 加入目前日本的數位電視廣播標準中。和 MHP 不同的是，在 ARIB-AE 裡，所使用的並不是 DVB 的 object carousel，而是 ARIB 原有的 data broadcasting 規格。

在 DVB 組織內，負責 GEM 標準制定的是 MHP Umbrella Group (MUG)。GEM 的最初版本 - GEM 1.0.0 版，是在 2002 年 9 月發表的。GEM 是以 MHP 1.0.2 版為基礎制定的，並考慮到 OCAP 1.0 與 ARIB-AE 兩個標準的需求。GEM 的最新版 - GEM 1.0.1，是在 2004 年 5 月發表的。這個標準在制定時是以 MHP 1.0.3 版為基礎，另外，再參考 OCAP、ARIB-AE 與 ATSC ACAP 的需求。原本 MHP 的標準所提供的功能，可以再被分成兩大類：第一類是程序性的功能 (procedural functionalities)，也就是 DVB-J 的部份；第二類則是宣告性的功能 (declarative functionalities)，也就是 DVB-HTML。因為目前的 GEM 是基於 MHP 1.0.X 所制定的，所以，目前 GEM 只定義了 MHP 程序性的功能與 GEM-based 標準的程序性功能，之間的互通能力。

2.3 ITU-T J.202

在 ITU 裡，ITU-R Study Group 6 (SG6, broadcasting service) 之下的 Working Party 6M (WP6M, interactive and multimedia service) 是負責研究 multimedia broadcasting 的單位。目前 WP6M 主要的工作，是協調全球的互動數位電視的內容格式與應用程式環境。此外，WP6M 和 ITU-T Study Group 9 (SG9, integrated broadband cable network and television) 還合作組織了 Joint Rapporteur Group 1 (JRG1)。JRG1 主要的工作是研究程序式應用程式的執行引擎 (execution engine, EE)，與宣告式應用程式的呈現引擎 (presentation engine, PE)。

在 JRG 1 2002 年 3 月的會議裡，最後提出了兩份報告：

- 關於 EE 的報告 – “Progress report of harmonization of procedural functionalities for interactive TV applications”
- 關於 PE 的報告 – “Preliminary considerations on harmonization of declarative functionalities for interactive TV applications”

這兩份報告指出了 ARIB、ATSC-DASE、DVB-MHP、OCAP、與 DDE-1 這些標準之間，重要的核心包括哪些，也指出了這些標準的差異所在。

在 JRG-1 於 2002 年 9 月的會議中，制定了 ITU-T J.202 標準的草案 - “Working document towards preliminary draft new recommendation - Harmonization of procedural content formats for interactive TV applications”。在 2003 年 4 月，ITU-T SG9 的會議中，通過了 ITU-T recommendation J.202 的草案，變成 ITU 的正式標準。ITU-T recommendation J.202 正式的全名是: “Harmonization of procedural content formats for interactive TV applications”。J.202 將 GEM 指定為所有數位互動電視 EE 的規格裡，所必備的共通核心。至於其他的數位互動電視 middleware 規格，如 ARIB、DASE、OCAP、與 MHP，裡面的 EE 相關部份的規格，則被視為是選擇性與資訊性的 (informative) EE 規格。

在 ITU 裡，還有其他關於數位多媒體廣播的研究。例如，在 2002 年 3 月 WP6M 的會議裡，就完成了關於 DRM 相關的標準草案: “Preliminary draft new question ITU-R rights management in a digital broadcast environment”; 以及關於行動接收方面的標準草案: “Preliminary draft new report on broadcasting of multimedia and data applications for mobile reception”。

簡單地說，ITU-T J.202 將 GEM 的 API 分成兩類。第一類是 GEM API 的共通核心。第二類是支援第一類 API 的輔助性 API。在互動數位電視的 middleware 規格裡，完整地支援這兩類 API，將可以確保數位互動電視程式的二進位碼相容性 (binary interoperability)。ITU-T J.202 還將這兩類的 API 所分屬的 Java package 詳列出來，如表 2-3 與表 2-4 所示。

java.awt
java.awt.event
java.awt.image
java.beans
java.io
java.lang
java.lang.reflect
java.net
java.security
java.security.cert
java.util
java.util.zip
javax.media
javax.media.protocol
javax.tv.graphics
javax.tv.locator
javax.tv.media
javax.tv.media.protocol
javax.tv.net
javax.tv.service
javax.tv.service.guide
javax.tv.service.navigation
javax.tv.service.selection

javax.tv.service.transport
javax.tv.util
javax.tv.xlet
org.davic.media
org.davic.resources
org.havi.ui
org.havi.ui.event

表 2-3、ARIB STD-B23、DASE-1、MHP 1.0.2、
MHP 1.1、與 OCAP 1.0 的共通核心 API (J.202 Annex A)

java.math
java.rmi
java.security.spec
javax.net
javax.net.ssl
javax.security.cert
org.davic.mpeg
org.davic.mpeg.sections
org.davic.net
org.davic.net.dvb
org.davic.net.tuning
org.dvb.application
org.dvb.dsmcc
org.dvb.event
org.dvb.io.ixc
org.dvb.io.persistent
org.dvb.lang
org.dvb.media
org.dvb.net
org.dvb.net.tuning
org.dvb.net.rc
org.dvb.test
org.dvb.ui
org.dvb.user

表 2-4、ARIB B24STD-B23、MHP 1.0.2、MHP 1.1、
與 OCAP 1.0 標準之間，其他共通的 API (J.202 Annex B)

除了各標準之間共通的 API 外，ITU-T J.202 還詳細列出了，其他每個標準各自專屬的 Java API package，如表 2-5 至表 2-9 所示。

org.davic.mpeg.dvb
org.davic.net.ca
org.dvb.net.ca
org.dvb.si

表 2-5、MHP 1.0.2 and MHP 1.1 彼此共通的額外專屬的 API (J.202 Annex C)

org.ocap.application
org.ocap.event
org.ocap.hardware
org.ocap.hardware.pod
org.ocap.media
org.ocap.net
org.ocap.resource
org.ocap.service
org.ocap.system
org.ocap.system.error
org.ocap.ui.event

表 2-6、OCAP 1.0 專屬的 API (J.202 Annex D)

jp.or.arib.tv.media
jp.or.arib.tv.net
jp.or.arib.tv.si
jp.or.arib.tv.ui

表 2-7、ARIB STD-B23 專屬的 API (J.202 Annex E)

java.applet
java.awt.datatransfer
java.text
org.dvb.application.inner
org.dvb.application.plugins
org.dvb.application.storage
org.dvb.dom.bootstrap
org.dvb.dom.css
org.dvb.dom.dvbhtml
org.dvb.dom.environment
org.dvb.dom.event
org.dvb.dom.inner
org.dvb.internet
org.dvb.smartcard
org.w3c.dom
org.w3c.dom.css
org.w3c.dom.events
org.w3c.dom.html
org.w3c.dom.stylesheets
org.w3c.dom.views

表 2-8、MHP 1.1 專屬的 API (J.202 Annex F)

com.sun.awt
com.sun.lang
java.text
java.util.jar
javax.tv.carousel
org.atsc.application
org.atsc.carousel
org.atsc.data
org.atsc.dom
org.atsc.dom.environment
org.atsc.dom.html
org.atsc.dom.views
org.atsc.graphics
org.atsc.management
org.atsc.net
org.atsc.preferences
org.atsc.registry
org.atsc.security
org.atsc.si
org.atsc.system
org.atsc.trigger
org.atsc.user
org.atsc.xlet
org.w3c.dom
org.w3c.dom.css
org.w3c.dom.events
org.w3c.dom.html
org.w3c.dom.stylesheets
org.w3c.dom.views

表 2-9、DASE-1 專屬的 API (J.202 Annex G)

2.4 MHP 的全球採用狀況

圖 2-6 是來自 DVB 公布的 MHP 的全球採用狀況，表 2-10 則是使用歐規數位電視標準的國家，對 MHP 的採用狀況。目前，採用 MHP 標準的國家還是以北歐、西歐、與亞太地區的國家為主。使用 GEM-based 標準的國家，則包括北美 (OCAP 與 ACAP)、日本 (ARIB-AE) 與南韓。下面，我們將會摘要說明，MHP 在這些國家的使用與發展現狀。

2.4.1 芬蘭

芬蘭是全球最早開播 MHP 服務的國家，開播的時間是 2001 年的 8 月。至 2004 年 6 月，芬蘭全國大約有 360000 個 DVB-T 數位電視接收機，65000 個 DVB-C 數位電視接收機，以及 10000

個 DVB-MHP set-top box。目前，市場上每個月大概可以售出 50000 台數位電視接收機。

芬蘭的無線數位電視 (DVB-T) 網路，目前由網路營運商 Digita 負責營運。Digita 旗下共有 3 個全國性的 mux，目前總共承載了 14 個數位電視頻道，其中 3 個 Canal+ 所營運的是付費頻道，其它的則是免費收視的頻道。在所有的這些頻道裡，superteletext (節目圖文資訊) 與 EPG service (電子節目表單服務) 是每個頻道都會提供的服務。網路營運商 Digita 也提供了一個 digital service portal (數位服務入口網站) 供所有的用戶使用。除了這些共有的 MHP 服務之外，以下的電視台還提供了一些額外的 MHP 應用程式。

- 公共電視台 YLE:
 - Games、government facts (議會資訊)、news ticker (新聞跑馬燈)
- MTV3:
 - Banking (電視銀行)、real estate finder (房地產資訊)、hockey score ticker (曲棍球比賽跑馬燈)

基本上，這些由芬蘭的無線數位電視電台所播送的 MHP 程式，仍是以與影音內容獨立的 MHP 程式為主。另外，芬蘭的有線數位電視業者，也將在今年年底之前提供 MHP 服務。

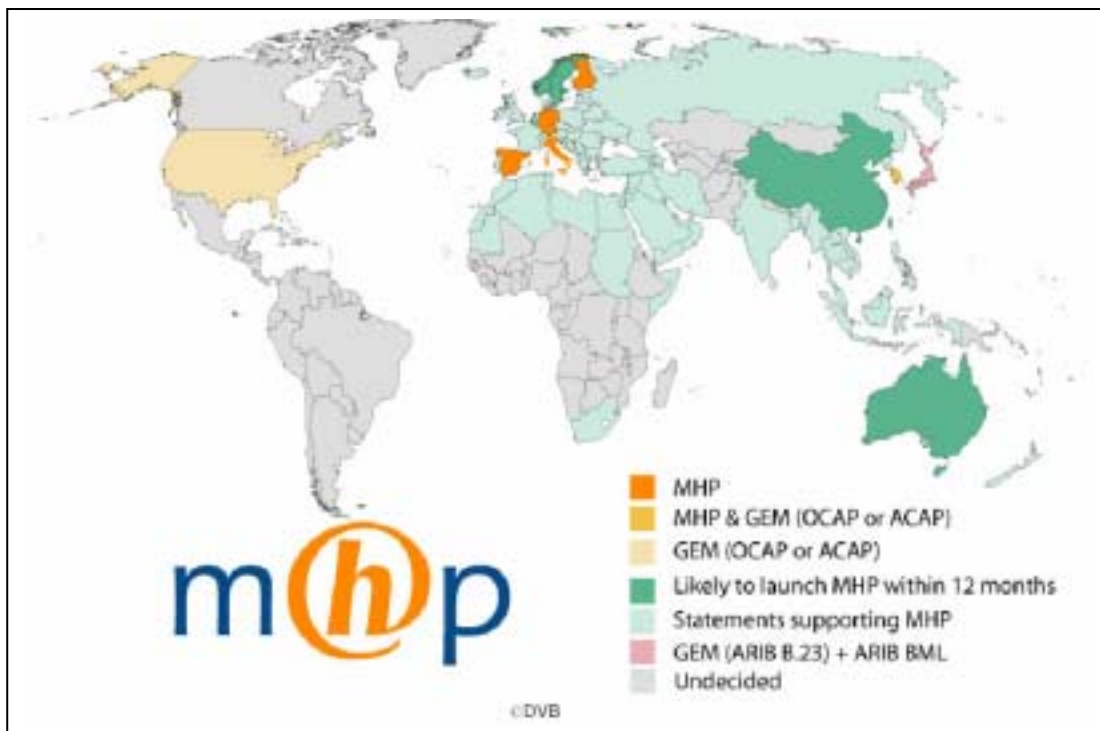


圖 2-6、MHP 的全球採用狀況

2.4.2 德國

德國是全球第二個播送 MHP 服務的國家。德國在 2002 年 10 月開始無線數位電視開始提供 MHP 服務。ARD、ZDF、RTL 等公共電視台停止使用 OpenTV middleware，改用 MHP。另外，德國的公共電視台也提供了衛星廣播數位電視上的 MHP 服務。這些電視台提供的 MHP 應用

程式包括了: TV portal (電視入口網站上)、new ticker、program preview (節目預告)、weather forecasts (氣象報告)、interactive quiz (互動式猜謎)、mail order (郵購)、magazine (電視互動雜誌)...等。和芬蘭一樣，德國也是採用零售數位電視 set-top box 的商業模式;目前在德國零售的 MHP set-top box 的廠牌包括: SONY、Philips、Panasonic 與 Nokia。

Countr	DVB-	DVB-	DVB-
Germany	●		●
Denmark	○		
Norway			●
Finland		●	●
Italy	●		○
France			
Spain			●
Singapore			●
Australia			○
Korea	●		

- : MHP application on air
- : MHP service planned for the near future

表 2-10、使用歐規數位電視標準的國家，對 MHP 的採用狀況

2.4.3 義大利

義大利政府希望能提供民眾方便的 E-government 服務。基於電視的穿透率遠高於其他的電子媒體，義大利政府為了推廣以 MHP 實作的 T-government 服務，決定補助 1 億 2000 萬 ~ 1 億 5000 萬歐元，在 MHP 數位電視接收機價格下降至民眾願意負擔的範圍之前，市場上能有足夠的 MHP 數位電視接收機。義大利的無線數位電視於 2004 年 1 月正式開播，五個月內總共銷售了 45 萬個 MHP 數位電視接收機。由於義大利類比無線電視的市場佔有率極高，因此，義大利是 MHP 數位電視接收機一個非常大的潛在市場。

2.4.4 美國

在 2002 年，美國第二大的 DVB-S 服務商 EchoStar，也跟它的 middleware 供應商 Canal+ Technology 簽訂了合約，Canal+ Technology 將會提供與 MHP 相容的 MediaHighway API 給

EchoStar 下一代的衛星解碼盒使用 (目前 MediaHighway 相關產品已被 NDS 公司併購至其產品線內)。另外,如前所述,北美的 cable 標準制訂組織 CableLabs, 已經在 2002 年接受了 MHP, 作為其 OpenCable 平台上的 middleware – OCAP 的發展基礎。Time Warner Cable 所開發的 OCAP 平台, 據悉目前已經進入測試的階段。

2.4.5 南韓

南韓目前有 1500 萬左右的電視收視人口。如表 2-11 所示, 在無線數位電視的部分, 南韓使用的是美規的 ATSC 標準, 所採用的 middleware 也是美規的 ACAP/DASE。在有線數位電視的部分, 南韓所採用的同樣是北美的 OpenCable 標準, 因此, 使用的 middleware 是 OCAP。至於衛星數位電視的部分, 因為南韓所採用的是歐規的 DVB-S 標準, 所以使用的 middleware 是 MHP。簡而言之, 南韓數位電視所使用的 middleware, 不管是無線數位電視、有線數位電視、或衛星數位電視, 都屬於 MHP 或 MHP 的衍生標準。南韓 DVB-S 的服務商 Skylife, 已經於 2003 年 5 月, 開始提供 MHP 的服務。截至 2004 年 7 月, Skylife 已有 650000 個客戶使用 MHP 數位電視機頂盒。目前, SkyLife 所提供的 MHP 應用程式包括: game、education (電視線上教育)、fortune (電視算命)、stock (股票服務)、weather (氣象服務)、real estate (房地產資訊)、還有 voting (投票)。將來還會有 T-Government 等 MHP 程式會上線。

Korea Broadcast System	DTV Standard	Datacast Standard
Digital Terrestrial	ATSC	ATSC-DASE
Digital Satellite	DVB-S	DVB-MHP
Digital CATV	Open Cable	OCAP

表 2-11、南韓所採用的數位電視傳輸與 middleware 的標準

2.4.6 西班牙

在 2003 年的第四季, 西班牙在加泰隆尼亞 (Catalonia) 地區, 開始了地面波數位電視的測試計畫, 同時也提供了 MHP 的數位互動服務。提供的 MHP 應用程式包括: EPG news ticker, weather forecast、programming and special event (節目與事件規劃表)、chat and text messaging (文字訊息交換與對談服務)、interactive advertising (互動廣告)。

2.4.7 丹麥

丹麥的 Danish Broadcast Company 在 2003 年 12 月, 選擇了 MHP 來提供互動數位電視的服務。Danish Broadcast Company 將提供一個 information portal (資訊入口網站), 內部將會提供新聞、運動、一般節目資訊與電視廣播資訊。此外, 他們還希望能提供各種不同的互動猜謎程式吸引觀眾。

2.4.8 澳洲

2001 年 10 月 31 日，澳洲商業電視台聯盟 (Federation Australian Commercial Television Station, FACTS) 的數位電視策略小組，宣布澳洲的 free-to-air 電視台使用 MHP 以提供互動電視服務。從 2002 年 1 月 30 日開始，Special Broadcasting Service (SBS) 與 Sun Microsystems 合作展開 MHP 的互動數位電視測試。

表 2-12、各種 middleware 所需要的最小系統資源

Solution	Processor	RAM	FLASH/ROM
Basic Zapper	30MHz+	1-2M	1-2M
MHEG-5	50MHz+	4M	2M
Open TV	50MHz+	4-8M	4M
MediaHighway	50MHz+	4-8M	4M
MHP Enhanced Broadcast profile*	80-130MHz+	8-16M	8M
MHP Interactive Broadcast Profile*	80-130MHz+	8-16M	8M
MHP Internet Access Profile*	150-200MHz+	16-32M	16M

2.4.9 東歐地區

保加利亞、捷克與匈牙利等國，今年起也開始有部份的無線及有線數位電視台，開始進行 MHP 的播送測試，但商業播送的時間仍尚未決定。

2.4.10 台灣

目前台灣所有的地面波電視廣播業者，都已經購買了 MHP 的頭端設備，並且已經展開 MHP 服務播送的測試工作。此外，國內也已經有數家 set-top box 廠商與開發數位電視 IC 的公司，正在進行 MHP 相關的開發工作。

2.5 MHP 的成本分析

因為 MHP 的 set-top box 的價格，對使用者會不會想使用 MHP 的服務來說，是一個關鍵的重要因素。所以，在這裡我們簡單地分析了一下 MHP 的 set-top box，與使用其他專屬 middleware 的 set-top box，在價格上的變化趨勢。

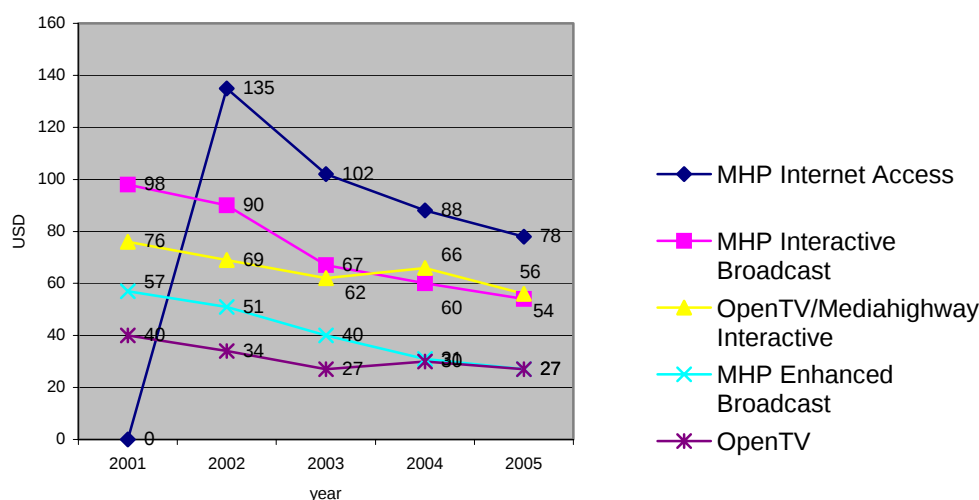


圖 2-7、使用各種不同的 middleware 的 set-top box 的 BOM cost

表 2-12 是 DVB MHP 組織所提供的，關於各種不同的 middleware，在執行時所需的最小系統資源的資訊 (不包括 DVB-HTML)。根據這個表，可以計算圖 2-7 的趨勢圖。圖 2-7 顯示了各種 set-top box 的 BOM cost (原始成本)，隨著時間的變化趨勢圖。圖中 OpenTV 與 MHP Enhanced Broadcast 代表的是沒有 return channel path 的 set-top box。OpenTV/Mediahighway 與 MHP Interactive broadcast，代表的都是具有 return channel path 的 set-top box。在這裡，假設 return channel 使用的是 56K 的 modem。基本上，使得 set-top box 成本下降的主要驅動力是，memory 與 processor 的價格會隨時間下降的趨勢。圖 2-7 中所顯示的 BOM cost，乘以 3 大概就是市場上沒有政府補貼的零售價格。在圖 2-7 中，我們並沒有計入每種 middleware 的授權金。不過，因為 MHP middleware 是開放式標準，價格長期來說會低於像 OpenTV 與 Mediahighway 這種專屬標準。另外，我們也假設因為 memory chip 會變的越來越大，middleware 的 memory footprint 大小的差異，對使用的 memory chip 的成本的影響也越來越小。

根據圖 2-7 的預測，我們可以發現，在 2004 年以後，使用功能類似的 middleware (OpenTV vs. MHP Enhanced Broadcast 或 OpenTV/Mediahighway vs. MHP Interactive broadcast) 的 set-top box，它們的 BOM cost 會變的很相近。這項結果顯示，未來 MHP set-top box 在市場上的接受程度，至少會和目前使用專屬系統 middleware 的 set-top box 一樣好。

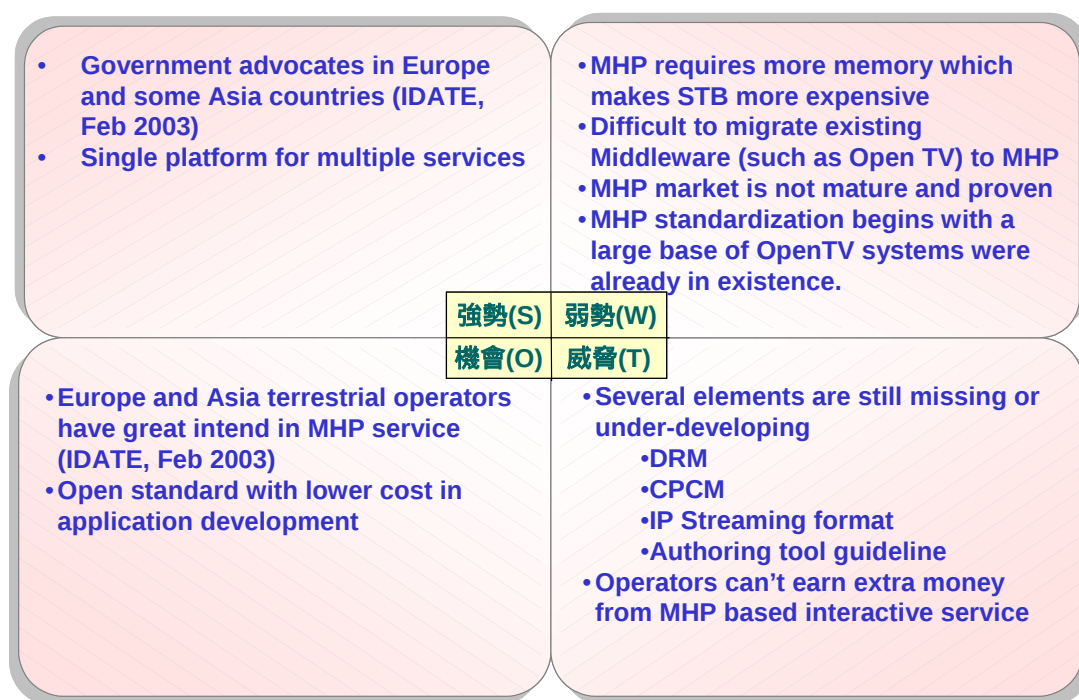


圖 2-8、MHP 的 SWOT 分析圖

Configuration	ROM	RAM	Remarks
MHP 1.0.2 minimal	8 Mb	16 Mb	Incl. resident applications, excl. return channel, CI/CA, FTA SW download
MHP 1.0.2 typical	16 Mb	24 Mb	Full system

表 2-13、MHP 對 RAM 與 Flash 的需求

2.6 MHP 的 SWOT 分析

圖 2-8 是 MHP 的 SWOT 分析圖。我們將分別分析 MHP 的強勢、弱勢、機會與威脅。MHP 的強勢主要有兩項。第一項強勢是某些歐洲與亞洲國家對 MHP 的強力支持。在歐洲，以比利時與西班牙與為例，與 MHP 相關的技術研究計畫，就有政府的強力支持。另外，義大利政府甚至還出資補助家庭購買 MHP 的 set-top box。在亞洲，南韓政府指定 MHP 為衛星數位電視上的 middleware 標準；另外，新加坡政府也宣布 MHP 為互動數位電視的標準 API。而且，GEM 目前已經是歐規、美規地面波與 cable、及日規數位電視標準上，所使用的 middleware 標準的核心。某種程度上來說，GEM 已經可以說是全球互動數位電視的標準 API 了。這對 MHP 的推廣相當有利。MHP 身為一種開放式的標準，是它的另一項強勢，開放式標準可以刺激市場競爭，降低頭端設備、authoring tool、middleware/set-top box/IDTV、以及應用程式的價格。而且因為市場規模擴大了，會刺激各種 MHP 相關應用程式的開發，這對互動數位電視的普及也會很有幫助。

Middleware Market in Asia

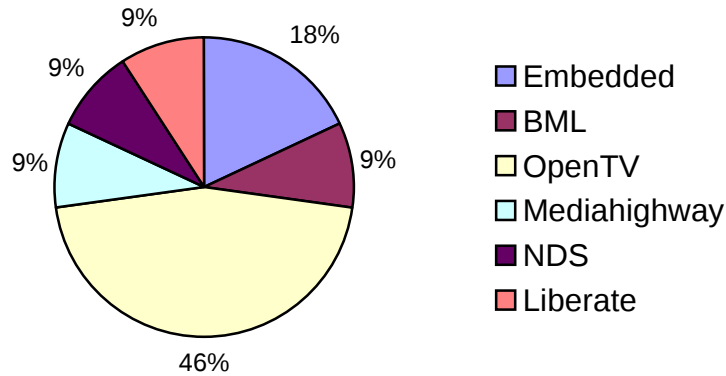


圖 2-9、亞洲的互動數位電視 middleware 市場

至於 MHP 的弱勢，則有以下幾項：第一、MHP 需要比其他 middleware 更多的記憶體，如表 2-13 所示、要執行 MHP 1.0.2 版，至少需要 16 MB 的 RAM、16 MB 的 flash 與 255 MHz 以上的 CPU。這使得 MHP 的 set-top box 的價格會比較高。不過，前面我們已經分析過，這些價格上的差異將會隨著時間而漸漸消失。MHP 的第二個弱勢是，目前還沒有成功營運的 business model 出現。在歐洲國家（例如芬蘭）所採用的 retail business model（由用戶自己買 MHP set-top box 回家使用），如果沒有國家的補助，執行的效果目前看起來並不是很成功。MHP 的第三個弱勢是，在 MHP 標準推出時，已經有很多數位電視的營運商採用了 OpenTV 的標準，MHP 將面臨很強烈的競爭。圖 2-9 是亞洲國家目前所採用的互動數位電視 middleware 種類的分析圖，我們可以發現，目前使用 OpenTV 標準的比例高達 46%。MHP 的第四個弱勢是，因為目前已經有不少專屬的互動數位電視 middleware（例如 Open TV）被採用了，要從這些 middleware 轉換到 MHP，營運商需要同時更換 set-top box 與頭端設備，這對營運商來說，會是一筆很沈重的負擔。

Platform Name	SAT/CAB/DTT	Country	CA	Middleware	Number of Digital Subs
SkyLife	SAT	South Korea	NDS	MHP middleware provided by Alticaster, not deployed yet	320,000 DTV Subs.
UBC	SAT	Thailand	IRDE	Embedded, probably migrating to OpenTv En2	291,902 (Jan/2003)
BS Digital	SAT	Japan	?	BML (Japan's own standard)	6000000
CBSAT	SAT	China	IRDE	OpenTV EN2	50,000

Astro	SAT	Malaysia	MGD	Mediahighway	750,000
IndoVision	SAT	Indonesia	NDS	NDS Core	60,000
Austar	SAT	Australia	IRDE	OpenTV En2	350,000 iTV capable
Foxtel (Satellite)	SAT	Australia	IRDE/NDS	Moving to OpenTV core 1.1	450,000 digital home
Foxtel	CAB	Australia	NDS	OpenTV core 1.1	500,000 not yet digital. Roll out Q1 2004
Optus	CAB	Australia	NAG	Liberate 1.2	Trial of 3,000 Extended
Sky NZ	SAT	New Zealand	NDS	OpenTV EN2	356,000

表 2-14、亞洲國家互動數位電視 middleware 的採用狀況

歐洲最大的研究集團 - IDATE 指出，MHP 的機會主要是在地面波、free-to-air 的營運商身上。表 2-14 詳列了亞洲國家的營運商，在互動數位電視 middleware 的採用狀況。以亞洲國家為例，目前使用專屬標準的 middleware 的數位電視營運商，主要都是 cable 或衛星的營運商。這些營運商已經投資了一種互動數位電視的 middleware，可能比較沒有餘力投資在 MHP 上。但對地面波的營運商來說，則沒有前述的這些問題，他們比較可能率先採用 MHP。

至於 MHP 的威脅方面，主要的問題出自於 MHP 的規格仍然不夠完備。雖然 MHP 的標準已經制定完成了，但 MHP 所提供的 framework 對如何建立一個穩固的 business 環境來說，可能還有不足的地方。MHP 目前缺乏的、或是仍在開發的機制包括：

- DRM
- CPCM
- IP Streaming format
- Authoring tool guideline

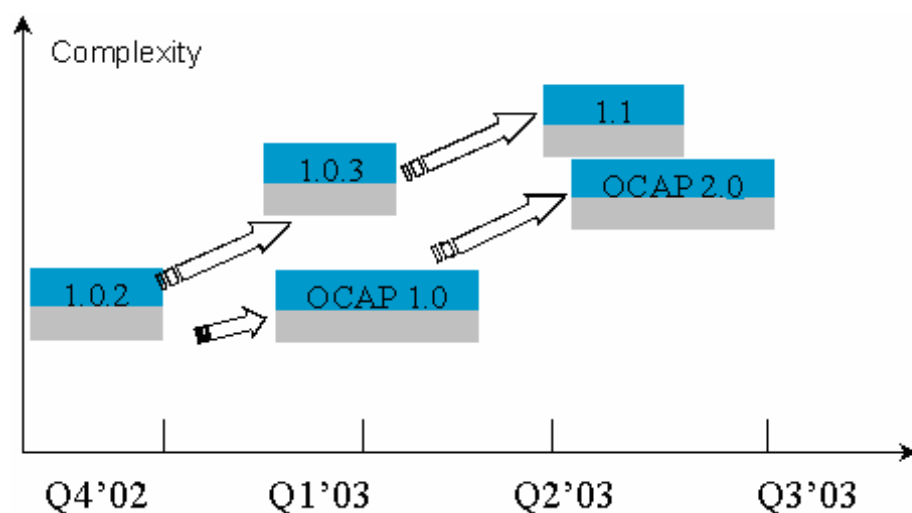


圖 2-10、MHP 和 OCAP 標準的演進關係

2.7 美國 MHP 相關標準介紹

目前美國 (北美) 採用的 MHP 相關標準，都是 GEM-based 的標準，包括 CableLabs 的 OCAP (OpenCable Application Platform，以及 ATSC 的 ACAP (Advanced Common Application Platform)。以下，我們會分別針對這兩個標準，做一番簡單的介紹。

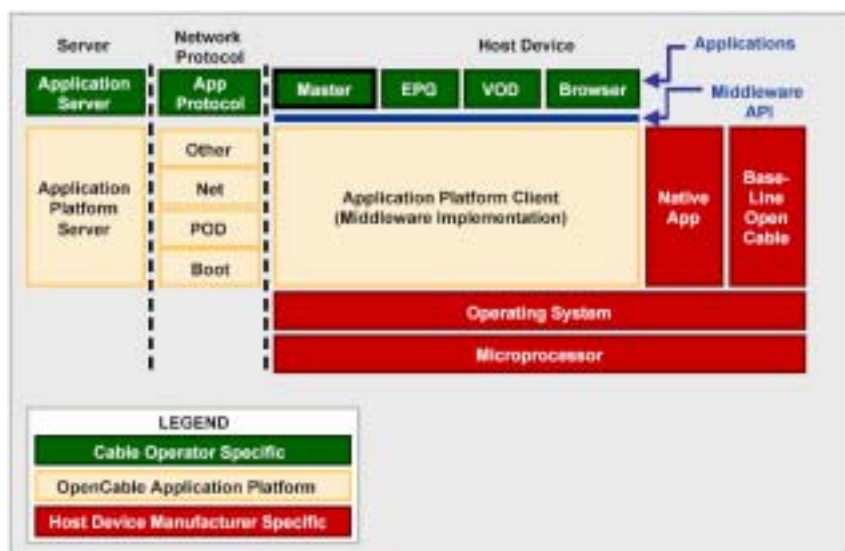


圖 2-11、OpenCable 的軟體架構圖

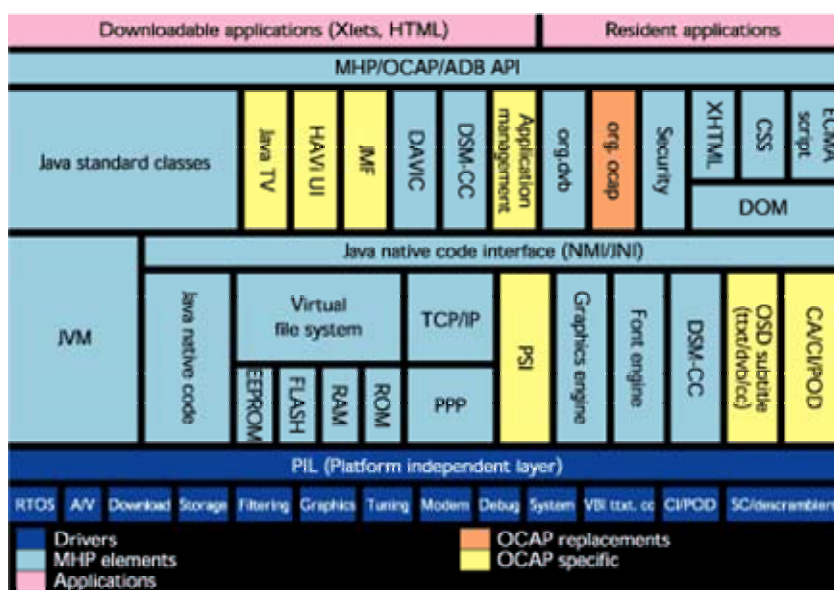


圖 2-12、OCAP 的細部架構圖

2.7.1 OCAP

CableLabs 是一個由 cable 營運商所組成的非營利性技術組織。CableLabs 雖然開放全球各國的 cable 營運商加入，但主要的成員還是集中在北美。在美國，CableLabs 佔有 70% 的 cable 用戶；在加拿大則是 60%；在墨西哥僅有 8%。另外，CableLabs 也有少數的南美洲 cable 營運商加入。CableLabs 的主要任務是追蹤與研究各種與 cable telecommunication 相關的新技術；並幫助 CableLabs 的成員將這些新技術整合進它們的營運目標內。為了達到上述的目的，CableLabs 會制訂與新技術相關的開放式標準。

OpenCable 計劃是 CableLabs 的 Advanced Platform and Service Group，旗下的一個計劃。這個計劃的目的是訂定下一代的 set-top box 的標準，以建立一個提供雙向 cable 服務的共通平台。為了刺激 set-top box 供應商之間在價格上與技術上的競爭，OpenCable 訂定了一組開放式的標準。這些標準可以再被細分為硬體和軟體兩組（請參考圖 2-11），軟體部分的標準就是 OCAP (OpenCable Application Platform)。OCAP 制訂了一組在 host device 上的標準應用程式介面 (API)，讓 host device 可以接受由 cable 營運商所提供的雙向互動服務與應用程式，這些新的服務包括了：

- Electronic program guide (EPGs)
- Video-on-demand (VOD)
- Interactive personal video recorders
- Interactive sports and game shows
- Impulse pay-per-view (IPPV)
- Web 存取，以及 email、chat 和 instant messaging
- Interactive games
- 商業服務，例如購物 (T-Commerce) 與家庭銀行業務 (home banking)

圖 2-10 是 MHP 和 OCAP 兩個標準之間的演進關係圖。OCAP 1.0 的標準是在一年前發表的，因為它是 GEM 1.0 的衍生標準，所以，它和 MHP 1.0.2 標準比較接近。OCAP 2.0 加入了對 web 相關技術的支援，如 XHTML、XML 與 ECMAScript。因此，它也比較類似 MHP 1.1。

圖 2-12 是 OCAP 標準的細部架構圖。OCAP 2.0 規格內的元件，可以再被細分為 presentation engine (PE) 和 execution engine (EE)，以及讓 PE 與 EE 上的內容可以彼此互動的 PE-EE bridge。PE 主要定義了一些和 web 技術相關的工具，例如 HTML、CSS、ECMAScript、DOM、XML、flash 動畫...等，還可以加入 plug-in，以支援一些新定義的 web content format。此外，PE 也加入了一些，上述技術應用在電視時所需要的擴充，如 ATVEF，可以在電視上提供額外的圖形合成 (graphic composing) 的能力。而且，ATVEF 的內容還可以和 video 同步。

至於 OCAP EE 的部分，因為 OCAP 是 GEM 的衍生標準，所以它的主要模組，如 PersonalJava core libraries、JavaTV API、HAVi API、DAVIC API...等，劃分都和 MHP 很相似。不過，內部還是有些細微的不同。圖 2-12 中的黃色和橙色部分，是 OCAP 規格裡與 MHP 不同的模組。黃色部分的模組，代表的是在 OCAP 中，實作了 GEM 規格中所定義的 functional equivalent 的模組。另外，橙色的模組代表的是 OCAP 為了滿足自己獨特的需求，所額外訂定的模組；例如，關於雙向 cable 應用程式的 lifecycle 的管理，與 OCAP 的 PE-EE bridge。PE 和 EE 之間的 bridge，是透過 DOM (document object model) 介面來達成的。EE 上的 Java 程式，可以透過 DOM 去存取 HTML；而 PE HTML 內的 ECMAScript，可以透過 DOM 達到 dynamic HTML 的效果。

因此，藉由 DOM API 和 DOM event，EE 上的 Java 程式和 PE 上的 ECMAScript，可以很方便的彼此通訊。

在制訂 OCAP 標準的過程中，Sun Microsystems 是 EE 最主要的 contributor。由於 Sun 已經將 Java TV API 規格和相關的 IP 授權給 CableLabs，任何想實作 OCAP API 的公司，就不需要再付授權金給 Sun 了。CableLabs 也會將 Sun 所提供的測試程式，加入 OpenCable 的 conformance test suite 中。但要注意的是，只有曾經對 Sun 授權 Java 相關技術的公司，才能在產品上打上使用 Java 技術的標籤。至於 OCAP 標準裡 PE 的部分，最大的 contributor 則是 Liberate 和 Microsoft。另外，Canal+、OpenTV、以及 PowerTV 等公司，也是 OCAP 規格訂定過程中，幾個關鍵的 contributor。

Advanced Digital Broadcast Ltd	IBM Corp	Panasonic
Alticast Corp	NEC Corp	Royal Philips Electronics
Canal Plus Technologies	Matsushita Electric Corp	Thomson Multimedia
Samsung	Sony Corp	

表 2-15、提供符合 OCAP 標準的硬體的公司

Alticast,	Tality	Tektronix
S&T Bulgaria Ltd.,	RTL Newmedia	DigiSoft.tv Ltd
Espial Group Inc		

表 2-16、提供符合 OCAP 標準的軟體的公司

表 2-15 與表 2-16 分別列出了，目前支援 OCAP 規格的軟體與硬體的公司。這些支援 OCAP 規格的軟體，除了互動數位電視的應用程式與內容之外，還包括了與應用程式溝通的應用程式伺服器、object carousel、以及其他將互動內容與 application signaling 放入 MPEG transport stream 裡的裝置。

最後，在今年 5 月，Sun 的 JCP (Java Community Process) PMO (Program Management Office) 接受了一項關於 OCAP 的 JSR (Java Specification Request) – JSR 242: Digital Set Top Box Profile – “On Ramp to OCAP”。希望在 J2ME CLDC 上，新增訂一個以 OCAP 標準的子集合為核心的 profile。目的是希望讓目前已經在使用中的，許多系統資源不足以執行 OCAP middleware 的 set-top box (如 Motorola DCT-2000)，也能夠執行某些 OCAP 的應用程式。這個 profile 所能使用的系統資源限制大致如下：

- 整個系統大概總共只有 700K 左右的 ROM
- 其中的 250K 會被 J2ME CLDC 佔用
- 另外的 450K 則是由 driver、java.awt、java.net subset、javaTV subset、OnRamp-specific API...等使用。

這個標準在制訂時，預計的 schedule 如下：

- Expert Group Formation and Initial Meeting: 2004 年 5 月
- Early Draft Review: 2004 年 7 月
- Public Review: 2004 年 10 月
- Proposed Final Draft: 2004 年 12 月
- Final Release: 2005 年 2 月

2.7.2 ACAP

ACAP (Advanced Common Application Platform) 是由 ATSC (Advanced Television Committee) 組織所制訂的互動數位電視 middleware 標準。ATSC 是在 1982 年，由以下的組織所合作組成的: Electronic Industries Association (EIA)、Institute of Electrical and Electronic Engineers (IEEE)、National Association of Broadcasters (NAB)、National Cable Television Association (NCTA) 與 Society of Motion Picture and Television Engineers (SMPTE)。目前，ATSC 大約有 140 個成員。

在制訂 ACAP 之前，ATSC 已經制訂過一套，執行於數位地面波廣播電視上的 middleware – DASE (DTV Architecture for Software Environment)。DASE 標準包括了 PE (Presentation Engine，採用的技術是 XHTML、CSS、DOM、還有 ECMAScript) 和以 Java 為基礎的 EE (Execution Engine) 兩個部分。DASE 標準雖然採用的技術跟 MHP 同源，但彼此之間仍有很多差異。

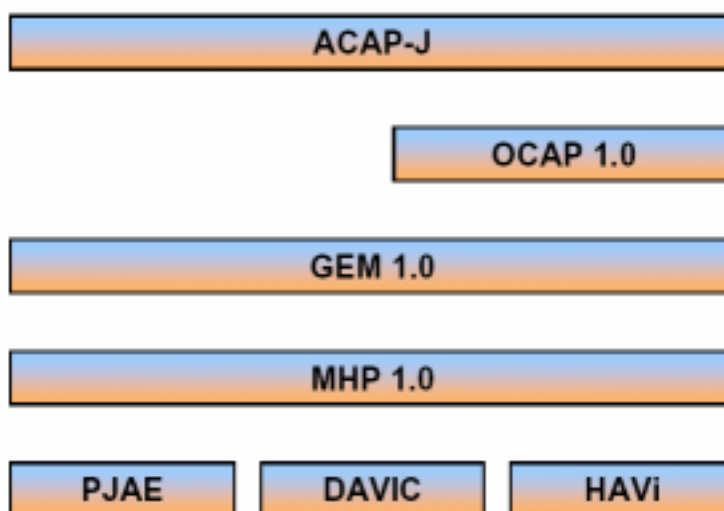


圖 2-13、ACAP-J 所採用的技術來源

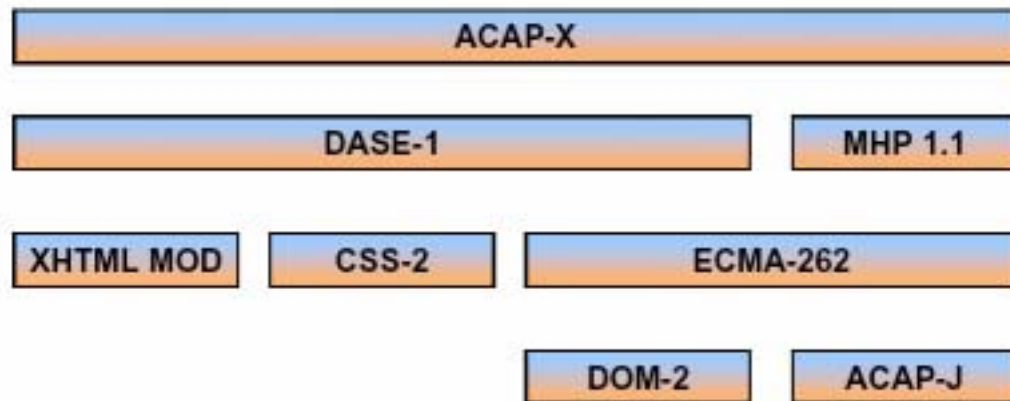


圖 2-14、ACAP-X 所採用的技術來源

為了解決美國的地面波 middleware 標準 - DASE 與北美的 cable 系統上的 middleware 標準 - OCAP，彼此之間不相容的問題，ATSC 的 T3/S2 Specialist Group 制訂了 ACAP。ACAP 是以 GEM 和 DASE 為基礎發展的。另外，為了支援北美的 cable 系統，ACAP 也加入了 OCAP 標準內的某些額外的功能。ATSC 訂定 ACAP 標準的目的是，制定一套 framework 規格，作為未來 DASE 與 OCAP 的共通基礎。比方說，將來的 OCAP 規格，將會是 ACAP 規格的一個衍生終端規格 (terminal specification); 除了 ACAP 原本所定義的 API 以外，OCAP 還額外增加了兩個自己專屬的 Java package - org.ocap 與 org.atsc。另外，ACAP 標準是一個和底層的傳輸系統獨立的 middleware 標準，所以，可以被使用在地面波、cable、與衛星的數位電視系統上。

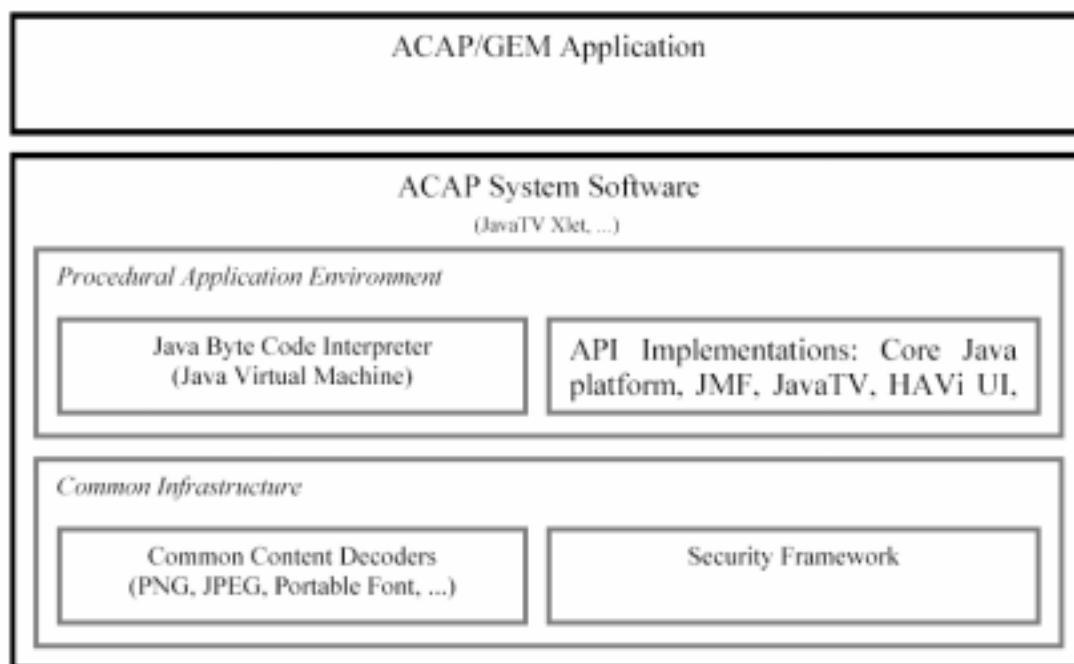


圖 2-15、ACAP-J 的系統架構

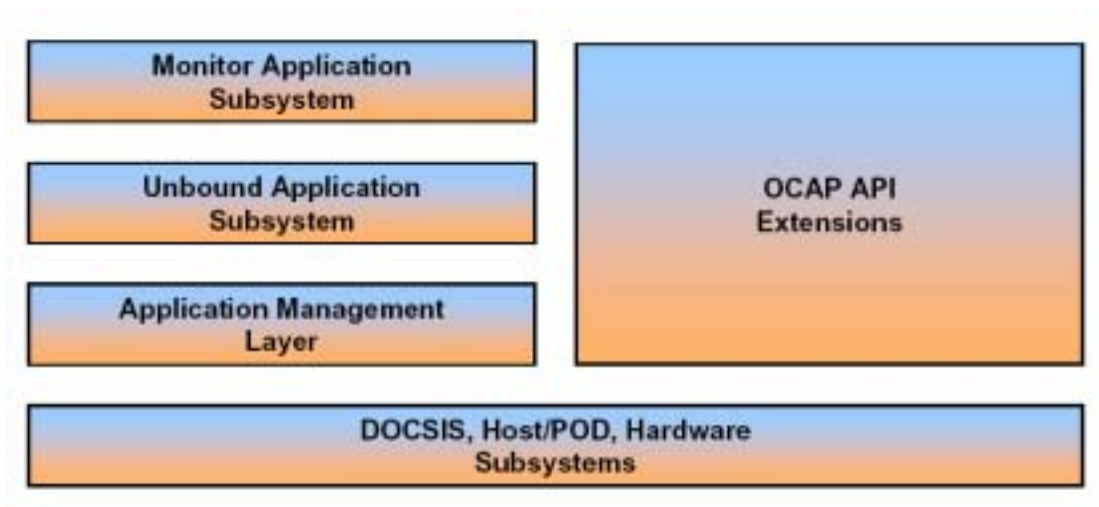


圖 2-16、ACAP-C 的子系統的架構圖

目前, ACAP 最新的版本是 2004 年 2 月發表的 candidate standard, 它與現有的 GEM、DASE-1、以及 OCAP 的相容性詳述如下:

- 與 GEM 1.0.1 版相容
- 與 OCAP 1.0 版相容 (但必須要加入 ACAP-C 子系統)
- 與現存的 DASE-1 無法後向相容: ACAP 所定義的是 DASE 功能上的 superset。

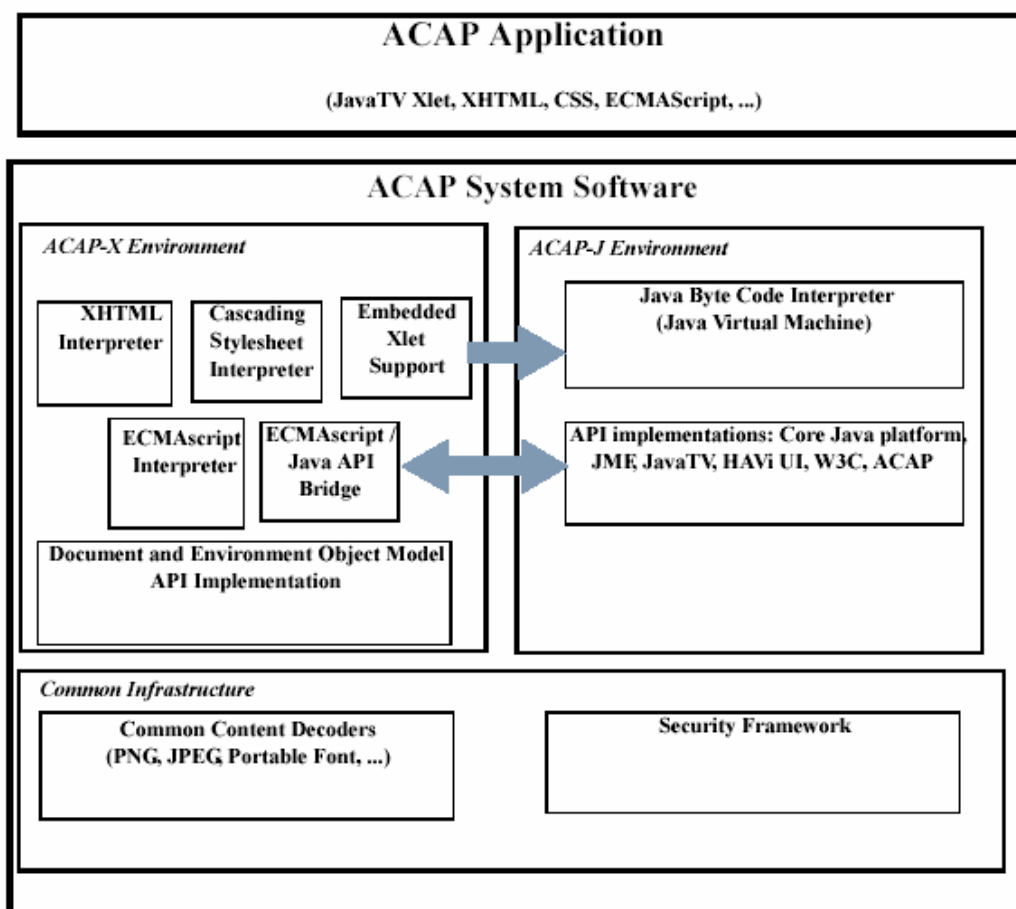


圖 2-17、ACAP-J 與 ACAP-X 合用時的系統架構圖

ACAP 上的應用程式執行環境，也一樣被分成 EE 和 PE 兩類。ACAP 上的 EE 被稱作 ACAP-J (ACAP Java Applications and Environment); 而 ACAP 上的 PE，則被稱作 (ACAP-X)。圖 2-13 與圖 2-14 分別摘要了 ACAP-J 與 ACAP-X 所採用的標準與技術。ACAP-J 的應用程式是 Java TV 的 Xlet，由編譯過的 Java byte code，以及其他的多媒體內容，如 graphics、video、audio 所組成的; ACAP-X 應用程式則是由 XHTML markup、style rules、script，以及 graphics、video、與 audio 所組成的多媒體文件。

ACAP 提供了兩種不同的 profile，第一種是 ACAP-J only profile，第二種則是 ACAP-J and ACAP-X profile。圖 2-15 是 ACAP-J 的系統架構圖 (ACAP-J only profile)，因為 ACAP-J 是 GEM-based 的規格，所以 MHP 1.0 裡的基本元件，在這裡都找得到，如 Java virtual machine、personal Java core libraries、JMF、JavaTV API、HAVi UI API...等。另外，為了支援 OCAP 1.0 所新增的功能，ACAP 定義了 ACAP-C 子系統。如圖 2-16 所示，ACAP-C 子系統，除了支援 OCAP 所定義的額外的 API 之外，也支援 OCAP 所定義的另外兩種應用程式 – monitor application 與 unbound application。ACAP 支援以下三種 application mode: 第一種是 service bound application，這種類型的應用程式，會附屬於某個特定的 service 之下，它的 lifecycle 會隨著 service 切換而結束。第二種是 shared application，它和 service bound application 類似，兩者間的差異只在 shared application 可以附屬於多個不同的 service。第三種是 unbound

application，這種類型的應用程式不附屬於任何的 service，而且可以在電視關機，再重新開機後繼續執行。

圖 2-17 是 ACAP-J 與 ACAP-X 合用時的系統架構圖 (ACAP-J and ACAP-X profile)。ACAP 支援兩種 ACAP-J 應用程式與 ACAP-X 應用程式，彼此互動的方式。第一種是在 ACAP-X 應用程式裡，使用內嵌的 xlet。如圖 2-17 所示，ACAP 必須要讓 xlet 在 ACAP-J 環境下被執行，並透過 ACAP-X 環境與使用者互動。第二種互動方式，則是透過 ACAP-J 與 ACAP-X 之間的 inter-environment API bridge。這個部分是根據 MHP 1.1 的 API bridge 設計的，與 OCAP 的 PE-EE bridge 類似，所以在此我們不再詳述。

基本上，ACAP 和目前的 DASE-1 有不少的差異。所以，雖然功能上 ACAP 是 DASE-1 的 superset，可是 ACAP 沒辦法和 DASE-1 後向相容 (backward compatible)。ACAP 和 DASE 之間最明顯的差異，是在 interactive channel 上。DASE-1 並沒有 interactive channel，但 ACAP 則支援了 interactive channel。在 ACAP-J 與 ACAP-X 環境內的 interactive channel 上，可以使用以下的 protocol: IP、ICMP、UDP、TCP、與 TLS (Transport Layer Security)。另外，在 ACAP-X 的 interactive channel 上，除了上述的 protocol 之外，還可以使用 HTTP 1.1。

2.8 日本 MHP 相關標準介紹

在日本，負責制訂數位電視多媒體廣播標準的組織是 ARIB (Association of Radio Industries and Businesses)，它是一個由分屬於無線通訊、廣播、電子通訊等產業內的公司，所組成的非官方組織。ARIB 在 1999 年 10 月，發表了 ARIB STD-B24 - “Data coding and transmission specification for digital broadcasting”，裡面定義了 BML (Broadcast Markup Language)、service information、與傳輸系統...等。BML 是一種宣告性的應用程式環境 (declarative application environment)，採用的技術類似 MHP 與 DASE，包含了: XHTML、CSS、DOM 與 ECMAScript。後來，ARIB STD-B24 又再增加了以下的幾點功能：

- 支援有數位媒體儲存能力的接收器
- 支援地面波廣播
- 供地面波廣播行動接收使用的 mobile profile
- BML 2.0: BML 2.0 又稱為 B-XML，支援大部分 XML 的功能，包含使用者所定義的 DTD。B-XML 的文件，會在接收器裡，經由 XSLT 的轉換，變成 BML 的文件。

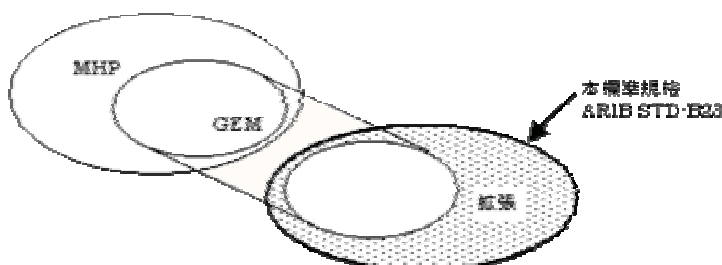


圖 2-18、ARIB STD-B23 與 GEM 及 MHP 之間的關係

目前，ARIB STD-B24 最新的版本是 2003 年 7 月發表的 3.8 版。日本於 2001 年 12 月開始了包含互動多媒體服務的衛星數位廣播。在 2002 年 10 月，全日本已經有 158 萬個包含 BML

engine 的數位電視接收器。2003 年 12 月，日本開始在主要的都會區(東京、大阪、名古屋)開播地面波數位電視，預計將在 2006 年底以前擴展到其他地區。

另外，在數位電視廣播裡的程序性內容 (procedural content) 方面，ARIB 於 2003 年 6 月發表了 ARIB STD-B23 – “Application execution engine platform for digital broadcasting”。如圖 2-18 所示，ARIB-AE 1.0 是 GEM 1.0 的一個衍生終端標準 (terminal specification)。ARIB-AE 1.0 與 MHP 1.0.2 的差異簡述如下：第一、在前面已經提過的，ARIB-AE 裡所使用是 ARIB 的 data broadcasting 規格，並不是 DVB 的 object carousel。第二、擴充了在日語環境裡必要的規定，如日文字型、切換語言聲音的 API、以及字型合成 (符號製作) ...等。

2.9 中國 DTV middleware 的發展現況

中國目前數位電視 middleware 的標準尚未公開，目前公開的資訊只有：

- “數字電視標準項目工作組 – 第四組：數字電視接口工作組”，相關的組長、副組長以及成員名單。目前，數字電視接口工作組的組長是中視聯公司的廖宏濤博士。
- 數字電視接收標準體系框架 – 中間件標準：
 - 數字電視接收設備軟件模型研究
 - 數字電視軟件功能與性能規範
 - 數字電視中間件技術要求與應用程序接口規範
 - 數字電視中間件軟體可恢復性與可靠性要求
 - 數字電視接受設備軟件一致性要求與測試規範
 - 數字電視中間件與 CA 接口規範
 - 數字電視中間件與驅動層軟件接口規範

不過，根據 Sun Microsystem 的 Bill Sheppard 所發表的演講 – “Open Standards for Television Services – MHP, OCAP, and the Global Movement for Multi-Network Interoperability” 指出，中視聯目前正在訂定的數位電視 middleware 標準，是以 MHP 為基礎的 middleware 標準。而且，GEM/ITU-T J.202 已經可以說是，全球共通的互動數位電視 middleware 的核心標準。因此，中國正在制訂中的 middleare 標準，極可能也是以 MHP 為基礎發展的。

另外，大陸的 Shenzhen Cable TV 已於 2001 年 11 月，和韓國的 Alticast 簽約，建構中國第一個採用 MHP 的互動電視服務。Shenzhen Cable TV 有 100 萬個用戶。Alticast 將提供 authoring tool、應用程式 (EPG、mosaic TV、T-commerce、weather、news、messaging、advertisements)、後端的互動服務自動化系統、以及 set-top box middleware 給 Shenzhen Cable TV。

2.10 MHP 其他相關技術問題分析

關於 MHP 的 DSM-CC 是否能支援中文檔名，請參考圖 2-19 的 DSM-CC protocol 架構圖。MHP 所採用的是 DVB datacasting 標準的 DVB object carousel，它建構於 MPEG-2 的 DSM-CC object carousel 之上。MPEG-2 的 DSM-CC object carousel，其 user-to-user Interfaces 是用 OMG 的 CORBA IDL 定義的。

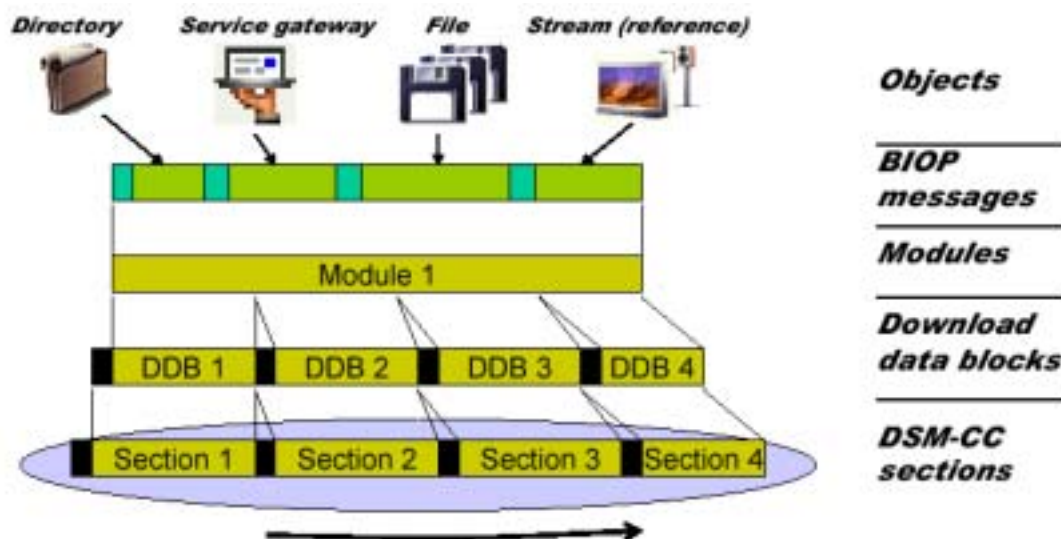


圖 2-19、DSM-CC 的 protocol 階層圖

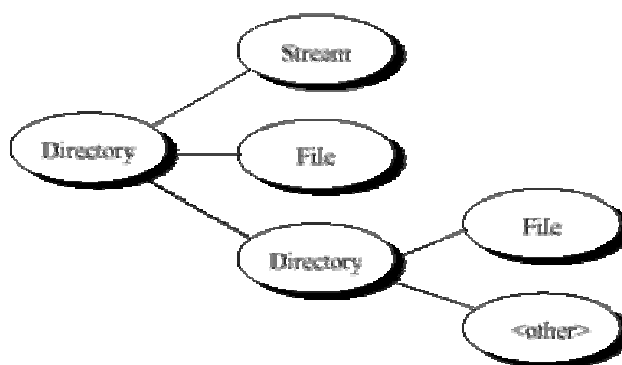


圖 2-20、DSM-CC 提供服務的 object 階層圖

圖 2-20 是以 CORBA IDL 所定義的，DSM-CC object carousel 所提供服務的 object 的階層圖。從 DSM-CC object carousel 被廣播下來的 CORBA 物件，會被 BIOP (Broadcast Inter ORB Protocol) 這個通訊協定承載。BIOP 再透過 DSM-CC 的 data carousel 傳送。

圖 2-21 是 DSM-CC object carousel 定義檔名的 CoSNaming 模組，此模組使用 string 這個 IDL 所支援 type 來儲存檔名。在 CORBA 的 IDL 裡，單位元字元字串的 type 是 string；雙位元字串的 type 是 wstring。因此，DSM-CC object carousel 的檔名是由單位元字串所構成的檔名。中文 Big 5 碼是一種單位元-雙位元混用的編碼方式，如要 DSM-CC object carousel 可以用 Big 5 檔名，則 DSM-CC object carousel 對檔名的處理，必須滿足以下的兩個條件：一、被 Big 5 編碼所使用的控制碼，在 DSM-CC object carousel 仍被當成一般字元處理。二、檔名內的字元，值若超過 127，仍會被當成一般字元處理。因為 DVB 在“DVB - DVB specification for data broadcasting” (ETSI EN 301 192)，也沒有討論 DSM-CC object carousel 檔名處理方式的敘述，因此，我們認為，DSM-CC object carousel 是否能支援中文檔名，事實上完全由實作者決定。

```

module CosNaming{
    typedef string lstring;
    struct NameComponent {
        lstring id;
        lstring kind;
    };
    typedef sequence<NameComponent> Name;
    // note: BindingType equates to the CORBA enum definition
    // while allowing extension for DSM-CC implementations
    typedef unsigned long BindingType;
    const BindingType noobject = 0;
    const BindingType ncontext = 1;
    struct Binding {
        Name binding_name;
        BindingType binding_type;
    };
    typedef sequence <Binding> BindingList;
};

```

圖 2-21、DSM-CC object carousel 定義檔名的 CoSNaming 模組

另外，關於在 MHP 上顯示中文字幕的問題，可以分成兩個子問題探討。如果中文字幕採用 DVB subtitle (DVB - subtitling systems, ETSI EN 300 743) 方式傳送，因為 DVB subtitle 可以用 Unicode 編碼，因此，只要接收機上的 DVB subtitle decoder，可以正確將 Unicode 內的 Big5 字集透過 GUI 顯示，提供中文字幕就不會有問題。至於 MHP 程式顯示中文的問題，因為 MHP 是一種 Java-based middleware，而 Java 內部處理字元也是用 Unicode，因此，中文在 MHP 上不能正確顯示的原因，應該是 Java AWT library 的字元引擎的實作方式有誤所致。

最後，我們在此討論數位電視導入 H.264 等新影像壓縮方式後，MHP 標準是否能繼續適用的問題。因為 MHP 是一種和底層的 transmission 機制相依的互動數位電視標準，因此，如果在導入 H.264 之後，transmission 機制改成非 MPEG-2 system layer (ISO/IEC 13818-1/H.222) 的形式，我們認為 MHP on H.264 的 API 會產生類似像 GEM (globally executable MHP 類似的調整)，下列的 API 可能會和 MHP 原本的 API 不一樣: service information API、data broadcasting API、application signaling API、MPEG-2 concept 與 MPEG-2 section filtering API。但若 transmission 機制仍是 MPEG-2 system layer，採用的仍是原有的 DVB-SI 與 DVB object carousel 的話，MHP 的 API 應該不會因為 video coding 改用 H.264 而會產生什麼比較大的變化。

參、其他相關數位電視多媒體平台現況

3.1 MHEG5

ISO/IEC/JTC1/SC29 下的 Working Group 12 (WG12)，即為一般所稱的 MHEG (Multimedia and Hypermedia information coding Expert Group)。MHEG-5 是 1994 年所提出的一個互動多媒體電視平台規格，並在 1996 年成為 ISO 國際標準規格。在此標準中，定義了數種處理多媒體/超媒體資訊的編碼表示方式。

MHEG 將「多媒體」定義成為幾種媒體型態的表示，如 audio、video、text 與 graphics。當系統增強，允許 DTV 用戶來瀏覽一畫面中的物件時，MHEG 採用專用的 link 作連結。除了定義 hypermedia 與 multimedia 物件，MHEG 標準也關注了這些物件由儲存設備到通訊與廣電網路的交換情形。

MHEG5 的概念類似 Internet 上的 HTML，用以描述數位廣播電視環境下互動應用中的物件及其行為，然後由內建在接收設備中 MHEG5 引擎來呈現其效果，很適合用於例如電子節目單、新聞、班車時刻等類型的互動資訊。

目前推動 MHEG 的 Cabot 公司是 Digital Terrestrial 委員會(名稱為 British Digital Broadcast, BDB)的成員，該組織定義和指定 MHEG-5 作為英國數位電視市場的中介軟體，並且有廣泛的專門技術來推進 MHEG-5 規格。Cabot 經常性參與 DTG 會議，討論 MHEG-5 標準之增修，Cabot 也介入許多新興 DTV 軟體標準與技術開發。

MHEG5 曾經爭取成為 DVB 的互動電視標準，最後仍不敵功能更為完整強大的 MHP 標準。然而由於 MHEG5 核心程式碼較小，較適合內建於低成本的數位電視接收設備上，而仍然能夠呈現相當豐富的互動電視應用環境。

由 BBC/Crown Castle 所營運的英國 Freeview 地面無線電視廣播服務指定 MHEG-5 為其互動電視服務的平台，各家接收設備/機上盒/數位電視機的製造商可以選擇性的將 MHEG5 核心建入其產品之中；至於其他比較主要的應用環境目前也多集中在英國。

MHEG-5 DTV 系統互動應用與獲利來源，主要有下列項目：

- Internet Digital TV
- (Near) Video on Demand
- Home-Shopping
- Interactive Advertising
- On-Screen Information
- Program Retailing and Merchandising
- Electronic Newspapers
- Home Banking
- Internet/Web Access

Instant Viewer Snapshot Polls
E-Commerce applications
On Line games

許多不同形式的互動應用透過不同的互動服務來進行，以下提供一些以互動技術能力來強化電視節目的實例：

Electronic Programming Guides
Animated cartoon-like graphics
Text (formatted with control, font, size, style, etc)
Buttons and other interactive hot spots
Hyper text-pages of linked information
Photographs
Video in a window or as a full screen backdrop
Still graphic illustrations
Interactivity
Digital Multi Media Teletext

MHEG-5 Applications

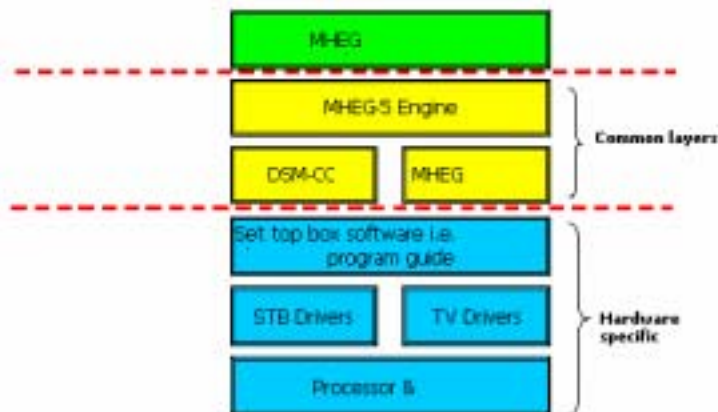
- Digital Teletext
- Info txt: push&pull model
- Program guides
- Stream select: viewer power
- Interactive advertising
- Educational
- Play along game show



MHEG-5 技術評估

MHEG 被設計能以系統最少的資源即可支持，這使它成為一個非常適當的數位機上盒中介軟體產品。MHEG 由七部份組成。一個 MHEG-5 應用，基本上是一套多媒體和超媒體物件，常駐在電視營運商頭端的電腦中，並被轉換成位元流格式並在寬頻網路上廣播。在接受端，一個數位機上盒與其軟體元件統稱為 MHEG-5 引擎或虛擬機器，它從廣播而來的數字式資料流中萃取 multimedia/hypermedia，解譯資料並且顯示在訂戶的電視螢幕上。除了處理接踵而來的資料之外，MHEG-5 引擎也負責同步並支持與訂戶的近端互動。

System Architecture



MHEG-5 軟體與電腦硬體之系統概要圖

數位機上盒硬體需求

MHEG-5 引擎的記憶體需求 (footprint) 約僅 300KB。因而，數位機上盒與處理器以每秒 40 百萬指令的速度 (MIPS) 運行，並結合 4MB RAM 與 2MB Flash 記憶體可輕易地支持這個軟體模組。從商業角度，英國數位地面廣播即已採取了 MHEG-5。隨後，在英國的人們，若想要接受數位地面廣播並與多媒體結合的內容時，將需要購買或租賃數位機上盒或數位整合型電視，來執行 MHEG-5 引擎。

DIGITAL TV MIDDLEWARE SOLUTIONS MEMORY REQUIREMENTS			
Solutions	Processor speed	RAM (Mbytes)	Flash/ROM (Mbytes)
MHEG-5	50MHz+	4	2
Open TV	50MHz+	4-8	4
Media Highway	50MHz+	4-8	4
MHP Interactive Broadcast Profile	80-130MHz+	8-16	8
MHP Internet Access Profile	150-200MHz+	16-32	16
Note: All figures quoted are subject to change and to applications in operation.			

Cabot 的 MHEG-5 引擎被設計和發展為數位式電視商品。Cabot MHEG-5 的主要特色為：

- 不斷被開發和持續發展成為為英國 DTT 技術規格
- 原始碼對各種硬體平台最佳化
- 記憶體管理模組
- 操作速度
- 原始碼可攜性
- 完整 Cabot DSM-CC 模組為標準
- 競爭性價格

MPEG Video 支援
PNG 圖形格式支援

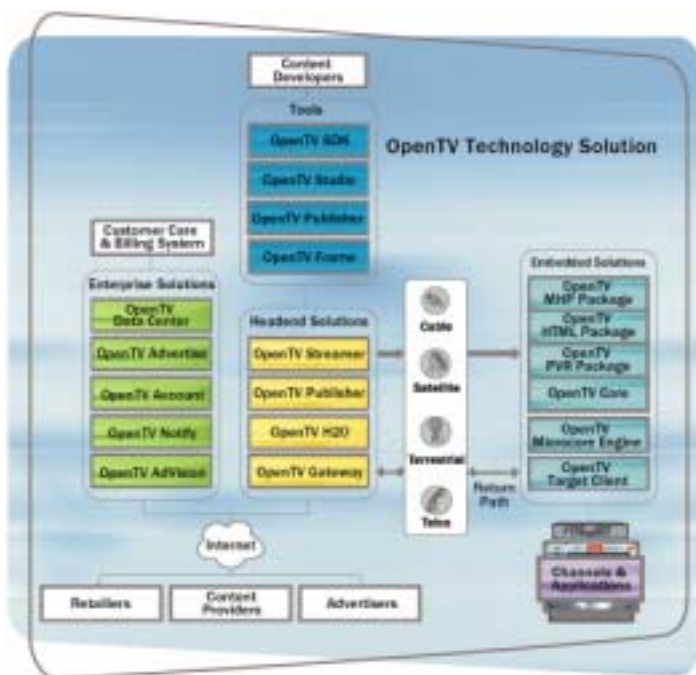
Cabot 提供廣泛的開發設施，努力與客戶採行伙伴關係，並提供端對端解決方案。包含移植/轉換 MHEG-5 到客戶的硬體平台以及完整的 DTG 測試。並已協助 Telewest 設計、開發了一系列完整的編輯與廣播管理系統。

3.2 OpenTV

OpenTV 成立於 1994 年，是目前主要的互動電視多媒體平台廠商之一，提供由節目製作、後台系統、前台系統、頭端廣播、到終端設備系統架構的 End-to-End 平台技術及整合服務。

當年 OpenTV 得到英國衛星直播業者 BSkyB 採用在其相當成功的 Sky Digital 的互動電視服務，並陸續取得十多家衛星及有線電視業者的採用(如法國 TPS 美國 EchoStar 澳洲 AUSTAR 等)，成為市場佔有率最大的互動電視系統平台，在全球有超過五百萬台內含 OpenTV Middleware 的 Set Top Box。

OpenTV 的技術開始即針對廣播的環境設計，非常適用於單向廣播的互動應用，也因此受到眾多廣電業者青睞；不過 OpenTV 的 Middleware 技術主要是以專屬的 o-code 語言開發應用軟體，並未對外公開其規格，而是將其包裝為各類的開發及營運系統產品；OpenTV 本身也同時提供為客戶開發各類應用軟體的服務，以擴展其業務角度。不過由於 OpenTV 相關產品及服務之成本相當高，近年來一些電視及系統業者在檢討互動業務的成效後，裁減了許多相關服務，對 OpenTV 的業務上造成不少壓力。加上近年來業界開始對 MHP 的發展發生興趣，也迫使 OpenTV 的相關系統廠商規劃將平台轉移至開放規格的研究及發展。



OpenTV 的技術解決方案

iTV 就是要創造更好的電視，是提昇某個特定節目、廣告甚至一整個頻道的大好機會。不論你是系統營運商，希望擴展您的收視戶數量；或是廣告商，設法加強您的品牌；或是開發商，想要創造一種新的動態 iTV 互動應用；或是程式員，想要測試互動應用的流暢度；OpenTV 都可以幫助您，使這些事情實現。

有線與衛星數位電視市場是高度競爭的，因為營運商在全世界尋找各種方式來區隔他們所提供的服務，以吸引新顧客，建立忠誠度，而產生源源不斷增加的收入，許多人發現到 iTV 是可以做到的。

成功地發射 iTV，業者需要的是可靠、大小可調整、和有實際驗證的一個技術平臺，一個已由內容提供者採用的平台，包括節目製作和廣告商。不論當前的基礎設施為何，OpenTV 都能提供平臺來處理操作和交易上全功能，可產生獲利的 iTV 服務的種種要求。當 iTV 演進到技術、操作、和專業支援時，OpenTV 具備了有實際證明的 iTV 應用並且能支援廣播網路和商業事務。

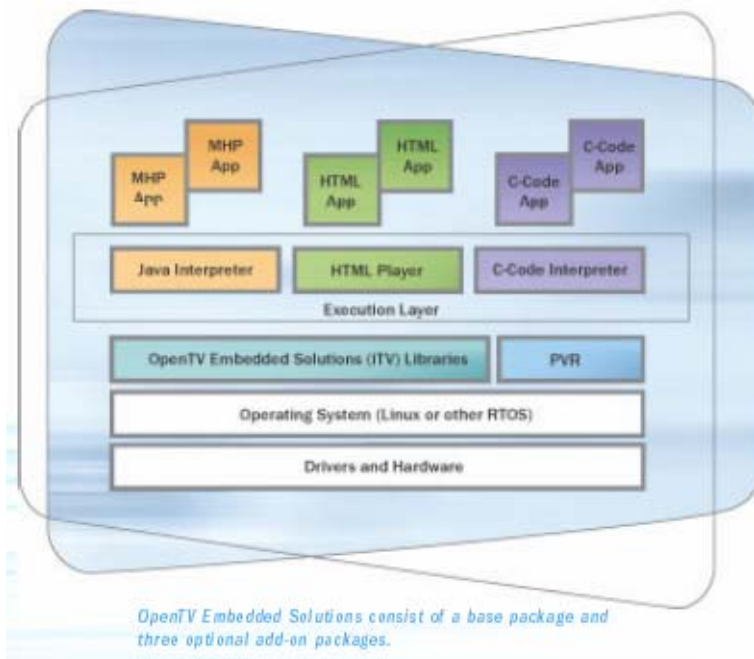
廣播與節目製作者藉由援用 OpenTV 開發出的上百種互動式程式和應用經驗，並由全球閱聽者參與，將可建立閱聽者的忠誠度。通過互動式廣告可以引起新的源源不斷的收入，因而廣告主能運用 OpenTV 驅動的網路，針對特定觀眾，瞄準特定消息，並且能通過節目贊助機會，創造新的實質廣告資產。有了更多參與的閱聽人和目標性互動式廣告，將有一套新的競爭優勢可以來提供廣告商運用，包括直接和可測量的廣告效益回應。

廣告與行銷人員根據產品的品牌宗旨，直接地與消費者參與在一個活躍的關係中，是廣告終極的理想目標。現在試著考慮一個贊助者的潛力，在螢幕上不會轉台的增強型互動節目，特別設計給特定的品牌！由此不難得知為什麼許多全球最大的廣告商已跳入 iTV，以 iTV 作為一個品牌的工具以及直接反應的載具。OpenTV 根據個別選擇並產生直接可測量的回應來提供資訊的能力，這無疑的是一種革命。並且 OpenTV 的廣告服務全面性提供，包括一切從資源分割和提供目標訊息以創造強制互動通訊，處理和分析反應，形成一種單一、易處理的解決方案。

當 iTV 市場全球性地增長，越來越多公司將依靠 iTV 應用或經驗上豐富的選擇，以交付給他們的閱聽者。並且創造這些新 iTV 經驗的工，就落在全世界應用開發社群的手中。OpenTV 給開發商一個有效和高效率的方式，以接觸到快速增加的全球觀眾。透過 OpenTV 的夥伴計畫，開發商得以存取廣大範圍的資源幫助，以協助他們成功。

OpenTV 與一些在產業界最大的設備製造商有商業往來。用 OpenTV 的解決方案，能大幅改進產品上市時間。藉著縮短 iTV 機上盒軟體的開發週期，並且 OpenTV 的專業服務團隊，將使在任何設備上的整合快速而簡單。通過 OpenTV 中介軟體解決方案，製造商能在產品中整合最新的功能，並超越競爭的產品。

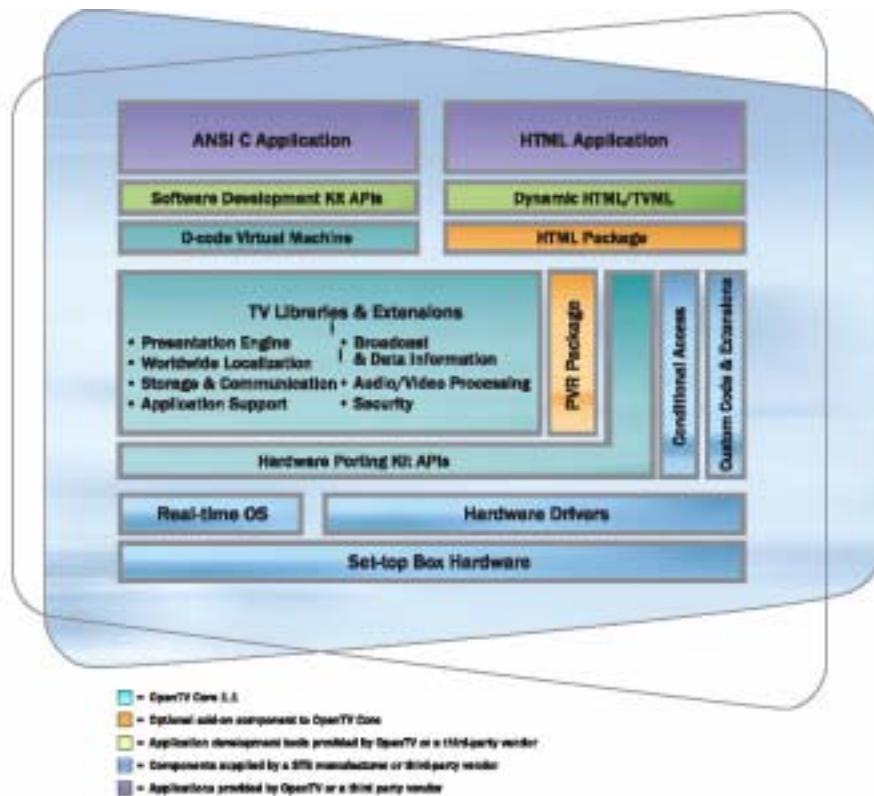
OpenTV Embedded Solution



OpenTV embedded solution 含有兩層：

Core middleware: 提供營運商經濟、有效的 STB 硬體平台，包括由撥接到寬頻、低階到高階的各種選擇，Core middleware 結構上是與硬體無關、模組化、可擴充的，因此可爭取產品最低成本、最快上市，也提供製造商快速開發客製化平台。

Execution layer: 提供業界標準相容性應用，如 C、HTML、JAVA 等程式，C 程式之執行引擎由 base middleware 提供，JAVA 與 HTML 則為選項。



OpenTV Middleware Architecture

OpenTV embedded solution 含有以下 package：

1. OpenTV core package:

此 package 提供 middleware core，並提供全功能的 iTV 環境，有下列特性：

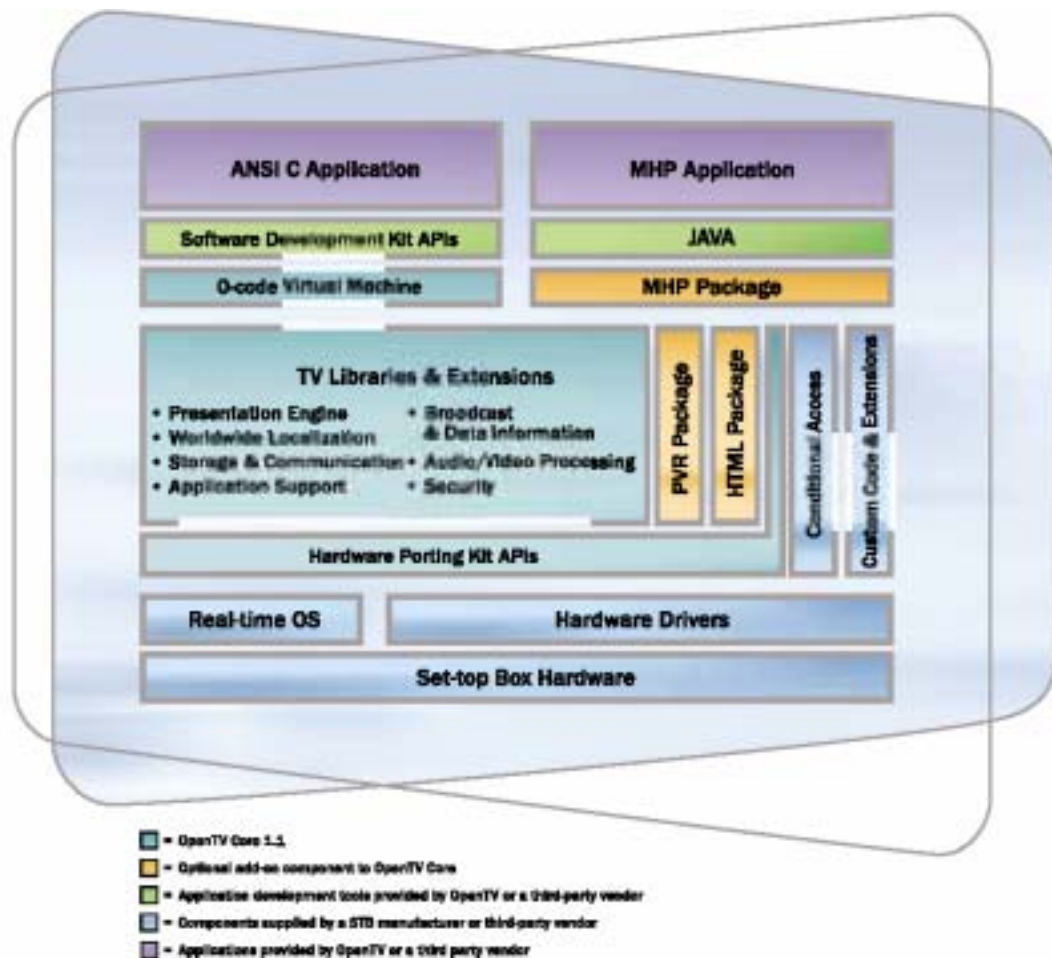
- *依標準開發：可執行 ANSI-C 應用程式，middleware 擴充與驅動程式，可提供 overhead 最小的 STB 執行環境
- *可擴充，模組化，可移植：使開發者可快速增加新功能，易於維護
- *支援 thin-client STB: OpenTV 可執行於有限的 CPU 運算與有限的記憶體環境中
- *硬體、OS 獨立性：OpenTV embedded solution 設計上使所有的 OpenTV 應用程式均可執行，並且可移植到多種不同的 OS 上
- *連結性：提供 PPP, DOSIS, DVB-RC 等支援
- *網路獨立性：可視用在衛星，有限或電信網路上
- *可靠性：已超過 4 千 2 百台 OpenTV STB 已在運作
- *在地化：可支援多種語言

2. OpenTV HTML package

可支援 HTML, JavaScript, TVML, CSS, DOM, ECMAScript 等協定

3. OpenTV DVR package

可支援 VCR 播放控制與錄影功能



OpenTV MHP Software Architecture

4. OpenTV MHP package

此 package 擴充 OpenTV，以支援 DVB-MHP，現支援版本為 1.0.1，並可整合 DVR 與 CA 功能，可支援多種 OS。

3.3 NDScore

中介軟體是開發數位運營潛能的關鍵，它不僅能支持一些必要的服務，如電子節目表(EPG)、互動和 PVR，而且還將不同的機頂盒統一在一個強大品牌之下。要充分發揮這一潛能，需要選擇正確的中介軟體，一個令營運商擁有開放的硬體平台及眾多應用開發合作夥伴的中介軟體。

NDS 公司提供的 MediaHighway™ 中介軟體解決方案裝備了全球各類型數位平台上的一千六百多萬台機頂盒—有線、衛星、寬頻和數位地面網絡。

MediaHighway 已有八年以上的豐富實際應用經驗，全球有 25 家以上的營運商使用 MediaHighway，包括法國 Canal Satellite、意大利 Sky、拉美 Sky、印度 StarTV、美國 RCN 和以色列 YES。它還被 DirecTV 選為其新互動平台的基礎組件。

MediaHighway 擁有以下明顯的優勢：

- *成熟的解決方案—MediaHighway 已與 30 多家領先廠商的數百種機頂盒型號進行了整合。
- *支持先進的服務和應用 —包括 VOD、PVR、寬頻 IP 和家庭網絡。
- *支持百餘種應用 —由數十家應用開發商所創建
- *基於標準的工具，簡化開發 —包括 MHP、HTML、和 Java™—令您開發更高效能的 iTV 應用。
- *與業界領先的 NDS 解決方案相集成 —包括 VideoGuard®有條件接收系統、XTV™ 個性化電視系統、Value@TV™ 互動架構、和 NDS Guide™ EPG。

MediaHighway 的兩套解決方案為在各種新舊機頂盒上實現豐富功能提供了極大的靈活度，令業界當前和未來的數字電視投資物超所值：

- * MediaHighway Core™ —可以快速輕鬆地在高效費比機頂盒上實施有效的解決方案。
- * MediaHighway Advanced™ —符合 MHP 的、多任務中間件解決方案，運行於功能全面的機頂盒上。

MediaHighway Core

MediaHighway Core™是一個可輕鬆升級的中間件解決方案，能實施眾多的服務—如電子節目指南 (EPG)、互動應用、和 PVR—快速而經濟有效。

作為一款已成功部署在全球數百萬家庭中的成熟中間件產品，MediaHighway Core 只要求很小的機頂盒存儲空間，從而可在低成本機頂盒上快速實施「必不可少」的服務。

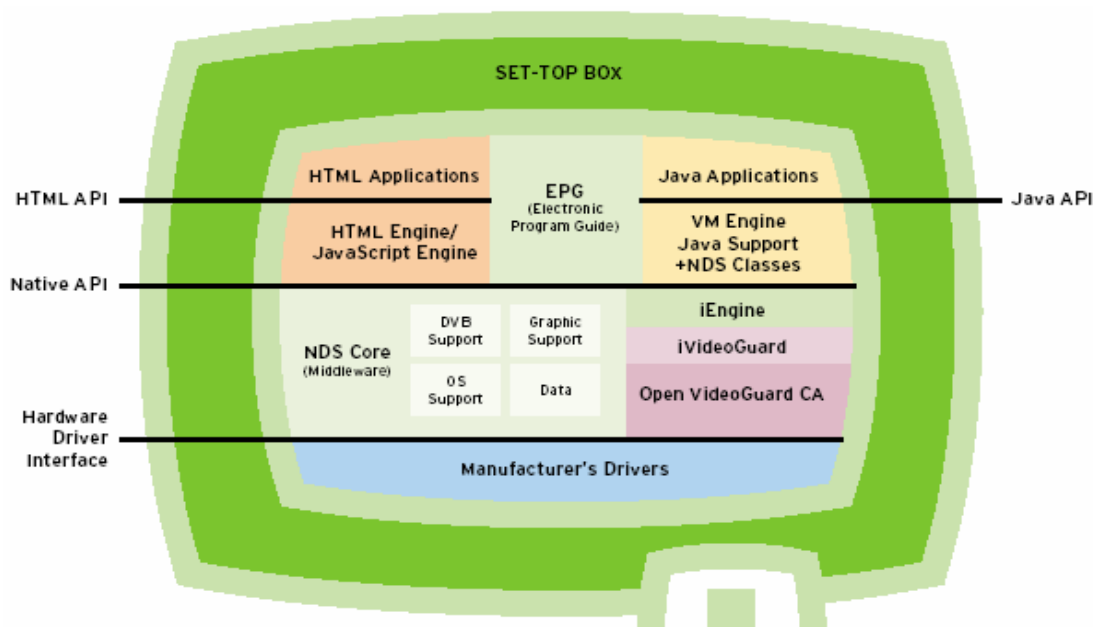
MediaHighway Core 可支持：

- *高性能 EPG
- *可迅速實施的 HTML & JavaScript 應用
- *用於低成本機頂盒的佔用內存小的虛擬機 (VM) 引擎
- *大量經濟有效的 STB
- *用於複雜字符集的 Bitstream Font Fusion Rendering Engine

關鍵的技術特點包括：

- *可靠、快速、高效的運行性能
- *以下支援 DVB 標準：
 - 字幕（編碼和位圖）
 - 馬賽克
 - DSM-CC 數據輪播
 - SI/PSI，包括 EIT 編排
- *完整的下載支持，使用高級的 boot loader 支持背景或前景下載
- *用於回傳通道的工業標準 TLS 安全支持
- *TCP/IP 和 DOCSIS 回傳通道
- *支援 24 位元彩色圖片，可含多個圖片層
- *採用多國語言支援的 Unicode 引擎

- *VOD 客戶端支持
- *高性能節目指南(EPG)
- *STB 整合服務，含全套文件檔和整合工具
- *支持 HTML 4.0、JavaScript 1.5、DOM 0、CSS1
- *VM 引擎，運行空間小於 64KB FLASH 和 64KB RAM（不包括應用所需的 RAM）
- *API 含式庫擴展，用於
 - 解譯 DVB SI
 - 256 色圖片庫
 - 音頻/視頻（包括可縮放視頻）
 - Unicode/雙字節字符集，用於多國語言支持



MediaHighway Advanced

MediaHighway Advanced(tm)符合主要的電視和 web 開放標準，包括 DVB-MHP、HTML 4.0 和 JavaScript。NDS 也致力於支持 OCAP/ACAP 標準。

利用 MediaHighway Advanced，可以既針對付費電視市場和水平免費廣播市場提供服務，其兼容性方面的優勢能帶來更多的內容及更多的機頂盒合作夥伴。

通過支持最新的數字電視技術和標準，MediaHighway Advanced 保護了中間件方面的長期投資。它支持 Java(tm)（Personal Java 1.2，優化的 Java 虛擬機）和 HTML。所有這些都完成無縫整合，為觀眾提供了更好的互動體驗。

MediaHighway Advanced 的架構和豐富功能使其成為非常適合複雜系統的解決方案，比如寬頻接取的家庭網絡等。此平台也因而能隨時應用消費電子領域最新的技術。

可擴展的模塊化架構

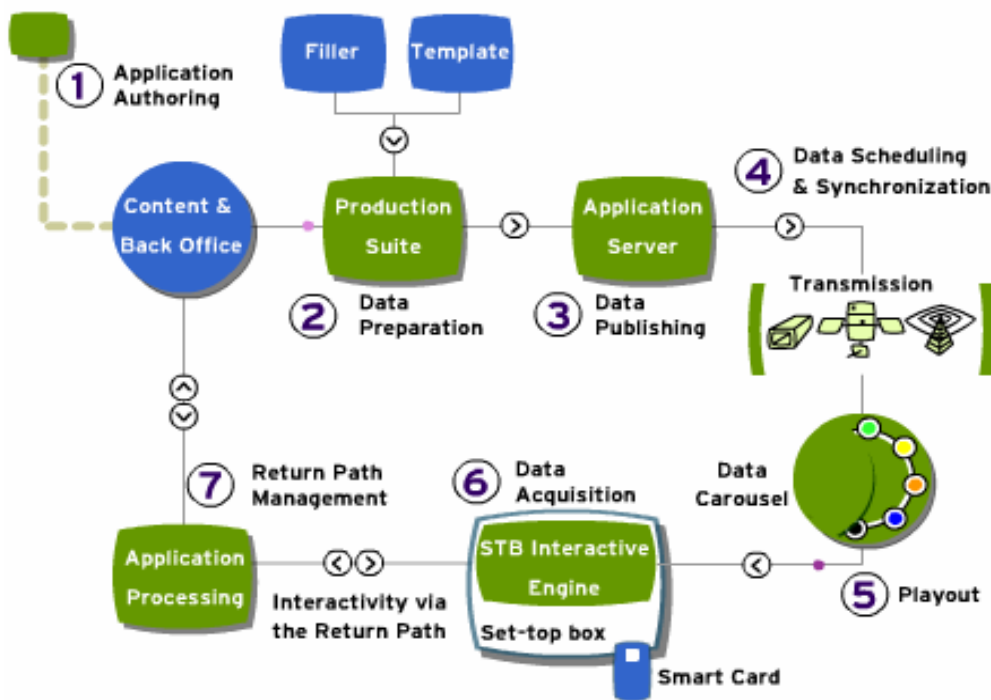
在 MHP 層之上，MediaHighway Advanced 提供了高層軟件組件，保障了符合平台品牌和外觀感受的用戶友好互動性。

- *PVR 支持
- *強大的高速緩存管理器
- *超鏈接和參考系統
- *一套特別設計的商務組件，保障了輕鬆開發和縮短項目週期

技術特點包括：

- *符合 Java DVB-MHP 1.03
- *支持新興的標準，如 OCAP
- *支持 HTML 4.0、JavaScript 1.4、DOM 1、CSS 2
- *靈活的架構，能輕鬆添加對附加連接設備的支持
- *一套完整的螢幕導航工具和業務組件
- *支持所有廣播/寬頻網路上的 DVB MPEG2 和 IP 傳輸

NDS 提供的解決方案



- (1) Application authoring: IADK 製作工具可創作 template，在 TV 螢幕上顯示應用，也可使用 NDS 或第三方的編輯服務與工具
- (2) Data preparation: Production suite 處理資料庫內容，轉換為可傳給用戶的格式
- (3) Data publishing: application Server 發佈資料，以傳送含有應用內容的 template
- (4) Data scheduling and synchronization: 互動內容廣播時，資料排程與同步要決定，並決定內容是否與影音節目內容協調
- (5) Payout: 輪播系統利用複雜的演算法來降低頻寬用量，並且循環傳送應用與資料給使用者
- (6) Data Acquisition: 在用戶端機頂盒中的元件「收聽」資料輪播，並由傳輸資料流中，拉入

所需的互動事件

- (7) Return path management: 用戶回應，回答民調或謎題，要求資訊，採購訂單，與其他使用者互動等等，將沿 return path 被傳到頭端，不包含在輪播中的資料也由此傳入 STB。

肆、我國數位廣播電視頭端系統平台特性研究與整合要素分析

4.1 衛星電視系統

雖然直播衛星是最早的數位電視，其接收及傳送都是數位的，但我國因為直播衛星市場不發達，約不到 10 萬戶，因此直播衛星轉與有線電視公司合作，採取所謂「廣義直播」的立場，從以前向頻道業者買節目並且自行租用轉頻器發射信號的方式，轉變到由頻道提供者出錢請直播衛星業者上下鏈的方式。

國內衛星直播電視最早於民國 88 年開播，雖然極力想藉由節目區隔以及加值服務（隨選視訊、互動電視），與有線電視的服務進行區隔，在偏遠地區有部分的客戶群，然而由於衛星電視平台營運成本過於高昂，客戶在還未普及之前就已經因為營運上的問題終止相關的研發。

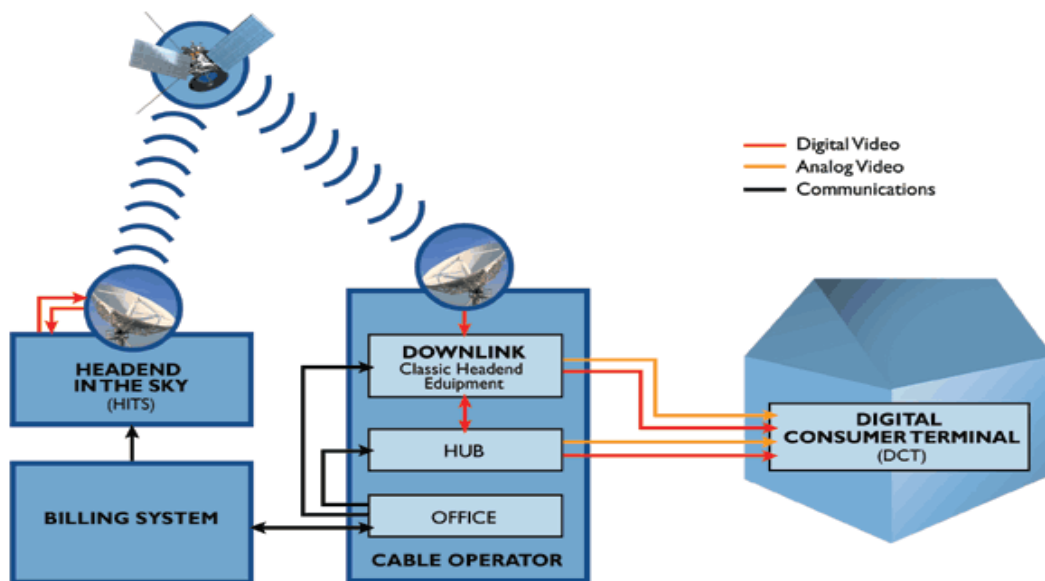


圖 4-1 衛星電視業者與有線電視業者合作模式 來源：” HITS Classic”

面對消費市場的不景氣，即使業者希望用 Digital pass-through，或是 Headend in the Sky 的營運模式經營，在有線電視佔有近乎獨佔地位的台灣媒體版圖，在取得有線電視業者支持之前，恐怕難有發展的空間，而固網業者過剩的頻寬，也讓衛星業者無法提供具有競爭力的價格，提供有線電視或其他業者頻寬轉賣的業務。

4.2 有線電視系統

目前國內五大有線電視集團包括東森、和信（中嘉）、太平洋、台基網、卡萊爾（台灣寬頻通訊）數位化的進度，以東森、和信、以及卡萊爾較有明確的計畫以及推動時間表。而目前則

以東森及和信數位頭端的建置較為成熟，在特定區域已經取得營運執照開始營運（參見「國內數位有線電視發展現況」）。

	東森	和信	卡萊爾
採用標準	歐規 DVB-C	歐規 DVB-C	美規 ATSC
節目鎖碼認證系統 (CA)	NagraVision	NDS Video Guard	Motorola Digi Cipher
中介平台 (Middleware)	OpenTV	NDS Core	--
機上盒供應商 (STB Vendors)	ADB, 欣象	ADB	Motorola
推廣地區	旗下全省系統台同步推廣	基隆及中永和等五個系統台	中部地區

表 4-1 國內數位有線電視發展現況 來源：「數位有線電視之發展與商業模式」，陳光毅（民 92）2003 數位電視及廣播國際研討會。

除開採用美規系統方案的卡萊爾集團以外，雖然國內兩大系統皆採用 DVB-C 標準的系統架構，不過在一些重要次系統選擇上（如定址鎖碼系統及機上盒）仍有相容性的困難，系統與系統間的訊號也由於封閉網路的特性而未曾進行相關整合，在將來可能在經營區域限制的法規變更而產生系統網路整合的需求時，已部署系統及設備的整合規模及難度並不因為皆採用歐規系統而有所降低。

基於上述之原因，以及數位有線電視系統業者專屬網路之特性，我們認為要制定一個一致性的規範，以整合數位有線電視系統業者現有的專屬系統網路，不易於本研究計劃之執行期間內完成。因此，關於本研究報告第七章內所提出之“有線數位電視多媒體共通平台技術規範”，僅為數位有線電視系統業者於開放式市場營運時之建議。但基於以下幾點原因，我們認為此工程規範草案，並不適合立刻公告。第一點原因、有線數位電視系統業者對電信法第四十六條第四項之適法性表示懷疑，但此種法務性爭議已超過工研院電通所之專業範圍，我們僅建議在本工程規範公告前須以其他途徑先釐清此項爭議。第二點原因、工研院電通所與有線數位電視系統業者於協商之後，對工程規範內以下的幾個重要論點，尚於計畫結案前仍無法達到一致之共識：

- 有線數位電視系統業者與工研院電通所，對所謂的開放性有線電視系統一詞之定義，彼此仍有相當大的歧異。工研院電通所認為所謂的開放性有線電視系統，乃指採行開放式之有線電視系統架構；但有部份有線數位電視系統業者認為，只要業者採行開放式之營運模式，其有線電視系統即可認定為開放性有線電視系統。
- 工研院電通所與部份有線電視系統業者，對於工程規範所應採行之條件接取標準，彼此間也無法達成一致之共識。工研院電通所認為應於頭端採用歐規 DVB 同步加密

(SimulCrypt) 標準界面，另於數位電視接收機上採行 multicrypt 方式 (第一種方式是採用 DVB-SSU 加上 EuroLoader，另一種方式則是採用 DVB-CI)，以便節目及用戶端之設備可以跨系統轉換，各有線電視業者仍可自行選用各公司的條件接取系統，例如：Nagra、NDS 等。但部份有線電視系統業者強烈認為，為了達到條件接取目的，應於開放式系統之數位電視接收機內，提供遵循 DVB-CI 標準之條件式接系統開放模組與實體介面。

第三點原因、有線電視系統業者強烈要求，如要制定具法律效力之有線電視系統業者工程規範，對其競爭者 (如 IPTV 之業者) 也須同時制定於法律面有同等地位之工程規範。此法律性問題已超過工研院電通所之專業範圍，但基於公平性之原則，我們建議在本工程規範公告前也須以其他途徑先釐清此項爭議。

第四點原因，有鑑於數位有線電視系統業者目前所營運之專屬網路之性質，與開放式市場之性質間有相當之差異，基於此項不確定性，工研院電通所亦尊重數位有線電視系統業者，於未來進入開放式市場營運時，依當時市場之實際情況以及業者間之共識，定位此建議規範之適用性之意見。

至於數位有線電視系統業者於其目前營運之專屬網路內，所應採行遵循之工程規範，我們建議電信總局採取技術中立之原則，並建議有線電視業者自行組成類似像美國 CableLabs 之標準制定組織，自行訂定相關之工程規範。

此外，基於以下幾點原因，我們暫時不建議強制數位有線電視系統業者於其專屬網路內導入 MHP：

- 國內數位有線電視系統業者，於 2~3 年前起，已經各自選擇了自己專屬網路內的互動數位電視中介軟體標準，並已投入相當多的成本建置頭端設備，以及鋪設數位電視機頂盒給終端客戶。是故，若於此時要求國內數位有線電視系統業者於其專屬網路內導入 MHP，會面臨互動數位電視中介軟體之轉移 (migration) 問題，業者須更換或更新已投資之頭端設備與數位電視機頂盒，這會是一筆為數不小的投資。另外，也須考慮業者於轉移中介軟體標準時，可能會因短期之混亂而喪失某些商業上之競爭優勢。
- 若考慮互動數位電視平台開放式標準，MHP/GEM 已經可以說是全球最受矚目之互動數位電視平台標準。然而，綜觀全球採用 MHP 之現狀，我們可以發現，目前會採用 MHP 的數位電視廣播業者，仍是以採行開放式市場商業模式之無線數位電視業者為主，例如芬蘭、德國、義大利均是如此。

4.3 無線電視系統

根據民國 90 年「數位電視傳輸標準公聽會」的決議，電視學會向交通部提出欲採用歐規之 DVB-T 為我國標準，故目前五家無線電視台在數位電視廣播上皆已選擇 DVB-T 系統設備並已完成開播初期環島廣播網路的建置。由於無線電視台的營運策略初期皆延續以往無收費的公共廣播作法，收視戶取得接收設備無法掌控，所以目前各家無線電視台所採用的廣播訊號參數大致一致，在初期廣播網路的整合上尚稱單純。不過就將來可能再釋出頻譜的情況下，為了提高整體營運效率，將來對無線電視廣播系統的整合，包括工程/導引頻道、接收設備共

同規範、服務及設備認證作法、數位版權保護及管理，都需要加以規劃。

關於有線必載之技術面問題，可大致分類為四層次。第一個層次是完全不作 Transcoding，將無線或是衛星之訊號直接轉送於有線電視網路上(可再經 Up-Converter)；這種做法的優點是訊號直接轉送，不需外加設備器材，缺點則是需使用無線或是衛星之接收設備，與有線系統所使用之接收設備不相容。

第二個層次到第四個層次則是經由 Transcoding 作訊號轉送，有以下三類方式。

一、Transmodulation

說明：將無線或是衛星之訊號調變(Modulation)轉換為有線系統之訊號調變(QAM Modulation)再轉送於有線電視網路上

優點：可使用 DVB-C 之 STB 作為接收設備

缺點：需於有線頭端設置 Transmodulator, 仍有二系統間 bitrate 不相符之困擾

二、Transmux

說明：將無線或是衛星之訊號解調變並解多工(De-Mux)後再重行做多工組合(Re-Mux), 加上 DVB-C 之訊號調變(QAM Modulation), 再轉送於有線電視網路上

優點：可使用 DVB-C 之 STB 作為接收設備, 並可解決二系統間 bitrate 不相符之困擾

缺點：需於有線頭端設置 Transmux 之設備, 仍有二系統間控制系統(如 CA, Middleware, EPG)不相符之困擾

三、Transcontrol

說明：在 Transmux 上再作操作控制系統之轉換

優點：可完全融入 DVB-C 之系統運作, 收視戶得在一致的操作環境中使用

缺點：需於有線頭端針對無線電視台之節目設置全套之設備

未來的趨勢是朝向製播分離的方向發展，必載的問題就不再是不同平台之間節目頻道的必載，而是上下游之間節目頻道在各種傳輸平台上的必載。「製播分離」政策，即電波訊號的傳輸與節目製作播送分開由不同公司經營。目前德國與英國各有一、二家電信公司經營傳輸業務。英國公共電視 BBC 於 1997 年出售工程部，不再經營傳輸業務，僅經營節目頻道。我國目前五家無線電視台個自建構傳輸網路，並非製播分離。「製播分離」在數位化後將有新的意義，以下我們釐清幾個概念。

- 多工器(MUX)的經營：多工器將在數位化之後，形成節目與傳輸之間的新營運層次。製播分離之後的電視節目或電視台，在多工器指配下，只經營頻道與節目內容。
- 公共化的無線頻譜與多工器：無線頻譜本身為公共資源，兼負公共服務的責任。因此無線多工器的經營亦須兼顧公共化的責任。
- 公共化的節目與必載：製播分離制度下節目將分類為公共化與商業化，公共化的電

視節目主要係由政府提供資源製作，因此有必要在各傳輸平台上必載(無線、有線、衛星、電信)。

4.4 寬頻網路電視及 MOD

隨著網際網路頻寬越來越大，內容的需求也越來越多，網際網路電視多半是資訊公司出身，本身掌握網際網路串流（Streaming）技術，進而可以向消費者收取收視費或是與固網業者或 ISP 聯盟提供服務。

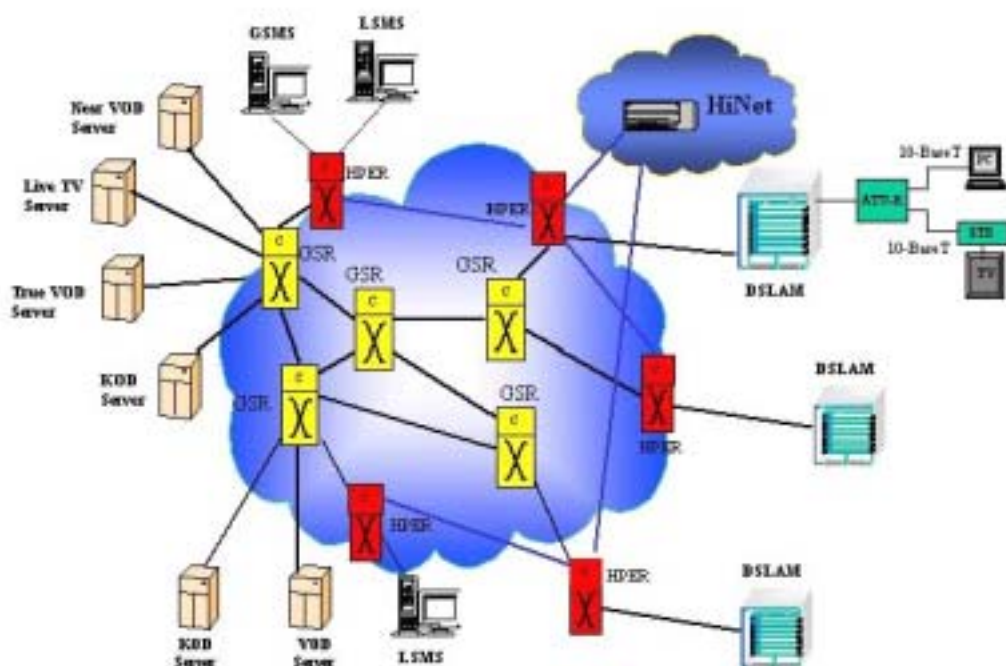


圖 4-2 中華電信 MOD 網路架構

如上圖所示，目前各種寬頻電視所使用的系統架構大同小異；至於串流技術的選擇，除了與數位電視相同的 MPEG2 技術外，目前主要是由 MPEG4 標準所衍生的各種技術規格，以及其他一些同樣由 MPEG4 技術所衍生出來專屬技術，如 Microsoft Windows Media 等；另外新發表的 H.264 標準由於其效能的表現，也頗受業界矚目。不過大致來說除了 MOD 使用 STB 收看以外，目前各家採用不同技術的系統間大都以 Wintel 架構的 PC 做為終端設備，所採用的播放軟體在收視環境上對 PC 族群的消費者並無太大的使用門檻限制，不過在推廣上也因而有所侷限。

以網際網路市調公司的資料來看，雖然在民國 92 年 10 月「線上影音」網域類別的訪客約有 504 萬人，也就是大約 50% 的網友曾經造訪過線上影音網站。然而平均使用的時間最多不超過 14 分鐘，節目觀賞的型態與一般電視服務比較起來相距甚遠。營運上來看，雖然網際網路購物系統以及小額付費機制已經成熟，但是在國內廣播電視系統的媒體版圖上，目前仍未能發揮影響力。

伍、廣播電視數位視訊電視網路(Network)及數位頻道服務碼代碼管理(ID)統一管理規範及註冊機制

5.1、背景

二十一世紀是數位化的時代，「數位」一詞不僅是視訊產業發展的必然趨勢，更與現代人的生活息息相關。目前各國如美國、英國、日本、韓國等均積極推動數位電視的發展，國內無線數位電視亦於九十三年七月一日全面多頻道營運，對台灣數位電視產業而言，是一項極具意義且重大的里程碑，電視數位化之後，其所帶來的娛樂與資訊革命，將深深改變民眾的生活模式；未來民眾的生活也將走向多元化資訊與娛樂型態的數位家庭環境。

由於數位電視屬於新興產業，不論在技術應用、頻道規劃、內容服務等方面，仍處於探索的階段，包括國內民眾對數位電視的認知度與接受度普遍不足、市場尚未明朗化的情況等，使得不論是數位電視業者或是開發人員仍有許多努力的空間。

由於視訊壓縮技術的大幅進步，使得數位電視在同樣的 6MHz 頻寬中，可以傳送 3~4 個 SDTV 的數位化節目，因而節目數量大幅增加，使用者的選擇亦較多樣化。以往使用者所查閱的節目表(包括節目時間、節目名稱)，也因數位化由報紙進步為電子節目選單(EPG)；除了供使用者查閱方便外，可提供使用者直接選取並觀看節目，以及目前觀賞的節目資訊(例如球員的個人戰績、購物頻道商品的詳細資訊等)，另外頭端節目業者即使更動節目亦可即時更新電子節目選單等等，尚有許多的優點，在此不及備載。

network_information_section {}	
table_id	3
section_syntax_indicator	1
reserved_future_use	2
section_length	12
network_id	16
reserved	2
version_number	5
current_next_indicator	3
section_number	8
last_section_number	4
reserved_future_use	4
network_descriptor_length	12
for (i=1; i<=N; i++)	
descriptor()	
{	
reserved_future_use	4
transport_stream_id	16
for (i=1; i<=N; i++)	
transport_stream_id	16
original_network_id	16
reserved_future_use	4
transport_descriptor_length	12
for (j=1; j<=N; j++)	
descriptor()	
}	
CRC_32	32

圖 5-1、NIT 的資料結構圖

service_description_section {} {	
table_id	8
section_syntax_indicator	1
reserved_future_use	1
reserved	2
section_length	12
transport_stream_id	16
reserved	2
version_number	5
current_next_indicator	1
section_number	8
last_section_number	8
original_network_id	16
reserved_future_use	8
for (i=0;i<N;i++) {	
service_id	16
reserved_future_use	8
EIT_schedule_flag	1
EIT_present_following_flag	1
running_status	2
free_CA_mode	1
descriptor_loop_length	12
for (j=0;j<N;j++) {	
descriptor()	
}	
}	
crc_32	32

圖 5-2、SDT 的資料結構圖

5.2、電視網路與數位頻道服務代碼

在前一章節中大略提及有關服務資訊的應用與優點，而台灣有關服務資訊的規範是採用 Digital Video Broadcasting (DVB) ; Specification for Service Information (SI) in DVB systems，在規範 ETSI EN 300 468 v1.5.1 中說明 SI 包含一些頭端必要傳送的表格，有 NIT(table id=0x40)、SDT(table id=0x42)、EIT(table id=0x4E)、TDT(table id=0x70)。NIT (Network information table) 如圖 5-1 所示，主要記載網路名稱、服務型態及服務識別碼、傳送系統的資訊、以及可能接收的頻率列表等等；另外，也記錄了該廣播網路所播送之 transport stream 之 transport stream ID 與 original network ID。SDT (Service description table) 如圖 5-2 所示，主要記載關於某個 transport stream 內的某個服務，一些使用者導向的資訊，例如提供服務業者之名稱、服務名稱、可接收國家、條件式存取資訊等等 (service name、service ID、service status)。EIT (Event information table) 如圖 5-3 所示，主要記載一個服務內的某個節目 (event) 的資訊，包括節目名稱、節目型態、條件式存取識別碼、節目分類、節目分級等等。TDT (Time and date table) 如圖 5-4 所示，主要記載 UTC-time 和日期資訊，數位電視接收機可利用 TDT 所記載之時間修正內部自己的時間。

而原始電視網路代碼(original network id)與電視網路代碼(network id)都是 NIT(Network information table)表格中的一個欄位，表示網路識別碼，在 ETSI TR 101 162 中並規範各個廣播業者的網路識別碼或是使用範圍，由於 original_network_ids 值數量不多愈顯珍貴，因而這個網路識別碼是需要向 DVB 官方組織申請的，本計畫亦將向 DVB 組織提出申請並予以規範管理。

而數位頻道服務代碼(service id)是 SDT(Service description table)表格中的一個欄位，表示服務識別碼，此識別碼是不需要向 DVB 官方組織申請的，因其值可重複使用也不會與其他廣播業者所提供的頻道服務混淆，依照規範 ETSI EN 300 468 v1.5.1 中所描述的，每個頻道服務(service)是經由 original network id\ transport stream id\service id 依次識別的，只要原始電視網路代碼(original network id)可唯一識別即可；而國內無線廣播業者也針對這些代碼部分多次開會討論各家業者的使用範圍，其結果將在以下章節中說明。

```

event_information_section() {
    table_id                8
    section_syntax_indicator 1
    reserved_future_use     1
    reserved                2
    section_length          12
    service_id              16
    reserved                2
    version_number          3
    current_next_indicator  1
    section_number          8
    last_section_number     8
    transport_stream_id     16
    original_network_id     16
    segment_last_section_number 8
    last_table_id           8
    for (i=0; i<N;i++){
        event_id            16
        start_time          40
        duration             24
        running_status       3
        free_CA_mode         1
        descriptor_loop_length 12
        for (j=0; j<M;j++){
            descriptor()
        }
    }
    crc_32                  32
}

```

圖 5-3、EIT 的資料結構圖

```

Time_date_section() {
    table_id                8
    section_syntax_indicator 1
    reserved_future_use     1
    reserved                2
    section_length          12
    UTC_time                40
}

```

圖 5-4、TDT 的資料結構圖

5.3、現況

本計畫與交通部電信總局密切配合之下，於 6 月下旬由本計畫主持人電通所視訊與光通訊技術組系統技術部王鴻智經理在歐洲向 DVB 組織註冊“原始電視網路代碼(original network id)”，並與 DVB 組織的窗口 Mr. Peter MacAvock 見面進一步討論註冊細節，雙方針對此部份交換意見並獲得共識，取得台灣數位廣播內容與服務產業專用的 Original Network ID 與

Network IDs，其中原始電視網路代碼(original network id)為：0x209E，而預留台灣專用的電視網路代碼(network id)值為：0x3301-0x3400（詳如附件四）。

目前中華民國電視學會工研會已針對傳輸位元資料流 ID、網路代號、原始 Network ID、Service ID、.....提出一份提案，並送交委員會審議。

提案內容如下：

各電視台使用之傳輸位元資料流 ID，依照實體頻道號碼編列。

	CTV	PTS	FTV	TTV	CTS
Network ID 1	0024 (0x18)	0026 (0x1A)	0028 (0x1C)	0031 (0x1F)	0034 (0x22)
Network ID 2	0025 (0x19)	0027 (0x1B)	0029 (0x1D)	0032 (0x20)	0035 (0x23)
Country code	158	158	158	158	158
Language	chi	Chi	chi	chi	chi

各電視台使用之原始網路代號【原始 Network ID】Table。

	CTV	PTS	FTV	TTV	CTS
Original Network ID	0x0101 (257)	0x0102 (258)	0x0103 (259)	0x0104 (260)	0x0105 (261)

各電視台使用之傳輸位元資料流 ID，依照實體頻道號碼編列。

	CTV	PTS	FTV	TTV	CTS
Transport Stream ID 1	0x18	0x1A	0x1C	0x1F	0x22
Transport Stream ID 2	0x19	0x1B	0x1D	0x20	0x23

TS 使用之【Service ID】Table。

TS 1, TS 2	CTV	PTS	FTV	TTV	CTS
Service ID 1	100	200	300	400	500
Service ID 2	101	201	301	401	501
ServiceID 100	199	299	399	499	599

由於本計畫已向 DVB 組織註冊拿到台灣專用的原始電視網路代碼值(0x209E)及電視網路代碼值使用範圍(0x3301-0x3400)，中華民國電視學會工研會此份提案在這兩個代碼值部分應作相對應更改。

另外，在章節 5.2 中提及「每個頻道服務(service)是經由 original network id\transport stream id\service id 依次識別的，只要原始電視網路代碼(original network id)可唯一識別即可」，由於本計畫向 DVB 組織註冊拿到台灣專用的原始電視網路代碼值(0x209E)只有一個，所以建議無線廣播業者可在傳輸資料流代碼(transport stream id)做區分，這樣頻道服務(service)也不會與其他廣播業者所提供的頻道服務混淆；至於服務代碼(service id)可重複使用，在此無特定建議值，視各廣播業者喜好而定。

本計畫建議各電視台使用之原始電視網路代碼(original network id)：

	CTV	PTS	FTV	TTV	CTS
Original Network ID	0x209E	0x209E	0x209E	0x209E	0x209E

本計畫建議各電視台使用之電視網路代碼(network id)：

	CTV	PTS	FTV	TTV	CTS
Network ID	0x3301	0x3302	0x3303	0x3304	0x3305

本計畫建議各電視台使用之傳輸資料流代碼(transport stream id)(同電視學會工研會)：

	CTV	PTS	FTV	TTV	CTS
Transport Stream ID 1	0x18	0x1A	0x1C	0x1F	0x22
Transport Stream ID 2	0x19	0x1B	0x1D	0x20	0x23

在前文提及規範中的每個頻道服務(service)是經由 original network id\ transport stream id\service id 依次識別的，由於原始電視網路代碼值(original_network_ids)數量不多且珍貴，加上本計畫向 DVB 組織註冊拿到台灣專用的原始電視網路代碼值(0x209E)只有一個，因而建議各電視台使用特定的傳輸資料流代碼(transport stream id)在服務代碼(service id)可重複使用下，來區分不同的 service，service 的意義可看成是所謂的數位頻道，有別於現在的類比頻道，當電視數位化後，頻道數量可能是現在的好幾倍(3~4)，因而區分數位頻道是非常重要的事。茲舉例如下：

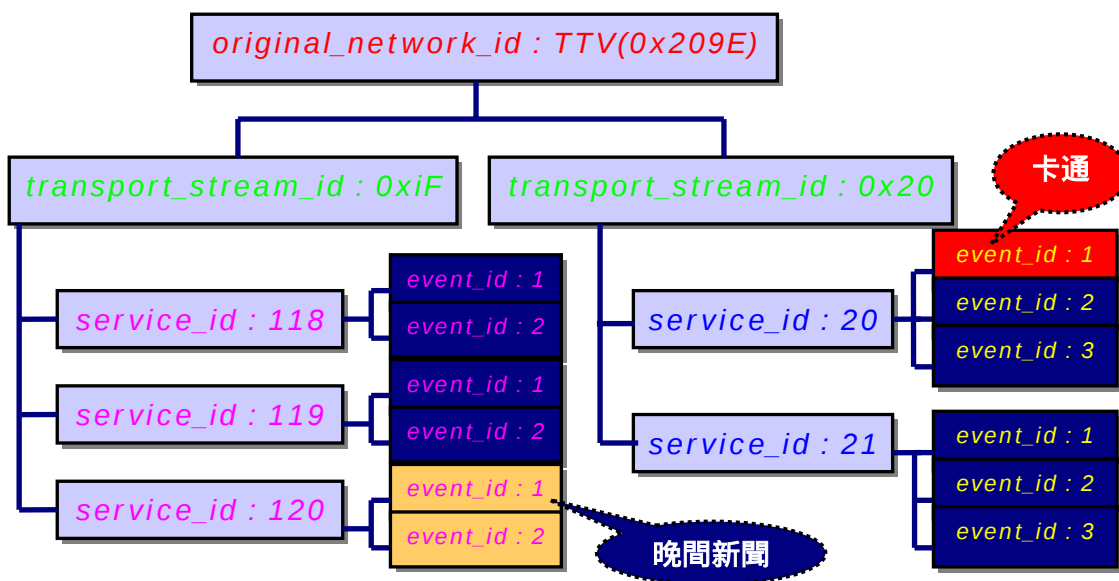


圖 5-5

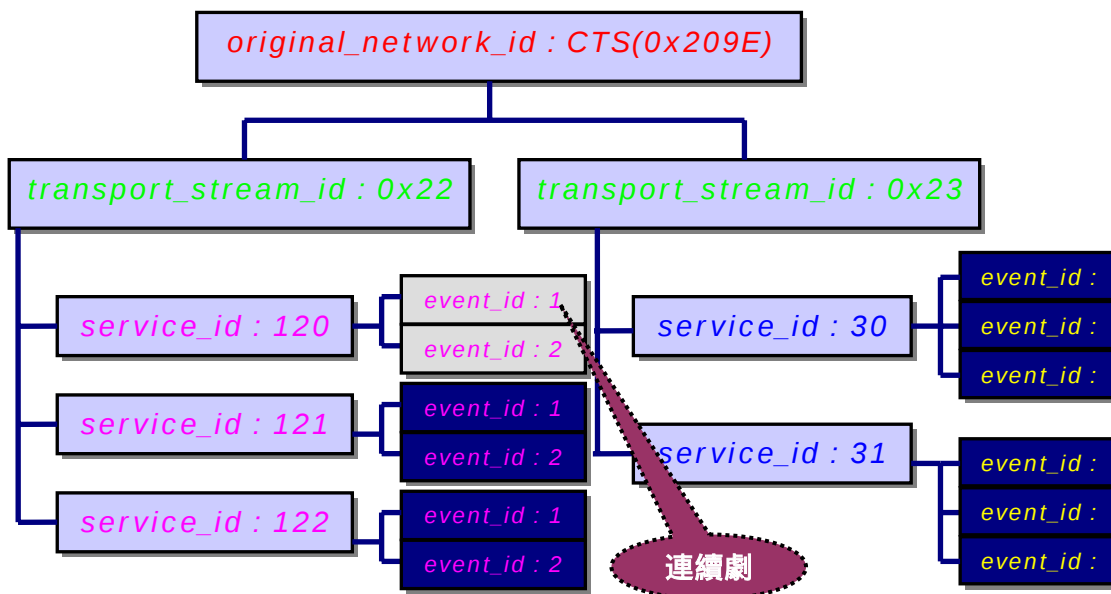


圖 5-6

圖 5-5 中 TTV 有兩個 physical channels,代碼分別為 0x1F 與 0x20,這兩個 physical channels 又個別有 3 個與 2 個 digital channels,代碼分別為 118~120 與 20~21(service id 可重複使用),而圖 5-6 中 CTS 有兩個 physical channels,代碼分別為 0x22 與 0x23,這兩個 physical channels 又個別有 3 個與 2 個 digital channels,代碼分別為 120~122 與 30~31(service id 可重複使用),在 TTV 中可見到數位頻道 0x209E\0x1F\120 正在撥放 " 晚間新聞 ",而在 CTS 中則可見到數位頻道 0x209E\0x22\120 正在撥放 " 連續劇 ",由於本計畫建議電視台使用特定的傳輸資料流代碼(transport stream id),使得 service id 在可重複使用的情況下,能夠識別不同的數位頻道。

由於電視廣播業者在使用不同的調變方式撥送節目內容,會造成同一頻寬(physical channel)會有不同的數位頻道數,因而本計畫並不建議電視台使用特定的服務代碼(service id),而採用與規範相同的想法,不去設限,讓其保有彈性。另外,本計畫雖未針對有線、衛星業者提出建議表格,但其實各 IDs 的管理方式相同,原始電視網路代碼值(original network id)為 0x209E,傳輸資料流代碼(transport stream id)可為避開無線業者的任意值,但建議使用特定可辨別的值(道理同無線業者),至於 service id 則可重複使用任意值。

陸、廣播電視業者之頭端編碼系統與條件式接取系統產生器間之統一介面格式與軟體協定建議

有關條件式接取/定址鎖碼對於終端設備的接收相容性問題，回顧技術的演進，以歐洲的 DVB 系統為例，首先被提出的是終端設備 Common Interface(CI)模組的規格(EN 50221 V1, 1997)，稍後再提出頭端系統整合之 DVB Simulcrypt 規格(TS 101 197 V1.2.1, 2002 / TS 103 197 V1.3.1, 2003)；同樣在美規系統以 OpenLabs 為例，也是先推動終端設備的 Point of Deployment (POD) 模組，然後再有頭端開放架構 OpenCAS 的發展。這種發展的過程，主要原因在於在數位電視系統發展初期，系統多半以專屬架構發展，不同頭端設備系統之間的整合困難，所以只能在終端設備端以安全模組介面來解決設備相容性的問題；但是隨著數位電視系統市場產品的多樣化，業界就會開始推動開放系統架構而提出規格標準。回顧國內定址鎖碼政策之研議，亦跟隨當時國外技術進展，由終端設備 Common Interface Module 之推動開始，以期解決終端設備解碼相容性問題。台灣數位電視系統發展至今，除了少數部署的數位有線電視系統外，數位地面無線電視系統尚未引進定址鎖碼系統；但是隨著無線電視台的數位電視開播，數位電視的收視將逐漸成長，各數位電視系統也將逐漸開始考慮定址鎖碼等作業以因應營運方式的調整。因此數位電視頭端系統的條件式接取/定址鎖碼系統介面，應考慮研議符合標準化，並搭配終端機 CI 模組之發展，以提供將來市場上的接收設備能接收各類系統的數位電視服務。

6.1 統一介面與軟體協定 – 同步鎖碼條件式接取(Simulcrypt)規範

由於我國各類數位電視系統已採用歐規DVB系統為主，是以頭端編碼系統與條件式接取系統之介面與軟體協定，亦建議要求符合DVB Simulcrypt規範為主。DVB Simulcrypt現有V1(TS 101 197)及V2(TS 103 197)兩種規格，兩者並沒有完整的回溯相容性，前後兩版規格並不適合混用。鑒於系統規劃完整性之因素，建議以TS 103 197 V1.3.1作為主要參考依據。(按：TS 103 197目前尚有V1.4.1草案在DVB組織內研議中)

目前市場主流之頭端設備及CA設備廠商，皆已有支援DVB Simulcrypt產品，並能提供客戶對現有系統設備進行升級更新以支援DVB Simulcrypt架構。

(1) DVB Simulcrypt系統架構

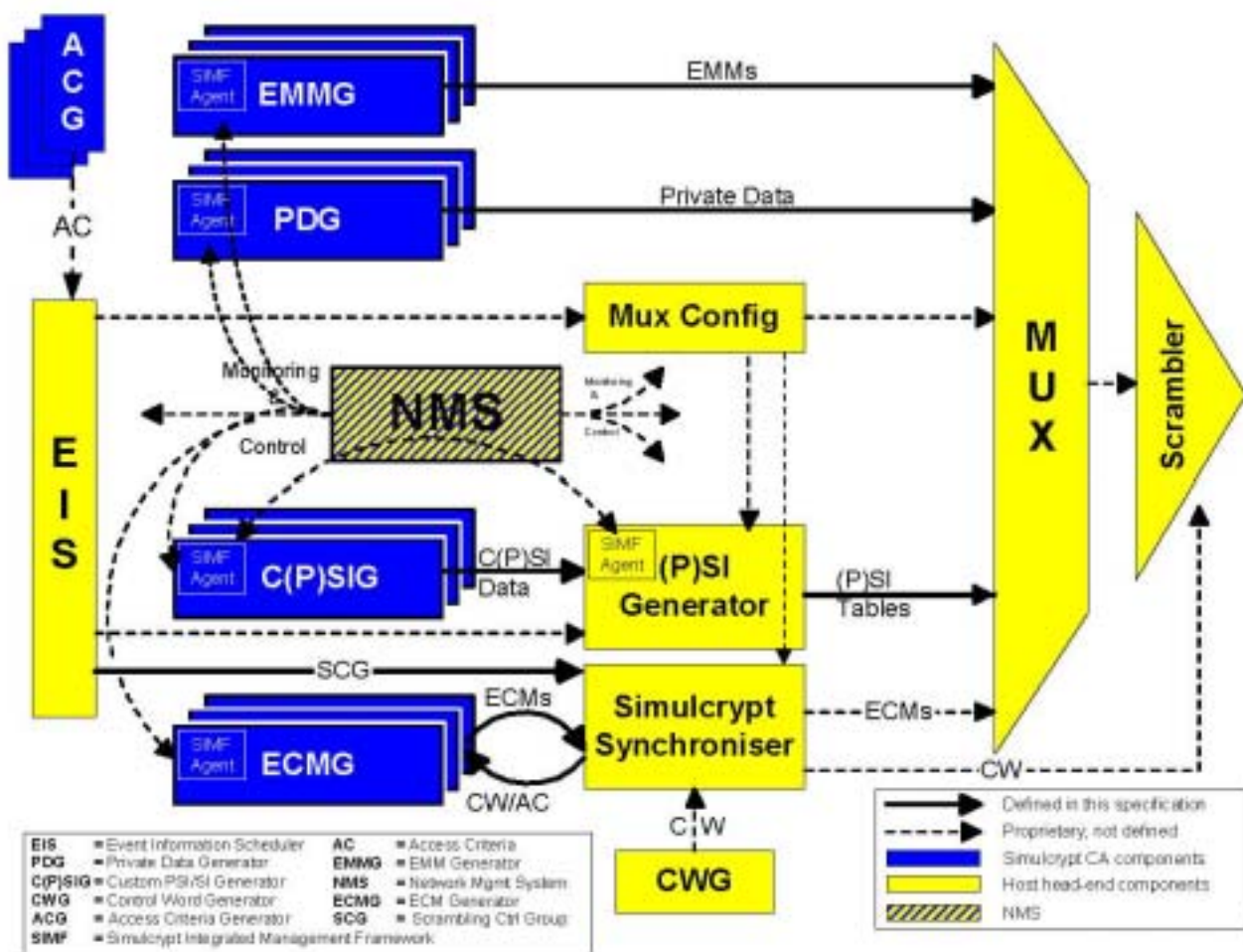


圖6-1 DVB-Simulcrypt 頭端系統架構 (ETSI TS 103 197 V1.3.1 [4])

系統架構圖中黃色部分為數位電視頭端系統之設備，藍色部分則為條件式接取系統之設備。這兩部分的分別主要是在定義該項設備的來源：黃色部分屬於頭端系統設備廠商 (Encoder/MUX) 所提供之設備種類，藍色部分則為CA廠商所提供之設備種類。

上述架構係為了配合現有各種條件式接取系統功能架構所作之完整規劃，在實作上則可依實際界接之系統增減非必要之設備種類（例如某些定址鎖碼系統即無PDG或C(P)SIG設備）。

(2) 系統設備定義

1. EIS – 節目排程系統 (Event Information Scheduler)

所有節目之內容、編碼系統設定、鎖碼條件等資料，以事件(event)之型態定義於EIS系統內，並按節目播出時刻與以排程後，EIS將依節目排程之時間，將相關資訊、設定、條件發送至各相關系統。

2. PDG – 私有資料產生器 (Private Data Generator)

各類非DVB定義之資料串流可透過PDG設備加入MUX的串流中。在某些條件式接取系統

中，安全資料係利用Private Data方式傳送至終端設備上。

3. (P)SIG – 系統資訊產生器 (PSI/SI Generator)

(P)SIG可產生DVB/MPEG2串流上必要之系統資訊(PSI/SI)。除了獨立的設備外，(P)SIG也常內建於MUX設備中。

4. C(P)SIG – 定製型系統資訊產生器 (Custom PSI/SI Generator)

在某些條件式接取系統中會利用修改之PSI/SI table(例如EIT)作為鎖碼條件之描述，故需利用C(P)SIG設備將相關資料插入(P)SIG設備中。

5. CWG – 控制碼產生器 (Control Word Generator)

CWG負責產生DVB Common Scrambling對訊號擾碼所需的控制碼(Control Word, CW)；所產生之CW一方面送到擾碼器(Scrambler)以供訊號擾碼，另一方面同時將該CW傳至SCS系統以分送各個不同的條件式接取系統予以加密以產生ECM。

6. ACG – 接取條件產生器 (Access Criteria Generator)

由於每一種條件式接取系統的接取條件(Access Criteria, AC)定義皆不相同，所以不同之條件式接取系統需要不同之ACG以便將接取條件之定義描述或是代號送至EIS系統作為鎖碼時的對應條件。

7. EMMG – EMM產生器 (EMM Generator)

EMMG根據客戶授權管理資料產生授權管理訊息(Entitlement Management Message, EMM)，以將授權資料傳送至終端設備上之智慧卡(Smart Card)。

8. ECMG – ECM產生器 (ECM Generator)

ECMG將由SCS傳來之CW以及AC加密包裝成為加密訊息(Entitlement Checking Message, ECM)，並與被該CW所擾碼之節目訊號同步送出。

9. SCS – Simulcrypt同步裝置(SimulCrypt Synchronizer)

由於各個條件式接取系統對同一段擾碼節目訊號需同步產生ECM，所以SCS負責接收來自CWG之CW以及由EIS所傳送之AC資料，並同時分送各不同條件式接取系統之ECMG，以便使不同條件式接取系統間之加密訊息能夠同步。

(3) 系統介面簡述

對於頭端設備與條件式接取系統之間的介面定義，根據TS 103 197 V1.3.1之設計，主要包括下列項目：

1. ECM產生器(ECMG) 與 Simulcrypt同步裝置(SCS)
2. EMM產生器(EMMG) 與 MUX
3. 網管系統(Network Management)
4. PSI/SI產生器((P)SIG) 與 定製型PSI/SI資料產生器(C(P)SIG)
5. PSI/SI產生器((P)SIG) 與 MUX
6. 節目排程系統(EIS) 與 Simulcrypt同步裝置(SCS)

上述系統介面之軟體協定規範，建議參照 TS 103 197 V1.3.1 定義。

柒、我國數位視訊共通平台建置整合之方案研究及建議

關於我國目前數位視訊平台建置之整合方案，我們認為必須以國內數位電視市場現況為研究前提，再試圖提出合理之解決方案。國內目前的數位電視平台可分為兩大類，第一類是由已大致完成整合之五家無線數位電視台所組成的平台，第二類是五大有線電視集團所自行擁有及維護之專屬平台，(國內目前除了原委會所委託之服務外，尚無直播衛星之衛星廣播數位電視台)。在此我們將探討兩種平台間整合的方式。第一種是以訂製 (customized) 的商業轉載整合方式，第二種則是以開放性市場平台為基礎的整合方式。至於數位電視平台間應採用哪種整合方式，應由各數位電視平台之營運商，依據自身之特性，考量每種方式之整合成本及複雜度，自行決定之。

在第一種整合方式裡，數位電視電台可以基於商業轉載的原則，以訂製之商業協商方式，轉載兩個原本並不相容的平台間之影音節目與互動內容。在兩個不相容的平台間轉載影音節目，主要涉及到 service information 與 conditional access 方式轉換的問題，難度並不高。但互動內容的部份，因為會與所採用之資料廣播與互動電視中介軟體標準相關，故轉換難度較高。可能互動內容之製作商，會需要為了商業轉載而為同一份互動內容，為不同的互動電視中介軟體標準各製作一個版本；或是製作所謂的可攜式互動內容 (portable content)，亦即一份互動內容可由兩種以上的不同標準之互動電視中介軟體執行。但此種整合方式的優點是不需要更換已經鋪設之數位電視接收機。英國 BBC 旗下的數位電視網路由三種不同的互動電視中介軟體標準 (MHP、MHEG-5 及 OpenTV) 所組成，近來 BBS 為此積極地研究關於可攜式互動內容之問題。

第二種整合方式是以開放式的市場為原則的整合方案。所謂開放式的市場，指的是消費者以零售方式購買，或是以租用等方式取得數位電視接收機，而消費者最終會擁有此數位電視接收機之產權的市場。在開放式的市場裡，為保障消費者權利，數位電視接收機必須有跨數位電視電台之交互操作性 (interoperability)。此交互操作性包括以下幾個層面：

- 數位電視傳輸系統: 依新聞局之規定，我國無線、有線及衛星數位電視系統，均將分別採用歐規之 DVB-T、DVB-C 以及 DVB-S 之標準。為保障開放式市場上消費者之權利，需確保數位電視接收機能在不同之數位電視傳輸系統上使用之能力。
- 條件接取系統: 如何確保數位電視接收機，能接受及處理由不同數位電視電台之各家頭端條件接取系統 (conditional access)，所傳送之條件接取相關訊息，以及解密經加密過之 transport stream 或 service。
- 互動應用平台: 如何確保數位電視接收機能處理由不同之數位電視電台所播送之互動應用服務。

我們在此提出在開放式市場 (open market) 上的數位視訊共通平台建置整合方案，並完成無線、有線及衛星各系統之“多媒體共通平台技術規範”建議規範草案。如果數位電視電台以封閉式專屬平台之方式營運，則不受上述三種技術規範之限制，但若數位電視電台進入前述定義之開放式市場，則應參考依循前述之工程規範。

總括來說，我們建議在國內的開放式市場上，各數位電視電台頭端之條件接取系統，建議使用 DVB SymulCrypt 界面，但不限制條件接取系統之廠牌。原因在於條件接取乃為數位電視最基本之影音節目之收費控管機制，是故規格必須統一。另外，為了使數位電視接收機能支援各家數位電視電台所採行之頭端條件接取系統，數位電視接收機上建議應納入符合 ECCA EuroLoader 標準之通用 loader 模組。因為數位電視接收機原本就一定會有 firmware 更新之 loader 機制存在，採用符合 ECCA EuroLoader 標準之 loader 模組，僅會增加數位電視接收機少量之額外成本。

關於開放式市場 (open market) 之數位視訊共通平台上所應採行之互動應用平台標準，我們僅建議各數位電視電台採用 MHP，但並非法規性之強制，採用時程也僅是建議，理由分析如下：

- 互動應用平台為選擇性額外之功能，可獨立於數位電視最基本之影音節目之外。
- 國內採行歐規 DVB 數位電視系統，因為多媒體共通平台之建置需採用開放式系統，目前應用於歐規 DVB 系統上之互動數位電視開放式標準，技術最成熟，最廣為使用的有兩個標準。第一個是 MHEG-5、另外一個則是 MHP。因為 MHEG-5 沒有 scripting 的功能，所以，在 DVB 制定 MHP 標準初期，就否決了以 MHEG-5 作為 MHP 核心之提案。而且，目前 MHP/GEM 已是歐規、美規、日規三個數位電視系統制定組織 DVB、ATSC/CableLabs 及 ARIB 官方之數位互動電視平台標準之共通核心，由此足見 MHP 標準之重要性。
- 因 MHP 標準定案時間較晚，數位電視接收機市場價格仍然偏高，在未經補貼之開放式零售市場上，市場穿透率均不高，如芬蘭、德國均是如此。但由於義大利無線數位電視開播，政府以補助方式強力導入 MHP，以推動 T-Government 服務，造成義大利 MHP 數位電視接收機市場之快速起飛。因義大利潛在的 MHP 數位電視接收機市場有 4500 萬部，此項事件刺激大量數位電視接收機製造商投入 MHP 數位電視接收機之研發，未來數年內可預見 MHP 數位電視接收機因激烈競爭而快速降價的情況發生。類似的降價經驗雖然之前已經在英國無線數位電視接收機的普及過程中發生過，但我們認為 MHP 數位電視接收機之普及狀況仍不易預測，故僅建議採用之時程。
- 從法規面上來看，日前歐盟委員會否決了在歐盟立法強制採用 MHP 之提案 (是一種弱性的否決，歐盟委員會還是贊同積極推廣 MHP，而且保留將來在數位電視消費性電子市場無法確保該有的交互操作性的時候，歐盟委員會仍有立法干預之權利)。另外，其他已採用 MHP/GEM 之國家，例如芬蘭、德國、義大利、南韓、美國，也未以法規層級強制 MHP 之使用，均是以業者所組織之協會名義，決定採用 MHP，並制定相關之工程規範。
- 行政院 NICI 小組建議 MHP 採用非強迫性之軟式導入。
- 是故，雖然 MHP 是建立開放式市場多媒體共通平台最合宜之開放式標準，但基於以上原因，我們在此僅建議國內數位電視電台採用 MHP，但暫時不建議以強制性之方式導入。

本章最後所附之無線、有線及衛星各系統之“多媒體共通平台技術規範”草案，若欲實現跨傳輸系統之開放式市場之整合，仍須無線、有線及衛星各數位電視台之學會就以下技術面之協調：

- 組態 (層次 0、位階 0):
 - 協調各數位電視台使用之 transport stream ID 及 service ID。
 - 協調各數位電視台所播送之 SI table、時間格式及 text field 編碼。

以開放式市場整合方式的優點是，影音節目及數位內容的轉載極為容易。但對採行封閉式專屬平台的數位電視電台而言，若已採用其他非 MHP 之互動數位電視中介軟體標準，則會面臨到已鋪設之數位電視接收機內中介軟體的轉移 (migration) 問題，對數位電視台來說會是一筆成本支出。

本計畫所訂定之相關數位電視多媒體共通平台技術規範(草案)內容與說明詳如表 7-1~表 7-3。

表 7-1 無線數位電視多媒體共通平臺技術規範（草案）及說明

第一章、總則
1.1 無線數位電視多媒體共通平臺技術規範 (以下簡稱本規範) 依電信法第四十六條第四項規定訂定之。
1.2 本規範適用對象為在規定頻道內發射數位電視信號供公眾接收之無線數位電視電臺。
1.3 本規範工程技術名詞定義如下： (1) 數位電視多媒體內容: 指數位電視電臺所播送數位電視節目，包括數位化影音、程序性數位內容、宣告性數位內容及其他數位化資料。 (2) 數位電視多媒體共通平臺: 指數位電視電臺播送數位電視多媒體內容與製播機制及數位電視接收機上所應提供之標準介面 (瀏覽器及中介軟體)。 (3) 程序性數位內容: 指以程式方式執行之數位內容，依據歐規 MHP 標準採用 Java 作為程序性數位內容之主體，稱為 DVB-J。 (4) 宣告性數位內容: 指以描述性標示 (descriptive markup) 方式呈現之數位內容，依據歐規 MHP 標準採用 XHTML 作為宣告性數位內容之主體，稱為 DVB-HTML。 (5) 近端互動: 指客戶端透過遙控器等人機介面及數位電視接收機上執行之多媒體內容互動，不涉及數位電視接收機上之回傳路徑 (return channel)。 (6) 遠端互動: 指客戶端透過人機介面於數位電視接收機上執行之多媒體內容，經由數位電視接收機回傳路徑，與遠端伺服器提供之服務互動。 (7) 網際網路 (Internet) 客戶端程式: 指 WWW 瀏覽器、郵件客戶端程式(SMTP client) 及新聞客戶端程式 (NNTP client)。
1.4 無線數位電視多媒體共通平臺之工程設備，除應符合本規範外，並得參採歐洲電信標準組織 (European Telecommunications Standards Institute, ETSI)及歐洲有線電纜通訊組織 (European Cable Communication Association) 所制定之下列數位電視相關規範： (1) MHP 1.0: ETSI TS 101 812 (2) MHP 1.1: ETSI TS 102 812 (3) DVB MPEG: ETSI TR 101 154 (4) DVB SI (Service Information): ETSI EN 300 468、ETSI TR 101 211、ETSI TR 101 162

- (5) DVB data broadcasting: ETSI TR 101 202、ETSI EN 301 192
- (6) DVB subtitling systems: ETSI EN 300 743
- (7) ECCA EuroLoader Specification
- (8) DVB SSU (System Software Update): ETSI TS 102 006
- (9) DVB measurement guideline for DVB systems: ETSI TR 101 290
- (10) DVB SimulCrypt: ETSI TS 101 197、ETSI TS 103 197、ETSI TR 102 035、ETSI ETR 289

第二章、無線數位電視多媒體共通平臺技術規範

2.1 本規範所訂定之無線數位電視多媒體共通平臺共有八種組態 (configuration)，各組態由四個層次 (profile) (層次 0、層次 1、層次 2、層次 3) 及兩個位階 (level) (位階 0、位階 1) 區分之 (參考如表 7-4 所示):

2.1.1 層次相同組態間之關係: 位階 1 之組態包含位階 0 組態定義之全部功能。

2.1.2 位階為 0 之四個組態間之關係：

- (1) 組態 (層次 1、位階 0) 包含組態 (層次 0、位階 0) 所定義之全部功能。
- (2) 組態 (層次 2、位階 0) 包含組態 (層次 1、位階 0) 與組態 (層次 0、位階 0) 定義之全部功能。
- (3) 組態 (層次 3、位階 0) 包含組態 (層次 2、位階 0)、組態 (層次 1、位階 0) 與組態 (層次 0、位階 0) 定義之全部功能。

2.1.3 組態定義之功能如下:

- (1) 組態 (層次 0、位階 0):
 - (A) 本組態定義之多媒體共通平臺母需 MHP 中介軟體，功能含影音串流播放及電子節目表單。
 - (B) 以 DVB 標準播送 MPEG-2 影音串流 (DVB MPEG: ETSI TR 101 154) 及服務資訊 (DVB SI: ETSI EN 300 468、ETSI TR 101 211、ETSI TR 101 162)。
 - (C) 應以中文方式顯示本臺七天內之電子節目表 (Electronic Program Guide, EPG)。
 - (D) 服務資訊之中文依據 DVB 資訊 DVB-SI 標準 (ETSI EN 300 468) 之規定編碼，以通用碼 (ISO/IEC 10646 內之 Unicode) 的編碼格式傳送。所使用字庫為通用碼內所包含之大五碼 (BIG5) 字庫。
- (2) 組態 (層次 0、位階 1):
 - (A) 本組態定義了組態 (層次 0、位階 0) 之選擇性功能，包括字幕及與資料廣播服務。字幕功能須依 DVB subtitling systems 標準 (ETSI EN 300 743)，提供中文字幕。資料廣播服務須依 DVB data broadcasting 標準提供之 (ETSI TR 101 202、ETSI EN 301 192)。
- (3) 組態 (層次 1、位階 0):
 - (A) 本組態使用 MHP 最基礎之多媒體共通平臺，定義數位電視多媒體內容近端互動功能。
 - (B) 支援 MHP 1.0 標準之增強型廣播層次 (enhanced broadcasting profile)，定義所有 Java API 及資料格式。
 - (C) 本組態僅支援程序性數位內容 DVB-J。
- (4) 組態 (層次 1、位階 1):
 - (A) 本組態定義以下列兩種選擇性功能，數位電視電臺得選擇播送數位電視多媒體

內容:

- (a) 選擇性功能一: 支援 MHP 1.1 標準定義智慧卡應用程式介面 (Smart Card API), 以提供具有近端互動能力之數位電視多媒體內容, 毋須與遠端伺服器互動之電子商務服務。
 - (b) 選擇性功能二: 支援 MHP-PDR (Personal Digital Recorder) 標準。
- (5) 組態 (層次 2、位階 0):
- (A) 本組態定義數位電視多媒體內容遠端互動之功能。
 - (B) 支援 MHP 1.0 標準之互動型廣播層次 (interactive broadcasting profile), 定義所有 Java API 及資料格式。
 - (C) 本組態僅支援程序性數位內容 DVB-J。
- (6) 組態 (層次 2、位階 1):
- (A) 本組態定義下列兩種選擇性功能, 數位電視電臺得選擇播送數位電視多媒體內容:
 - (a) 選擇性功能一: 支援 MHP 1.1 標準所定義之智慧卡應用程式介面 (Smart Card API), 以提供具有遠端互動能力之數位電視多媒體內容, 且須與遠端伺服器互動之電子商務服務。
 - (b) 選擇性功能二: 支援 MHP-PDR 標準。
- (7) 組態 (層次 3、位階 0):
- (A) 本組態定義數位電視多媒體內容與數位電視接收機上之網際網路客戶端程式互動之功能。
 - (B) 本組態應支援 MHP 1.1 標準之網際網路存取層次 (Internet access profile) 定義之所有 Java API 及資料格式。
 - (C) 本組態僅支援程序性數位內容 DVB-J。
- (8) 組態 (層次 3、位階 1):
- (A) 本組態定義下列兩種選擇性功能, 數位電視電臺得選擇播送數位電視多媒體內容:
 - (a) 選擇性功能一: 支援宣告性數位內容 DVB-HTML。
 - (b) 選擇性功能二: 支援 MHP-PDR 標準。

2.2 通用載入器 (common loader): 為進行系統軟體更新, 無線數位電視電臺應可提供 DVB-SSU 標準所定義之軟體更新服務:

2.2.1 服務定位 (signaling) 方式: 無線電視電台應至少提供 ETSI TS 102 006 所定義之簡要層次 (simple profile) 之服務定位方式。

2.2.2 服務傳送方式: 無線電視電台應至少提供 ETSI TS 102 006 所定義之標準資料巡迴播送方式 (standard data carousel)。

2.2.3 更新軟體遞交格式: 無線電視電台應至少能接受數位電視接收機製造商, 以 MPEG-2 傳輸位元資料流之格式遞交須更新之軟體。

2.3 條件接取系統 (conditional access, CA): 若無線數位電視電台欲將所播送之節目編碼加密, 則須採用採用歐規 DVB 同步加密 (SimulCrypt) 標準。

2.3.1 本規範並未指定無線數位電視電臺所應採用之條件接取系統, 僅制訂無線數位電視電台頭端設備整合條件接取系統所應採行之架構, 及頭端設備與條件接取系統間之介面規格。

2.3.2 以下頭端設備與條件式接取系統間之介面, 須符合 ETSI TS 103 197 之規定:

- (1) 授權檢查訊息產生器 (entitlement checking message generator) 與同步加密同步器 (SimulCrypt synchronizer) 間之介面。
- (2) 授權管理訊息產生器 (entitlement management message generator) 與多工器間之界面。
- (3) 服務資訊產生器 (PSI/SI generator) 與客製型服務資訊產生器 (custom PSI/SI generator) 間之界面。
- (4) 服務資訊產生器與多工器間之界面。
- (5) 節目排程系統 (event information scheduler) 與同步加密同步器間之界面。

2.4 支援其他之影音串流標準：本規範將依 MHP 相關之新標準定案後增訂之。

第三章、多媒體共通平臺技術規範實施時程

3.1 地面無線數位電視電臺依下列時程，播送符合本規範所訂定之數位電視多媒體內容：

3.1.1 民國 95 年 1 月 1 日起，開始播送符合組態 (層次 0、位階 0) 之數位電視多媒體內容。

3.1.2 民國 95 年 7 月 1 日起，開始播送符合組態 (層次 1、位階 0) 之數位電視多媒體內容。

3.1.3 民國 97 年 1 月 1 日起，開始播送符合組態 (層次 2、位階 0) 之數位電視多媒體內容。

3.1.4 民國 99 年 1 月 1 日起，開始播送符合組態 (層次 3、位階 0) 之數位電視多媒體內容。

3.2 本規範組態 (層次 0、位階 0) 須強制查驗，其餘各組態係建議採用之標準。

第四章、多媒體共通平台組態 (層次 0、位階 0) 工程技術規範

4.1 各無線數位電視電台使用之網路代號 (network ID)，依實體頻道號碼編列，以表 7-5 所示規範之。

4.2 各無線數位電視電台使用之原始網路代號 (original network ID)，依實體頻道號碼編列，以表 7-6 所示規範之。

4.3 無線數位電視電台使用之傳輸位元資料流代號 (transport stream ID) 以表 7-7 所示規範之。

4.4 無線數位電視電台傳送之服務資訊表 (SI table)，至少須含 NIT (actual network)、SDT (actual transport stream)、TDT 及 EIT。EIT 須遵循第 4.6 節之定義。TDT 須使用國際標準 UTC 時間。

4.5 服務資訊之文字欄位 (text field) 以通用碼格式編碼，文字欄位之起始碼 (leading code) 為 0X14。

4.6 電子節目表單須以 EIT 服務資訊表傳送之，至少須含 EIT (actual transport stream - present/following)、及 EIT (actual transport stream - schedule)。

4.7 無線數位電視電台播送之 MPEG-2 傳輸位元資料流，須通過 ETSI TR 101 290 所訂定之 DVB 系統第一優先測試項目 (參考如表 7-8 所示) 與 DVB 系統第二優先測試項目 (參考如表 7-9 所示)。

第五章、附則

5.1 本規範自公告日起實施。

無線數位電視多媒體共通平台技術規範說明

第一章、組態定義及功能之說明

- 1.1 基於“數位廣播電視系統整合之研究”案研究方法之定義，所提出之 MHP 系統服務規格標準草案，需依功能分級。
- 1.2 本技術規範之組態 (層次 0、位階 0) ~ 組態 (層次 3、位階 0) 定義分級之精神，主要參考了北歐斯堪地那維亞 (Scandinavia) 地區所採行之數位電視 NorDig specification，它包含了以下的四個 profile: NorDig Basic TV profile (無應用程式及互動功能)、NorDig Enhanced profile (MHP 1.1 Enhanced Broadcast profile)、NorDig Interactive profile (MHP 1.1 Interactive Broadcast profile) 以及 NorDig Internet Access profile (MHP 1.1 Internet Access profile)。四個 NorDig profile 間，有相同於本技術規範之組態 (層次 0、位階 0) ~ 組態 (層次 3、位階 0) 間之功能涵括關係。
- 1.3 有鑑於市場上支援 MHP 1.1 標準之 MHP 中介軟體產品並不普遍 (芬蘭及義大利目前的無線數位電台均採用 MHP 1.0.2 版)，故本技術規範組態 (層次 1、位階 0) 及組態 (層次 2、位階 0) 之功能，均改為以 MHP 1.0 版內所定義之 Enhanced Broadcast profile 與 Interactive Broadcast profile 為基礎。此外，再依據實際需求增加少數 MHP 1.1 標準定義之功能 (智慧卡應用程式介面)，及 DVB 對 MHP 1.0 標準之新增功能 (MHP-PDR 標準)，進入相對應之選擇性組態中。
- 1.4 本規範建議無線數位電視電台以類似之進程，依序逐步導入 MHP 之所有功能，以期降低無線數位電視電台導入 MHP 之困難：
 - (1) 先導入毋需 MHP 中介軟體之基礎功能，包影音串流播放 (MPEG-2 與 DVB-SI) 及電子節目表單。因字幕可直接內嵌於影像之中，資料廣播服務可由無線數位電視電台依其營運之商業模式決定是否採用，故兩者均列入選擇性組態 (層次 0、位階 1) 中。
 - (2) 因無線數位電視電台導入 MHP 1.0 Enhanced Broadcast profile，僅需增加某些頭端設備 (例如 MHP authoring tool 及 MHP object carousel generator)，以及將新增之頭端設備與原有之製播機制整合。導入此 profile 後之營運模式，與現有之影音節目播送之營運模式最為相近，故優先導入此 MHP 定義之 profile。
 - (3) 在無線數位電視電台熟悉 MHP 1.0 Enhanced Broadcast profile 之營運模式後，再行導入 MHP 1.0 Interactive Broadcast profile。無線數位電視電台需額外維護回傳網路及處理回傳網路回送訊息之伺服器。
 - (4) 在無線電視電台熟悉 MHP 1.0 Interactive Broadcast profile 之營運模式後，再行導入 MHP 1.1 Internet Access profile。無線數位電視電台需額外維護 Internet portal 等新增服務。
- 1.5 選擇性功能一 - 支援宣告性數位內容 DVB-HTML:
 - (1) DVB-HTML 是 MHP 1.1 標準中定義的選擇性功能，在 MHP 1.0 標準中並沒有定義。
 - (2) 為因應部分電視電台提出對宣告性數位內容的需求，本規範於組態 (層次 1、位階 1)、組態 (層次 2、位階 1) 與組態 (層次 3、位階 0)，定義 DVB-HTML 為選擇性功能。
- 1.6 選擇性功能二 - 智慧卡應用程式介面 (Smart Card API):

- (1) 智慧卡應用程式介面定義於 MHP 1.1 標準，故於組態 (層次 1、位階 0) 與組態 (層次 2、位階 0) 中沒有支援。為因應電子商務之需求，本規範於組態 (層次 1、位階 1) 與組態 (層次 2、位階 1) 中，定義智慧卡應用程式介面為選擇性功能。義大利無線數位電視電台即採行此種方式，擴充 MHP 1.0.2 版以支援智慧卡之使用。
- (2) 於組態 (層次 1、位階 1) 中，具有近端互動能力之數位電視多媒體內容，藉由智慧卡應用程式介面，以提供毋需與遠端伺服器互動之電子商務服務。
- (3) 於組態 (層次 2、位階 1) 中，具有遠端互動能力之數位電視多媒體內容，藉由智慧卡應用程式介面，以提供需與遠端伺服器互動之電子商務服務。

1.7 選擇性功能三 - MHP-PDR (Personal Digital Recorder) 標準:

- (1) DVB-TAM 即將於 2005 年 1 月，正式發表 MHP-PDR 標準，作為 MHP 1.0.3 標準之擴充功能。
- (2) 由於目前市場上主流之 PVR 產品，均僅具影音錄放功能，無法錄放影音之外的互動內容，故此標準定義了 MHP 應用程式與僅具影音錄放功能之 PVR (Personal Video Recorder) 互動之能力。
- (3) 有鑑於 PVR 將來極可能具備錄放互動內容之能力，本規範相關之條文須依將來 MHP 相關標準之擴充變更之。

1.8 關於 MHP 數位電視接收機應採行之組態之建議:

- (1) 欲製作具吸引力之 MHP 應用程式，例如: T-Government、T-Commerce、電視民意調查...等，MHP 數位電視接收機須支援回傳網路。所以，建議國內 MHP 無線數位電視接收機之標準，至少須包含組態 (層次 2、位階 0) 內所定義之功能，並考慮是否包含組態 (層次 2、位階 1) 內所定義之智慧卡應用程式介面。
- (2) 由於 MHP 1.1 標準尚未商品化，因此，若將來國內 MHP 數位電視接收機，選擇 MHP 1.0 作為中界軟體之標準，無線數位電視電台可不需導入組態 (層次 3、位階 0) 及組態 (層次 3、位階 1) 內所定義之功能。
- (3) 芬蘭與義大利之 MHP 無線數位電視接收機之規格，均是基於 MHP 1.0.2 版所延伸制定的。

第二章、多媒體共通平台組態 (層次 0、位階 0) 工程技術規範說明

2.1 本規範第四章內所訂定之網路代號 (network ID)、原始網路代號 (original network ID)、傳輸位元資料流代號 (transport stream ID) 以及服務資訊表 (SI table) 之相關規定，係參考中華民國電視學會工研會之“無線電視台數位頻道 DVB-T 之 SI Table 建議表”訂定之。

2.2 五家無線數位電視電台使用的原始網路代號及網路代號，均為 DVB 分配給台灣無線數位電視電台 (Taiwanese Digital Terrestrial Television) 使用的，登記的組織為交通部電信總局 (Directorate General of Telecommunications)。

2.3 服務資訊 DVB-SI 之規範補充說明:

- (1) 台灣的 country_code (ISO 3166) 為 158，可能被使用在以下的 descriptor 內:
 - country availability descriptor
 - local time offset descriptor
 - parental rating descriptor
- (2) 中文的 24-bit ISO_639_country_code (ISO 639-2) 為 ‘chi’，可能被使用在以下的 descriptor 內:

- component descriptor
- data broadcast descriptor
- extended event descriptor
- multilingual bouquet name descriptor
- multilingual component descriptor
- multilingual network name descriptor
- multilingual service name descriptor
- short event descriptor
- subtitling descriptor
- teletext descriptor
- VBI teletext descriptor

(2) DVB-SI 文字欄位中文的編碼為通用碼內之大五碼字庫 (Big-5)，此方案為工研院電通所提出 (leading code 0X14)，並已由 DVB 組織納入 DVB-SI 標準 (ETSI EN 300 468) 內。通用碼內之大五碼字庫之總字數為 13051 字。另外，大五碼 (Big-5) 之總字數為 13053 字，扣除兩個重複字編碼，和通用碼內之大五碼字庫之字數相同。但因兩種中文編碼方式之間的對應關係並非簡單的移位 (shifting) 關係，兩種編碼間的轉換必須使用查表法。

2.4 電子節目表單之規定係依 ETSI EN 300 468 訂定之：

(1) 暫不要求無線數位電視電台電子節目表單之跨載 (cross carriage)：

(A) 傳輸位元資料流承載 EIT_{schedule, other} 表原本即為 DVB-SI 標準內之選擇性功能。

(B) 其他採用歐規系統之國家，由數位電視電台自行擁有之傳輸位元資料流間，通常並不提供 EPG 跨載；EPG 跨載 (cross carriage) 通常僅由網路營運商 (network operator) 提供，例如：

(a) 芬蘭 Digita：為網路營運商，出租頻寬給數位電視電台提供無線數位電視服務。其負責在所有的傳輸位元資料流上，跨載 Digita 所有客戶提供之節目之 EPG 資訊。

(b) 義大利無線數位電視台：建議於國家級之網路營運商，提供 EPG 之跨載服務。

(C) 因國內各無線數位電視電台目前可用之頻寬仍不充裕，故建議目前各無線數位電視台僅承載本台之 EPG 資訊。

(2) 由於 NIT、SDT、TDT、EIT present/following、EIT schedule 之重複週期，與各傳輸位元資料流內可用之頻寬相關，故在本規範內暫時不訂定之，由各無線電視電台自行決定。

2.5 本規範組態 (層次 0、位階 0) 與數位電視接收機標準相容性問題：

(1) 經濟部數位電視機之標準草案是一個以歐規 DVB 標準為基礎之 tool box 標準集合。標準內分別訂定數位電視機之 channel coding (transmission、format) 與 source coding (video、audio、interactive content) 所應遵循之標準。本規範組態 (層次 0、位階 0) 與經濟部數位電視機標準草案相容。

(2) 與數位電視委員會之“台灣地面數位電視接收機技術需求”第 4.0 版相容：上述“台灣地面數位電視接收機技術需求”，所制定的是無電子節目表單與中介軟體之數位電視接收機標準。本規範組態 (層次 0、位階 0) 與數位電視委員會之“台灣地面數位電視接收機技術需求”相容。

第三章、其他工程技術規範說明

3.1 通用載入器:

- (3) 由於 DVB-SSU 標準在施行時有相當大的功能選擇彈性，故本規範 2.2 節定義了無線電視電台在導入 DVB-SSU 服務時，所應提供之功能的最小集合。
- (4) 此功能最小集合與台灣數位電視委員會之“台灣地面數位電視接收機技術需求 - 技術規範”內所定義之“數位接收機標準載入器以及軟體更新規範”彼此相容。
- (5) 與 DVB-SSU 相關規範制定時，參考了以下國外無線數位電視電台之工程規範：芬蘭 Digita、荷蘭 Digitenne 與義大利無線數位電視標準。

第四章、實施時程說明

4.1 基於以下原因，自民國 95 年 1 月 1 日起，開始播送符合組態 (層次 0、位階 0) 之數位電視多媒體內容:

- (1) 自 93 年 5 月起，國內五家無線電視台開始試播 MHP，預計 93 年 12 月前完成 MHP 測試。於測試完成後，再加上一年的頭端設備調整期相當合宜。
- (2) 工研院電通所，與中視等無線電視台，於 93 年 7 月起，開始 DVB-SI 表格的測試，預計 93 年 12 月完成。
- (3) DVB-SI 表格之測試完成，會繼續進行電子節目表單 (EPG) 與字幕 (subtitle) 的測試，預計 94 年 12 月前完成。

4.2 基於以下原因，自民國 95 年 7 月 1 日起，開始播送符合組態 (層次 1、位階 0) 之數位電視多媒體內容:

- (1) 經濟部 93 年 1 月 2 日向 NICT 提出之“台灣數位電視導入政策規劃”，軟式導入建議於 95 年 7 月 1 日開播 MHP，以提供增強型互動服務，吸引收視戶。
- (2) 電視學會建議國內無線電視台於 95 年 7 月 1 日開播 MHP。

4.3 基於以下原因，自民國 97 年 1 月 1 日起，開始播送符合組態 (層次 2、位階 0) 之數位電視多媒體內容:

- (1) 配合西元 2008 年奧運提供互動型廣播服務，以吸引消費者觀賞數位電視。
- (2) MHP 增強型廣播層次的機頂盒，廠商需一年左右時間，修改為 MHP 互動型廣播層次的機頂盒；再加上通路及後勤支援等改變所需之時間，由組態 (層次 1、位階 0)，過渡至組態 (層次 2、位階 0) 的時間，應以一年半為宜。

4.4 基於以下原因，自民國 99 年 1 月 1 日起，開始播送符合組態 (層次 3、位階 0) 之數位電視多媒體內容:

- (1) 由組態 (層次 2、位階 0) 低頻寬的回傳路徑，改變為組態 (層次 3、位階 0) 高頻寬的回傳路徑，終端設備業者需要更長之時間改變產品設計、生產線及市場營運方式，故間隔應以兩年為宜。

表 7-2 有線數位電視多媒體共通平臺技術規範 (草案)

第一章、總則

- 1.1 有線數位電視多媒體共通平臺技術規範 (以下簡稱本規範) 依電信法第四十六條第四項規定訂定之。
- 1.2 本開放性有線數位電視系統規範之適用對象為採行開放式市場營運模式之有線數位電視系統業者。開放式市場係指消費者以零售方式購買開放性系統之數位電視接收機之市場營運模式。對於採行專屬平台 (proprietary) 營運模式之有線數位電視系統業者，不在本規範之適用對象之內。
- 1.3 本規範僅軟式建議有線數位電視業者，於開放性系統之多媒體共通平臺上，採用歐規 MHP 作為互動數位電視中介軟體之標準。但本規範公告前，須依當時市場之實際情況以及開放性有線數位電視業者間之共識，考量與調整本條文之適用性。
- 1.4 本規範工程技術名詞定義如下：
- (1) 數位電視多媒體內容：指數位電視電臺所播送數位電視節目，包括數位化影音、程序性數位內容、宣告性數位內容及其他數位化資料。
 - (2) 數位電視多媒體共通平臺：指數位電視電臺播送數位電視多媒體內容與製播機制及數位電視接收機上所應提供之標準介面 (瀏覽器及中介軟體)。
 - (3) 程序性數位內容：指以程式方式執行之數位內容，依據歐規 MHP 標準採用 Java 作為程序性數位內容之主體，稱為 DVB-J。
 - (4) 宣告性數位內容：指以描述性標示 (descriptive markup) 方式呈現之數位內容，依據歐規 MHP 標準採用 XHTML 作為宣告性數位內容之主體，稱為 DVB-HTML。
 - (5) 近端互動：指客戶端透過遙控器等人機介面及數位電視接收機上執行之多媒體內容互動，不涉及數位電視接收機上之回傳路徑 (return channel)。
 - (6) 遠端互動：指客戶端透過人機介面於數位電視接收機上執行之多媒體內容，經由數位電視接收機回傳路徑，與遠端伺服器提供之服務互動。
 - (7) 網際網路 (Internet) 客戶端程式：指 WWW 瀏覽器、郵件客戶端程式 (SMTP client) 及新聞客戶端程式 (NNTP client)。
- 1.5 有線數位電視多媒體共通平臺之工程設備，除應符合本規範外，並得參採歐洲電信標準組織 (European Telecommunications Standards Institute, ETSI) 及歐洲有線電纜通訊組織 (European Cable Communication Association) 所制定之下列數位電視相關規範：
- (1) MHP 1.0: ETSI TS 101 812
 - (2) MHP 1.1: ETSI TS 102 812
 - (3) DVB MPEG: ETSI TR 101 154
 - (4) DVB SI (Service Information): ETSI EN 300 468、ETSI TR 101 211、ETSI TR 101 162
 - (5) DVB data broadcasting: ETSI TR 101 202、ETSI EN 301 192
 - (6) DVB subtitling systems: ETSI EN 300 743
 - (7) ECCA EuroLoader Specification
 - (8) DVB SSU (System Software Update): ETSI TS 102 006
 - (9) DVB measurement guideline for DVB systems: ETSI TR 101 290
 - (10) DVB SimulCrypt: ETSI TS 101 197、ETSI TS 103 197、ETSI TR 102 035、ETSI ETR 289
 - (11) DVB CI (common interface): ETSI EN 300 175

第二章、有線數位電視多媒體共通平臺技術規範

- 2.1 本規範所訂定之有線數位電視多媒體共通平臺共有八種建議組態 (configuration)，各組態由四個層次 (profile) (層次 0、層次 1、層次 2、層次 3) 及兩個位階 (level) (位階 0、位階 1) 區分之 (參考如表 7-4 所示)：
- 2.1.1 層次相同組態間之關係：位階 1 之組態包含位階 0 組態定義之全部功能。

2.1.2 位階為 0 之四個組態間之關係：

- (1) 組態 (層次 1、位階 0) 包含組態 (層次 0、位階 0) 所定義之全部功能。
- (2) 組態 (層次 2、位階 0) 包含組態 (層次 1、位階 0) 與組態 (層次 0、位階 0) 定義之全部功能。
- (3) 組態 (層次 3、位階 0) 包含組態 (層次 2、位階 0)、組態 (層次 1、位階 0) 與組態 (層次 0、位階 0) 定義之全部功能。

2.1.3 組態定義之功能如下：

(5) 組態 (層次 0、位階 0)：

- (A) 本組態定義之多媒體共通平臺毋需 MHP 中介軟體，功能含影音串流播放及電子節目表單。
- (B) 以 DVB-C 標準播放 MPEG-2 影音串流及服務資訊 (service information)。
- (C) 應以中文方式顯示本臺之電子節目表 (Electronic Program Guide, EPG)。
- (D) 服務資訊之中文依據 DVB 資訊 DVB-SI 標準之規定編碼。建議以通用碼 (Unicode) 之編碼格式傳送，所使用字庫為通用碼內所包含之大五碼 (BIG5) 字庫。

(6) 組態 (層次 0、位階 1)：

- (A) 本組態定義了組態 (層次 0、位階 0) 之選擇性功能，包括字幕及與資料廣播服務。字幕功能須依 DVB subtitling systems 標準 (ETSI EN 300 743)，提供中文字幕。資料廣播服務須依 DVB data broadcasting 標準提供之 (ETSI TR 101 202、ETSI EN 301 192)。

(3) 組態 (層次 1、位階 0)：

- (A) 本組態使用 MHP 最基礎之多媒體共通平臺，定義數位電視多媒體內容近端互動功能。
- (B) 支援 MHP 1.0 標準之增強型廣播層次 (enhanced broadcasting profile)，定義所有 Java API 及資料格式。
- (C) 本組態僅支援程序性數位內容 DVB-J。

(4) 組態 (層次 1、位階 1)：

- (A) 本組態定義以下列兩種選擇性功能，數位電視電臺得選擇播送數位電視多媒體內容：
 - (a) 選擇性功能一：支援 MHP 1.1 標準定義智慧卡應用程式介面 (Smart Card API)，以提供具有近端互動能力之數位電視多媒體內容，毋需與遠端伺服器互動之電子商務服務。
 - (b) 選擇性功能二：支援 MHP-PDR (Personal Digital Recorder) 標準。

(5) 組態 (層次 2、位階 0)：

- (A) 本組態定義數位電視多媒體內容遠端互動之功能。
- (B) 支援 MHP 1.0 標準之互動型廣播層次 (interactive broadcasting profile)，定義所有 Java API 及資料格式。
- (C) 本組態僅支援程序性數位內容 DVB-J。

(6) 組態 (層次 2、位階 1)：

- (A) 本組態定義以下列兩種選擇性功能，數位電視電臺得選擇播送數位電視多媒體內容：
 - (a) 選擇性功能一：支援 MHP 1.1 標準所定義之智慧卡應用程式介面 (Smart Card API)，以提供具有遠端互動能力之數位電視多媒體內容，且需與遠端伺服器互動之電子商務服務。

- (b) 選擇性功能二: 支援 MHP-PDR 標準。
 - (7) 組態 (層次 3、位階 0):
 - (A) 本組態定義數位電視多媒體內容與數位電視接收機上之網際網路客戶端程式互動之功能。
 - (B) 本組態應支援 MHP 1.1 標準之網際網路存取層次 (Internet access profile) 定義之所有 Java API 及資料格式。
 - (C) 本組態僅支援程序性數位內容 DVB-J。
 - (8) 組態 (層次 3、位階 1):
 - (A) 本組態定義下列兩種選擇性功能，數位電視電臺得選擇播送數位電視多媒體內容:
 - (a) 選擇性功能一: 支援宣告性數位內容 DVB-HTML。
 - (b) 選擇性功能二: 支援 MHP-PDR 標準。
- 2.2 通用載入器 (common loader): 為進行系統軟體更新，有線數位電視電臺應可提供 DVB-SSU 標準所定義之軟體更新服務:
- 2.2.1 服務定位 (signaling) 方式: 上述有線電視電台應至少提供 ETSI TS 102 006 所定義之簡要層次 (simple profile) 之服務定位方式。
 - 2.2.2 服務傳送方式: 上述有線電視電台應至少提供 ETSI TS 102 006 所定義之標準資料巡迴播送方式 (standard data carousel)。
 - 2.2.3 更新軟體遞交格式: 上述有線電視電台應至少能接受數位電視接收機製造商，以 MPEG-2 傳輸位元資料流之格式遞交須更新之軟體。
- 2.3 條件接取系統 (conditional access, CA): 若有線數位電視系統業者欲將所播送之節目編碼加密，則建議採用歐規 DVB 同步加密 (SimulCrypt) 標準或於數位電視接收機提供多重解密 (MultiCrypt) 之功能。
- 2.4 同步加密 (SimulCrypt):
- 2.4.1 同步加密並未指定有線數位電視系統業者所應採用之條件接取系統，僅制訂有線數位電視系統業者頭端設備整合條件接取系統所應採行之架構，及頭端設備與條件接取系統間之介面規格。
 - 2.4.2 若有線數位電視系統業者欲採用同步加密，則以下頭端設備與條件式接取系統間之介面，須符合 ETSI TS 103 197 之規定:
 - (1) 授權檢查訊息產生器 (entitlement checking message generator) 與同步加密同步器 (SimulCrypt synchronizer) 間之介面。
 - (2) 授權管理訊息產生器 (entitlement management message generator) 與多工器間之介面。
 - (3) 服務資訊產生器 (PSI/SI generator) 與客製型服務資訊產生器 (custom PSI/SI generator) 間之介面。
 - (4) 服務資訊產生器與多工器間之介面。
 - (5) 節目排程系統 (event information scheduler) 與同步加密同步器間之介面。
- 2.5 多重解密 (MultiCrypt): 若有線數位電視系統業者欲採用多重解密之功能，可選擇以下任一方式提供之:
- (1) 建立數位電視接收機條件式接系統開放模組與實體介面，遵循 DVB-CI 標準。

(2) 建立系統軟體通用載入器 (common loader) 規格，遵循 DVB-SSU 及 ECCA EuroLoader 標準。

2.6 支援其他之影音串流標準：本規範將依 DVB 與 MHP 相關之新標準定案後增訂之。

第三章、多媒體共通平台組態 (層次 0、位階 0) 工程技術規範

3.1 有線數位電視系統業者傳送之服務資訊表 (SI table)，須由各有線數位電視系統業者協商訂定之。

3.2 服務資訊之文字欄位 (text field) 編碼格式，須由各有線數位電視系統業者協商訂定之。

3.3 電子節目表單建議以 EIT 服務資訊表傳送之，含 EIT present/following 及 EIT schedule。

第四章、附則

4.1 本規範自公告日起實施。

4.2 有鑑於有線數位電視系統業者目前所營運之專屬網路之性質，與開放式市場之性質間有相當之差異，工研院電通所尊重有線數位電視系統業者，於未來進入開放式市場營運時，依當時市場之實際情況以及業者間之共識，定位此建議規範部分內文之適用性之意見。

表 7-3 衛星廣播數位電視多媒體共通平臺技術規範（草案）

第一章、總則

1.1 衛星廣播數位電視多媒體共通平臺技術規範（以下簡稱本規範）依電信法第四十六條第四項規定訂定之。

1.2 本規範適用對象為在地球上與衛星間做廣播電視節目信號發射之衛星廣播數位電視電臺。

1.3 本規範工程技術名詞定義如下：

- (1) 數位電視多媒體內容：指數位電視電臺所播送數位電視節目，包括數位化影音、程序性數位內容、宣告性數位內容及其他數位化資料。
- (2) 數位電視多媒體共通平臺：指數位電視電臺播送數位電視多媒體內容與製播機制及數位電視接收機上所應提供之標準介面（瀏覽器及中介軟體）。
- (3) 程序性數位內容：指以程式方式執行之數位內容，依據歐規 MHP 標準採用 Java 作為程序性數位內容之主體，稱為 DVB-J。
- (4) 宣告性數位內容：指以描述性標示 (descriptive markup) 方式呈現之數位內容，依據歐規 MHP 標準採用 XHTML 作為宣告性數位內容之主體，稱為 DVB-HTML。
- (5) 近端互動：指客戶端透過遙控器等人機介面及數位電視接收機上執行之多媒體內容互動，不涉及數位電視接收機上之回傳路徑 (return channel)。
- (6) 遠端互動：指客戶端透過人機介面於數位電視接收機上執行之多媒體內容，經由數位電視接收機回傳路徑，與遠端伺服器提供之服務互動。
- (7) 網際網路 (Internet) 客戶端程式：指 WWW 瀏覽器、郵件客戶端程式 (SMTP client) 及新聞客戶端程式 (NNTP client)。

1.4 衛星廣播數位電視多媒體共通平臺之工程設備，除應符合本規範外，並得參採歐洲電信標準組織 (European Telecommunications Standards Institute, ETSI)及歐洲有線電纜通訊組織 (European Cable Communication Association) 所制定之下列數位電視相關規範：

- (1) MHP 1.0: ETSI TS 101 812
- (2) MHP 1.1: ETSI TS 102 812
- (3) DVB MPEG: ETSI TR 101 154
- (4)DVB SI (Service Information): ETSI EN 300 468、ETSI TR 101 211、ETSI TR 101 162
- (5)DVB data broadcasting: ETSI TR 101 202、ETSI EN 301 192
- (6)DVB subtitling systems: ETSI EN 300 743
- (7)ECCA EuroLoader Specification
- (8)DVB SSU (System Software Update): ETSI TS 102 006
- (9)DVB measurement guideline for DVB systems: ETSI TR 101 290
- (10)DVB SimulCrypt: ETSI TS 101 197、ETSI TS 103 197、ETSI TR 102 035、ETSI ETR 289

第二章、衛星廣播數位電視多媒體共通平臺技術規範

2.1 本規範所訂定之衛星廣播數位電視多媒體共通平臺共有八種組態 (configuration)，各組態由四個層次 (profile) (層次 0、層次 1、層次 2、層次 3) 及兩個位階 (level) (位階 0、位階 1) 區分之 (參考如表 7-4 所示)：

2.1.1 層次相同組態間之關係：位階 1 之組態包含位階 0 組態定義之全部功能。

2.1.2 位階為 0 之四個組態間之關係：

- (1)組態 (層次 1、位階 0) 包含組態 (層次 0、位階 0) 所定義之全部功能。
- (2)組態 (層次 2、位階 0) 包含組態 (層次 1、位階 0) 與組態 (層次 0、位階 0) 定義之全部功能。
- (3)組態 (層次 3、位階 0) 包含組態 (層次 2、位階 0)、組態 (層次 1、位階 0) 與組態 (層次 0、位階 0) 定義之全部功能。

2.1.3 組態定義之功能如下：

(1)組態 (層次 0、位階 0)：

- (A) 本組態定義之多媒體共通平臺毋需 MHP 中介軟體，功能含影音串流播放及電子節目表單。
- (B) 以 DVB 標準播送 MPEG-2 影音串流 (DVB MPEG: ETSI TR 101 154)及服務資訊 (DVB SI: ETSI EN 300 468、ETSI TR 101 211、ETSI TR 101 162)。
- (C) 應以中文方式顯示本臺七天內之電子節目表 (Electronic Program Guide, EPG)。
- (D) 服務資訊之中文依據 DVB 資訊 DVB-SI 標準 (ETSI EN 300 468) 之規定編碼，以通用碼 (ISO/IEC 10646 內之 Unicode) 的編碼格式傳送。所使用字庫為通用碼內所包含之大五碼 (BIG5) 字庫。

(2)組態 (層次 0、位階 1)：

- (A) 本組態定義了組態 (層次 0、位階 0) 之選擇性功能，包括字幕及與資料廣播服務。字幕功能須依 DVB subtitling systems 標準 (ETSI EN 300 743)，提供中文字幕。資料廣播服務須依 DVB data broadcasting 標準提供之 (ETSI TR 101 202、ETSI EN 301 192)。

(3)組態 (層次 1、位階 0)：

- (A) 本組態使用 MHP 最基礎之多媒體共通平臺，定義數位電視多媒體內容近端互動功能。

- (B) 支援 MHP 1.0 標準之增強型廣播層次 (enhanced broadcasting profile), 定義所有 Java API 及資料格式。
 - (C) 本組態僅支援程序性數位內容 DVB-J。
 - (4)組態 (層次 1、位階 1):
 - (A) 本組態定義以下列兩種選擇性功能, 數位電視電臺得選擇播送數位電視多媒體內容:
 - (a) 選擇性功能一: 支援 MHP 1.1 標準定義智慧卡應用程式介面 (Smart Card API), 以提供具有近端互動能力之數位電視多媒體內容, 毋需與遠端伺服器互動之電子商務服務。
 - (b) 選擇性功能二: 支援 MHP-PDR (Personal Digital Recorder) 標準。
 - (5)組態 (層次 2、位階 0):
 - (A) 本組態定義數位電視多媒體內容遠端互動之功能。
 - (B) 支援 MHP 1.0 標準之互動型廣播層次 (interactive broadcasting profile), 定義所有 Java API 及資料格式。
 - (C) 本組態僅支援程序性數位內容 DVB-J。
 - (6)組態 (層次 2、位階 1):
 - (A) 本組態定義下列兩種選擇性功能, 數位電視電臺得選擇播送數位電視多媒體內容:
 - (a) 選擇性功能一: 支援 MHP 1.1 標準所定義之智慧卡應用程式介面 (Smart Card API), 以提供具有遠端互動能力之數位電視多媒體內容, 且需與遠端伺服器互動之電子商務服務。
 - (b) 選擇性功能二: 支援 MHP-PDR 標準。
 - (7)組態 (層次 3、位階 0):
 - (A) 本組態定義數位電視多媒體內容與數位電視接收機上之網際網路客戶端程式互動之功能。
 - (B) 本組態應支援 MHP 1.1 標準之網際網路存取層次 (Internet access profile) 定義之所有 Java API 及資料格式。
 - (C) 本組態僅支援程序性數位內容 DVB-J。
 - (8)組態 (層次 3、位階 1):
 - (A) 本組態定義下列兩種選擇性功能, 數位電視電臺得選擇播送數位電視多媒體內容:
 - (a) 選擇性功能一: 支援宣告性數位內容 DVB-HTML。
 - (b) 選擇性功能二: 支援 MHP-PDR 標準。
- 2.2 通用載入器 (common loader): 為進行系統軟體更新, 衛星廣播數位電視電臺應可提供 DVB-SSU 標準所定義之軟體更新服務:
- 2.2.1 服務定位 (signaling) 方式: 衛星廣播電視電台應至少提供 ETSI TS 102 006 所定義之簡要層次 (simple profile) 之服務定位方式。
 - 2.2.2 服務傳送方式: 衛星廣播電視電台應至少提供 ETSI TS 102 006 所定義之標準資料巡迴播送方式 (standard data carousel)。
 - 2.2.3 更新軟體遞交格式: 衛星廣播電視電台應至少能接受數位電視接收機製造商, 以 MPEG-2 傳輸位元資料流之格式遞交須更新之軟體。
- 2.3 條件接取系統 (conditional access, CA): 若衛星廣播數位電視電台欲將所播送之節目編碼

加密，則須採用採用歐規 DVB 同步加密 (SimulCrypt) 標準。

2.3.1 本規範並未指定衛星廣播數位電視電臺所應採用之條件接取系統，僅制訂衛星廣播數位電視電台頭端設備整合條件接取系統所應採行之架構，及頭端設備與條件接取系統間之介面規格。

2.3.2 以下頭端設備與條件式接取系統間之介面，須符合 ETSI TS 103 197 之規定：

- (1) 授權檢查訊息產生器 (entitlement checking message generator) 與同步加密同步器 (SimulCrypt synchronizer) 間之介面。
- (2) 授權管理訊息產生器 (entitlement management message generator) 與多工器間之介面。
- (3) 服務資訊產生器 (PSI/SI generator) 與客製型服務資訊產生器 (custom PSI/SI generator) 間之介面。
- (4) 服務資訊產生器與多工器間之介面。
- (5) 節目排程系統 (event information scheduler) 與同步加密同步器間之介面。

2.4 支援其他之影音串流標準：本規範將依 MHP 相關之新標準定案後增訂之。

第三章、多媒體共通平台組態 (層次 0、位階 0) 工程技術規範

3.1 衛星廣播數位電視電台傳送之服務資訊表 (SI table)，須由各衛星廣播數位電視電台協商訂定之。

3.2 服務資訊之文字欄位 (text field) 編碼格式，須由各衛星廣播數位電視電台協商訂定之。

3.3 電子節目表單須以 EIT 服務資訊表傳送之，含 EIT present/following、及 EIT schedule。

第四章、附則

4.1 本規範自公告日起實施。

上述規範所訂定之數位電視多媒體共通平台共有八種組態 (configuration)，各組態由四個層次 (profile) (層次 0、層次 1、層次 2、層次 3) 與兩個位階 (level) (位階 0、位階 1) 區分之 (參考表 7-4)。

表 7-4 多媒體共通平台組態間的關係

(層次 0、位階 1) ● 字幕 (中文) ● 資料廣播服務	(層次 1、位階 1) ● 智慧卡應用程式 介面 ● DVB-PDR 標準	(層次 2、位階 1) ● 智慧卡應用程式 介面 ● DVB-PDR 標準	(層次 3、位階 1) ● DVB-HTML ● DVB-PDR 標準
(層次 0、位階 0) ● MPEG-2 ● 服務資訊 (中文) ● 電子節目表單	(層次 1、位階 0) ● MHP 1.0 標準的增 強型廣播層次	(層次 2、位階 0) ● MHP 1.0 標準的互 動型廣播層次	(層次 3、位階 0) ● MHP 1.1 標準的網 際網路存取層次

表 7-5 無線數位電視電台採用之網路代號表 (network ID table)

	CTV	PTS	FTV	TTV	CTS
網路代號 1 (network ID 1)	0X3324 (13092)	0X3326 (13094)	0X3328 (13096)	0X3331 (13105)	0X3334 (13108)
網路代號 2 (network ID 2)	0X3325 (13093)	0X3327 (13095)	0X3329 (13097)	0X3332 (13106)	0X3335 (13109)

表 7-6 無線數位電視電台採用之原始網路代號表 (original network ID table)

	CTV	PTS	FTV	TTV	CTS
原始網路代號 (original network ID)	0X209E (8350)	0X209E (8350)	0X209E (8350)	0X209E (8350)	0X209E (8350)

表 7-7 無線數位電視電台採用之傳輸位元資料流代號表 (transport stream ID table)

	CTV	PTS	FTV	TTV	CTS
傳輸位元資料 流代號 1 (transport stream ID 1)	0X18 (0024)	0X1A (0026)	0x1C (0028)	0X1F (0031)	0X22 (0034)
傳輸位元資料 流代號 2 (transport stream ID 2)	0X19 (0025)	0X1B (0027)	0x1D (0029)	0X20 (0032)	0x23 (0035)

表 7-8 DVB 系統第一優先檢測項目

錯誤名稱	錯誤條件
TS_sync_loss	連續三個以上之傳輸位元資料流封包 (TS packet) 未偵測到同步位元
PAT_error	1. PID 0X0000 未至少每隔 0.5 秒出現 2. PID 0X0000 未含有 table_id 0X00 (即 PAT) 3. PID 0X0000 之 Srambling_control_field 不為 00
PAT_error_2	1. table_id 為 0X00 之 section，未至少每隔 0.5 秒出現於 PID 0X0000 上 2. table_id 不為 0X00 之 section，出現於 PID 0X0000 上 3. PID 0X0000 之 Srambling_control_field 不為 00
PMT_error	1. 於 PAT 所參用之 PID 上，table_id 為 0X02 (即 PMT) 之 section 未至少每隔 0.5 秒出現 2. 若某 PID 上有 table_id 為 0X02 (即 PMT) 之 section，該 PID 之 Srambling_control_field 不為 00
PMT_error_2	1. table_id 為 0X02 (即 PMT) 之 section，未至少每隔 0.5 秒出現於 PAT 參用之 program_map_PID 上 2. table_id 為 0X02 (即 PMT) 之 section，出現於 PAT 參用之 program_map_PID 上，其 Srambling_control_field 不為 0X00
PID_error	在使用者所規定之週期 (在本工程規範內定義為 1 秒) 內，被參用之 PID 未出現

表 7-9 DVB 系統第二優先檢測項目

錯誤名稱	錯誤條件
PCR_error	1. 出現超過 100 ms 之 PCR 不連續狀況，卻無標示 2. 兩個連續 PCR 之間隔時間超過 40ms
PCR_repetition_error	兩個連續 PCR 值之間隔時間超過 40ms
PCR_discontinuity_error	兩個連續 PCR 值之差 ($PCR_{i+1} - PCR_i$) 不在 0...100 ms 範圍內，卻無標示
PCR_accuracy_error	所選定節目之 PCR 精確度不在 ± 500 ns 之內
PTS_error	PTS 重複週期超過 700ms

參考文獻

1. “ Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.1.1,” ETSI TS 102 812 V1.2.1 (2003-06).
2. "Digital Video Broadcasting (DVB); DVB specification for data broadcasting", EN 301 192 v1.2.1
3. "Digital Video Broadcasting (DVB); Implementation guidelines for the use of MPEG-2 Systems, Video and Audio in satellite, cable and terrestrial broadcasting applications," TR 101 154 1.4.1.
4. "Digital Video Broadcasting (DVB); Guidelines on implementation and usage of Service Information (SI)", ETSI TR 101 211 v1.5.1
5. “ Information technology - Generic coding of moving pictures and associated audio information: Systems, ” ISO/IEC 13818-1 1996.
6. “ Information technology—Generic coding of moving pictures and associated audio information—Part 2: Video (MPEG-2 Video) , ” ISO/IEC 13818-2 1996.
7. "Digital Video Broadcasting (DVB); Guideline for the use of DVB specifications and standards," TR 101 200 1.1.1.
8. “ Digital broadcasting systems for television, sound and data services; Allocation of Service Information (SI) codes for Digital Video Broadcasting (DVB) systems, ” ETR 162.
9. “Harmonization of Procedural Content Formats for Interactive TV Applications,” ITU-T Recommendation J.202, April 2003.
10. “Digital Video Broadcasting (DVB); Globally Executable MHP (GEM) 1.0.1,” ETSI TS 102 819 V1.2.1, May 2004.
11. “Digital Video Broadcasting (DVB); Globally Executable MHP (GEM) 1.0.0,” ETSI TS 102 819 V1.1.1, Jan. 2003.
12. “Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.0.3,” ETSI TS 101 812 V1.3.1, June 2003.
13. “Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.0.2,” ETSI TS 101 812 V1.2.1, June 2002.

14. "OpenCable Application Platform Specification - OCAP 1.0 Profile," Cable Television Laboratories OC-SP-OCAP1.0-I11-040604, June 2004.
15. "OpenCable Application Platform Specification - OCAP 2.0 Profile," Cable Television Laboratories OC-SP-OCAP2.0-I01-020419, April 2002.
16. "ATSC Candidate Standard: Advanced Common Application Platform (ACAP)," Advanced Television Systems Committee, Feb. 2004.
17. 「數位有線電視之發展與商業模式」, 陳光毅 (民 92) 2003 數位電視及廣播國際研討會。
18. "Digital Video Broadcasting (DVB); Implementation Guidelines of the DVB Simulcrypt Standard", ETSI TR 102 035 V1.1.1 (2002-04).
19. "Digital Video Broadcasting (DVB); Head-end Implementation of SimulCrypt", ETSI TS 103 197 V1.3.1 (2003-01).
20. "Digital Video Broadcasting (DVB); Specification for System Software Update in DVB Systems," ETSI TS 102 006 V1.2.1 (2002-10).

附 件

(一) 技術規格摘錄

1. DVB-MHP 規格書中文摘錄(Reference:Digital Video Broadcasting (DVB);
Multimedia Home Platform (MHP) Specification 1.0.3) 附 1-1

(二) 出國報告

1. 2004 年新加坡亞洲廣播通訊展 , Broadcast Asia 2004 附 2-1
2. DVB 出國報告 附 2-19
3. 出國報告-中國 附 2-21

(三) 會議記錄

1. 台灣數位電視產業聯盟技術發展工作小組第一次座談會－探索數位電視技術的需求與發展方向 附 3-1
2. Network ID 註冊機制的討論－日本 Data Component ID 的做法 附 3-4
3. 台灣數位電視產業聯盟標準制定工作小組第一次座談會－台灣數位電視產業技術規範研討 附 3-6
4. 無線數位電視多媒體共通平台技術規範研討會 附 3-11
5. 「數位廣播電視系統整合之研究」座談會 附 3-12

(四) DVB-SSU 與 EuroLoader 附 4-1

(五) 其他

1. DVB's email for Original_Network_Ids and Network_Ids in Taiwan 附 5-1

附件（一） 技術規格摘錄

1. DVB-MHP 規格書中文摘錄(Reference:Digital Video Broadcasting (DVB);
Multimedia Home Platform (MHP) Specification 1.0.3)

附 1-1

1 基本架構

1.1 整體環境

從最單純的角度來看，多媒體家庭平台（Multimedia Home Platform，簡稱 MHP）的整體環境如下（圖 1-1）。MHP 的軟體可以存取串流（stream）和資料，也可以將資料寫入儲存。MHP 平台也可以將串流與資料導出至一個資料槽（sink）或是貯藏庫（store）。

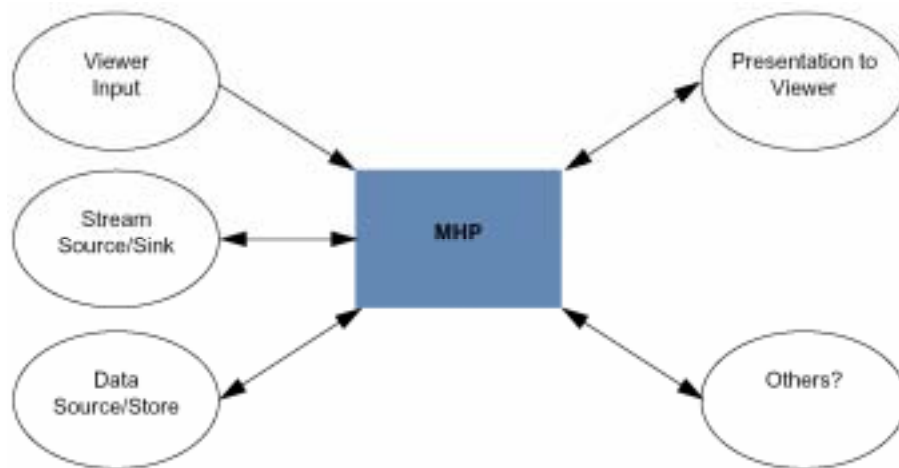


圖 1-1：MHP 整體環境

此平台會從觀賞者（Viewer）輸入裝置收到資料輸入，然後由螢幕或是其他輸出設備（例如喇叭），呈現給觀賞者。此平台也有管道與遠端個體（remote entities）進行通訊。

圖 1-2 顯示一些 MHP 可能與外界接觸的介面。雖然這只是一個範例，卻已經點出一些重要的精神。

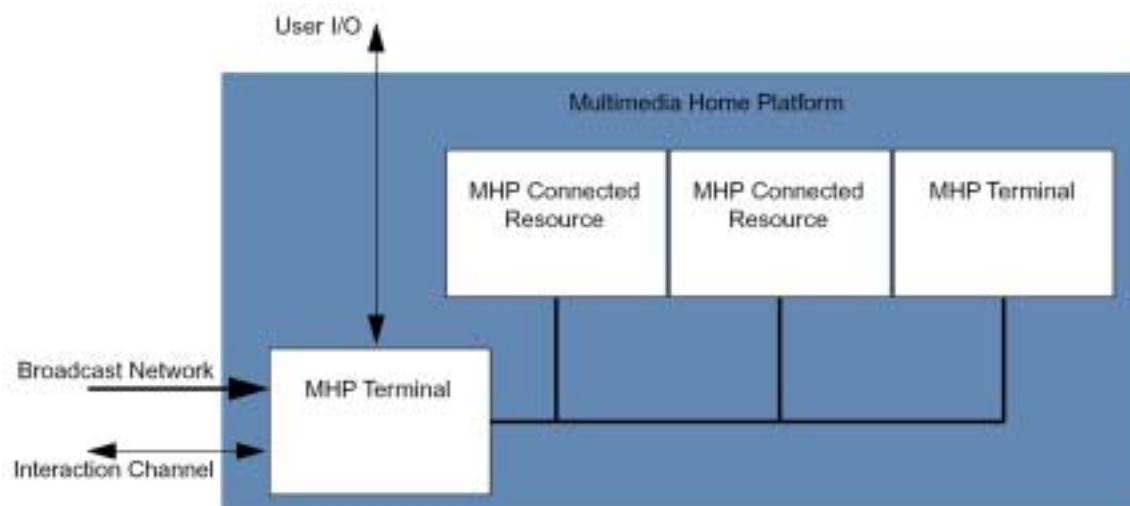


圖 1-2：MHP 與外界接觸的介面

應用程式所能夠存取的 MHP 資源，可能會分散在幾個互相連接的硬體裝置中。

另外，可能會有局部性叢集（local cluster）將幾個 MHP 終端機（MHP terminal）與資源連接起來。

此類叢集也可能含有一些 MHP 基礎架構以外的資源，這些資源應用程式並無法存取。

此類局部性叢集必須符合 DVB IHDN 規格。有關 MHP 在此類局部性叢集中的定義，並不在本規格第一版的範圍內。

1.2 架構

本架構說明 MHP 軟體元素的組織。

MHP 模型基本上包含三層（圖 1-3）：

資源

系統軟體

應用程式

以應用程式的角度來看，應用程式開發介面（API）則位於應用程式與系統軟體這兩層之間。



圖 1-3：基本架構

1.2.1 資源

MHP 平台中的硬體裝置所擁有的功能，是以硬體或軟體資源方式呈現，至於這些資源如何群

組，則不一定。本模型假設在整個平台中應該不會只有一個硬體裝置。

從抽象的層面來看，邏輯上的資源是對應一個或多個硬體裝置，其實並沒有差別。重要的是，這些資源在 MHP 平台內使用時不必在乎這件事情。一個應用程式在存取所有本地連接的資源時，必須能夠將這些資源看成是單一個體中的一些元素。

1.2.2 系統軟體

應用程式無法對資源本身直接存取。因為，系統軟體已經將這些資源與應用程式抽離，應用程式看到的是抽象的資源，藉由這一個抽象層，可以讓應用程式具備可攜性 (portability)。本文件並不包含資源與系統軟體的實做說明。

1.2.2.1 應用程式管理員 (Application Manager)

在系統軟體中，包含了一個應用程式管理員，專門負責管理所有應用程式的生命週期，包括交互運作 (interoperable) 的應用程式。

1.2.3 應用程式

所謂應用程式，就是實現互動式服務的軟體，可以在一個或多個硬體裝置內執行。而 MHP 應用程式介面，就是從應用程式的角度往下所看到的系統軟體。

圖 1-3 所顯示的就是 MHP 的一個理想化流程架構模型。整個流程的控制是階層式的，上層的流程會控制下層的流程。最頂端的一層，則利用互動式的應用程式，來控制整個運作。由於應用程式管理員是系統軟體的一部分，因此，會隨平台實做而定 (implementation specific)。它負責與所有應用程式互動。

系統軟體負責執行所有 API 的功能，並提供下列抽象物件：

- 不同播放來源的串流與傳導這些串流的管道 (pipe)

- 指令與事件 (event)

- 資料記錄或檔案

- 硬體資源

API 提供這些物件的相關服務給應用程式使用。

事實上，應該會有許多執行不同服務與介面的 API 才對。這些在本規格當中都有詳細說明，若沒有，也會指向外部文件供讀者參考。

本規格描述廣播網路、應用程式以及 MHP 系統軟體之間的介面。而這三者本身內部的實做 (implementation)，則不在本文件的範圍內。

1.3 MHP 應用程式與 MHP 系統之間的介面

應用程式需透過 API 來存取接收器 (receiver) 資源,包括:資料庫 串流媒體解碼器 (streamed media decoder)、靜態內容解碼器 (static content decoder) 以及通訊。這些資源都屬於接收器的功能，而最後也可能會對應到接受器中的某些硬體裝置。

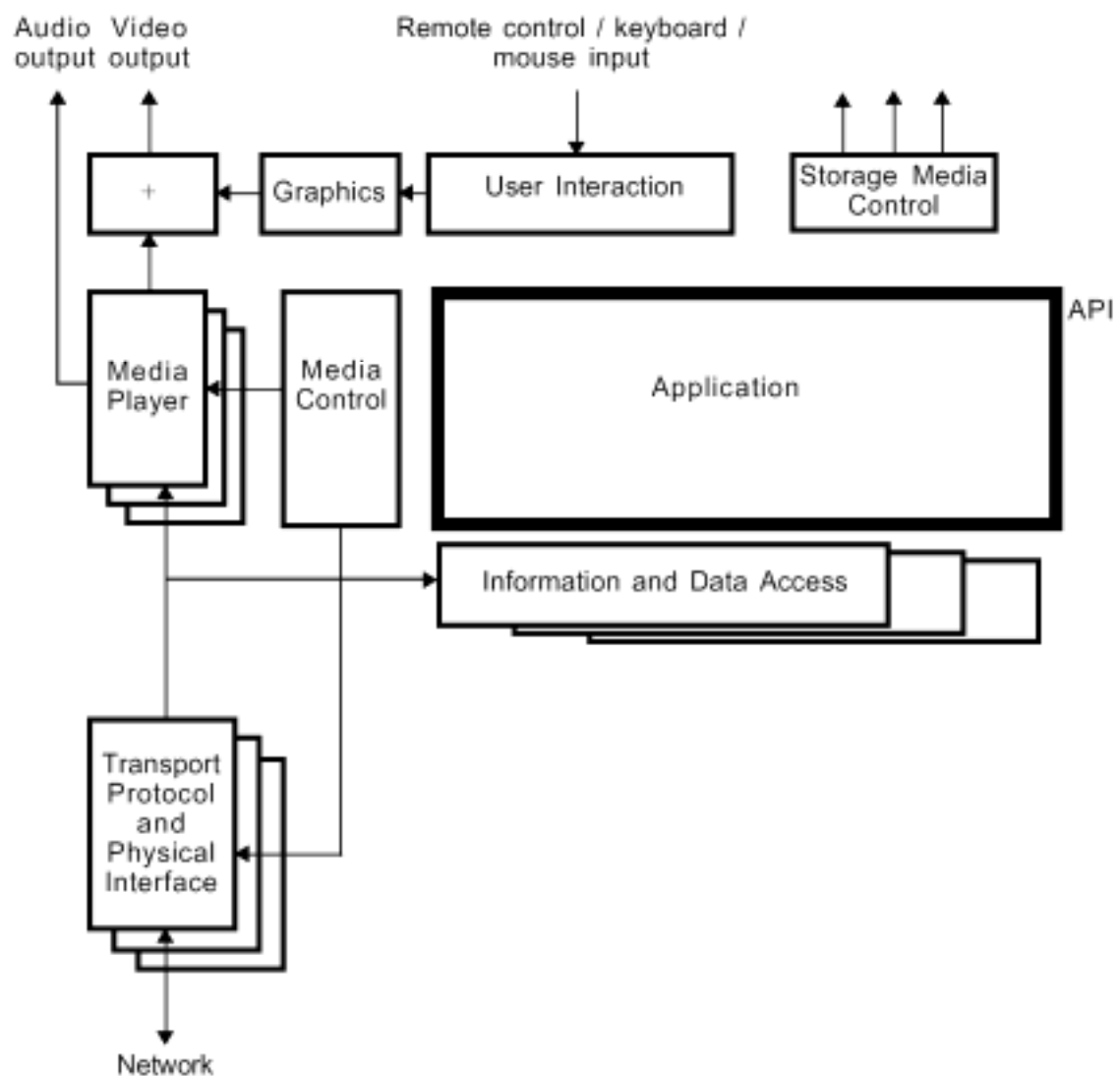


圖 1-4：MHP 應用程式與 MHP 系統之間的介面

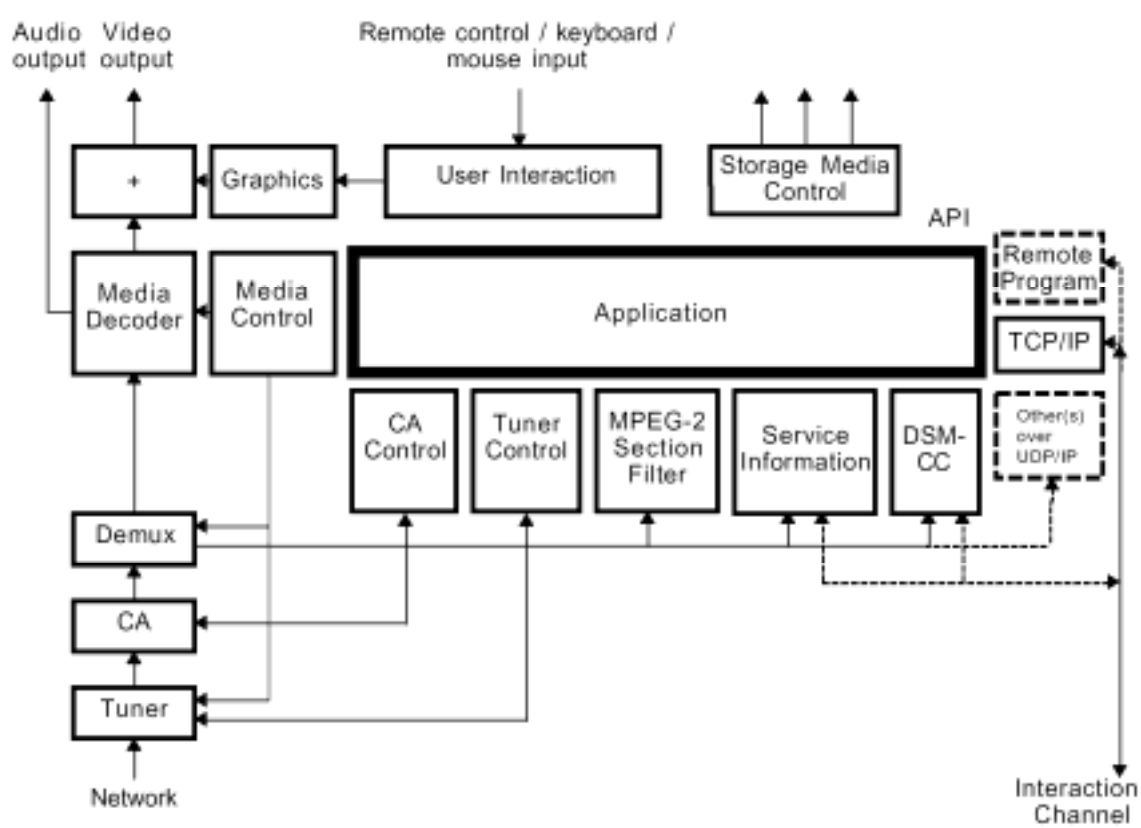


圖 1-5：MHP 應用程式與 MHP 系統之間介面的詳細內容

圖 1-4 和圖 1-5 顯示這些介面以及這些介面與 MHP 系統內部的媒體與資訊流動的關係（不考慮局部性叢集）。第一張圖顯示一個通用的 MHP 系統，第二張圖，則是一個增強型廣播服務或是互動式電視系統，其中還包括一些額外的選項。

在圖 1-4 和圖 1-5 中，只有應用程式的邊界部分（也就是介面部分）才是本規格訂定的範圍。其他都是實做的部分，在此僅供參考。

1.4 外掛模組

所謂外掛模組（plug-in），是指一組能夠加入這個通用平台的功能，以提供本規格中沒有訂定的應用程式與內容格式的解譯，加強 MHP 終端機的功能。

註：要讓標準 MHP 平台與其他平台交互運作的機構，必須清楚定義這些平台的外掛模組。

要使用哪一種外掛模組，必須由使用者來決定，這樣，使用者才能選擇服務來源。使用者的選擇可以有很多種方式，包括另外購買內建外掛模組的設備、手動下載外掛模組、自動下載外掛模組（記憶體足夠的話）等等。

只要是該平台實做上允許，外掛模組可以隨時保留在記憶體內。MHP 系統在載入外掛模組並正常運作之後，必須能夠像內建這種格式支援的平台運作一樣。

圖 1-6 顯示 MHP 兩種外掛模組類型。

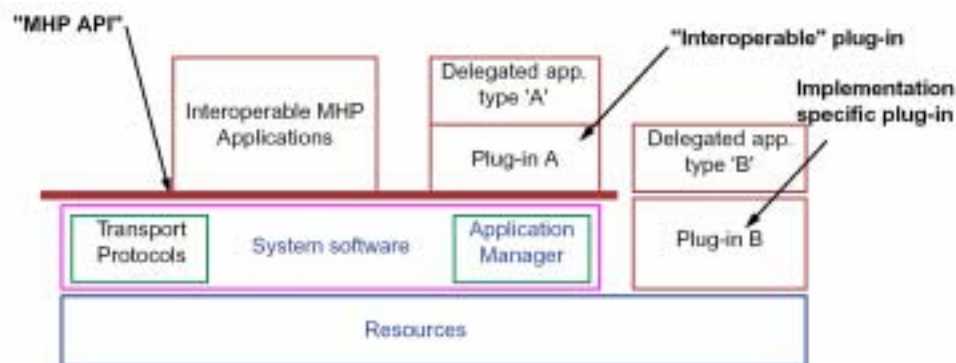


圖 1-6：不同方式的外掛模組

外掛模組在實做上有兩種方式：

使用隨實做而定的程式碼（也就是原生程式碼 native code，或是隨實做而定的 Java API）。這就是所謂的「隨實做而定的外掛模組」（implementation-specific plug-in），也就是圖中的外掛模組 B。

使用 MHP 應用程式。這種方式稱為「交互運作式外掛模組」（interoperable plug-in），也就是圖上的外掛模組 A。

至於這兩種外掛模組（A 和 B）的內部規格，並不在本規格的範圍之內。在這裡只是讓讀者知道它們與 MHP 平台的關係而已。

1.4.1 安全模型

外掛模組必須要有足夠的權限來存取 MHP 平台內的資源，才能夠進行相關規格的實做。隨實做而定的外掛模組可以跳過 MHP 的安全模型而直接存取平台中的許多資源。所有外掛模組所執行的應用程式，其安全性都必須由外掛模組自行負責管理。

如果外掛模組為了支援舊的（legacy）功能，而需要特殊權限來存取所有下載的應用程式皆無法

存取的資源（也就是不在「沙盒」sand box 內），外掛模組就需要適當的認證。

2 傳輸協定

2.1 簡介

為了能夠與外界溝通，MHP 要有能力透過不同的網路類型連繫。這部分的 MHP 規範是處理與網路獨立的協定，這裡所指的網路在 DVB 計畫的兩份規範中有定義：ETSI ETS 300 802[16]和 ETSI EN 301 192[5]。

在 ETSI EN 301 192[5] 標準中所定義的協定，提供一般性的解決方案給純粹的廣播服務 (Broadcast only)；在 ETSI ETS 300 802[16] 標準中所定義的協定，提供一般性的解決方案給互動服務。這些服務透過使用 ISO/IEC 13818-6[26] 標準詳載的 DSM-CC 使用者對使用者協定 (User-to-User)，資料連續撥放協定 (Data Carousel) 及物件連續撥放協定 (Object Carousel)；另外在互動頻道可支援 IP 協定，在廣播頻道透過 ETSI EN 301 192[5] 定義的多重協定封裝協定標準 (Multiprotocol Encapsulation) 的方法支援 IP 協定。

在一條從上而下的資料流動頻道組成的系統上所提供的服務，即是純粹的廣播服務。服務提供者在上，接受服務的消費者在下。

在一條從上而下的資料流動頻道與互動頻道共同組成的系統上所提供的服務，即是互動服務。

有很多可能的網路配置型態，涵蓋現有明定在 DVB 廣播型態及其他互動頻道型態。DV 型態的選擇包括衛星、地面廣播、有線傳輸、與 PSTN、ISDN 及有線傳輸結合的 SMATV 和 MMDS。

DVB 文獻中提及的互動頻道，與網路相依協定分別在 ETSI ETS 300 800[14] 定義 CATV 網路，在 ETSI ETS 300 801[15] 定義 PSTN/ISDN 網路，在 ETSI EN 301 193[6] 定義 DECT 網路，在 ETSI EN 301 195[7] 定義 GSM 網路，在 ETSI EN 301 199[8] 定義 LMDS 網路，在 ETSI TR 101 201[50] 定義 SMATV 網路。網路相依協定與網路獨立協定共同運作。

2.2 廣播頻道協定

這個章節是關於 DVB 定義的廣播頻道協定或參照其他既有的廣播頻道協定。這章節並不考慮其他協定，以及將會提供存取它們的 API。其他協定和它們的 API 是被考慮成 DVB MHP 平台的延伸。

Figure XX 圖示一套 DVB 所定義的廣播協定，MHP 應用程式適於某些協議 (profiles) 或全部協議

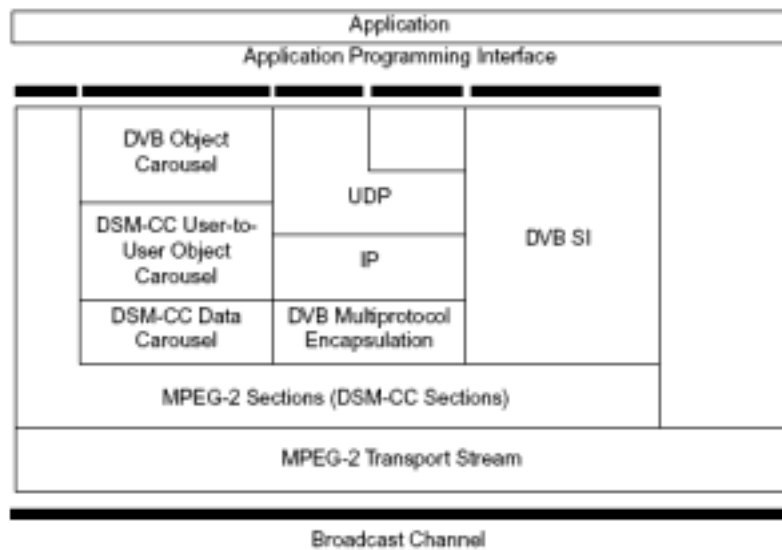


圖 2-1: 廣播頻道協定堆疊。

除了 MPEG-2 sections(見 2.2.2, "MPEG-2 sections")的情況以外, 當一個 MHP 應用程式想要透過廣播頻道, 存取混雜著條件式存取限制的資料時, 這個應用程式不須明確地要求它, MHP 的終端程式就會企圖對這筆資料, 做解除條件式存取限制的初始化工作。但在 MPEG-2 sections 這層, 若要存取混雜著條件式存取限制的資料時, 須要對它明確地要求做解除條件式存取限制的初始化工作, 否則就無法取得這筆資料。

2.2.1 MPEG 傳輸流(MPEG Transport Stream)

MPEG-2 傳訊串列是定義在 ISO/IEC 13818-1[23]。

2.2.2 MPEG-2 區段(MPEG-2 sections)

MPEG-2 私有的區段基於在 ISO/IEC 13818-1[23] 規範, 2.2.1 中定義的 MPEG-2 傳輸流協定。

2.2.3 DSM-CC 私有的資料(Private Data)

DSM-CC 私有的資料定義在 ISO/IEC 13818-6[26]。

2.2.4 DSM-CC 資料連續撥放(Data Carousel)

DSM-CC 資料連續撥放定義在 ISO/IEC 13818-6[26]。

2.2.5 DSM-CC 使用者對使用者物件連續撥放(User-to-User Object Carousel)

DSM-CC 使用者對使用者物件連續撥放協定,定義在 ISO/IEC 13818-6[26]。在 ETSI EN 301 192[5] 規範對其限制的部分,在 ETSI TR 101 202[51]中則有延伸定義。

2.2.5.1 DVB-J 類別檔案

每個 Java 類別的 Java 位元碼編碼方式明確的定義於 Java VM[36]。Java 類別在利用 `BIOP::FileMessage` 訊息攜帶時,訊息的內容位元就是一個如前所述一致的 " 類別 " 檔案。

2.2.5.2 DVB-HTML 文件檔案

這一套文件定義 DVB-HTML 應用程式在利用 `BIOP::FileMessage` 訊息傳輸時,訊息的內容位元對應到一致的文件內容(即是這個 `BIOP::FileMessage` 訊息不包括任何 HTTP headers,等)。

2.2.5.3 連續撥放內容遺失的作用方式(Loss of Carousel Behaviour)

在某些情況下,連續撥放的資料流檔案廣播系統可能失去功效。

當這種情況發生時,若實作方法中有將資料存入快取,則可能會從連續撥放內容中,繼續傳送遺失的資料。這個方法的使用範圍很明顯地和實作方法相依。這個方法隨著時間改變的特性也和實作方法相依。連續撥放內容永久性遺失將永遠無法被自動復原。暫時性斷線和重新連線會自動回復,而且對應用程式而言,是非常地不容易覺察到。然而,實作方法必需預留 Java IO 相關語法 (semantic)。例如:`InputStream.available()`方法必需精確地回報除了被阻絕的部分,可用的位元組數量。

若資料未在某一個快取中,則應用程式將無法取得這些資料。當應用程式企圖從永久性遺失的連續撥放內容中存取沒有用的資料,這個動作會導致存取失敗。這個失敗模式將會成為一個適當的內容格式和被用來存取這類資料的機制。

在連續撥放內容的重新連接上,系統必須開始擷取任何未接收的資料。每次收到資料後,任何阻斷式 I/O 運作都要回報。

假使系統裁定連線失效是無法復原的話,一個暫時性的斷線可能會變成永久性的失效。

可能發生這類實例的情況如下:

- 連續撥放內容變成永久性失效。
- 連續撥放內容斷線超過 60 秒時。

假設這類實例發生了,任何阻斷式 I/O 運作應該要用 `InterruptedException` 的方式中止服務,而且 `ServiceDomain` 必須進入被分離(detached)的狀態。一個實作方式可能選擇使用一個更長的過時(timeout)數值,以決定超過 60 秒時限後,連續撥放內容才會進入永久性失效的策略。例如,為了執行應用程式的啟動型連續撥放內容,會比由應用程式附掛的連續撥放內容有較高的優先權。

2.2.6 DVB 多重協定封裝

DVB 多重協定封裝定義在 ETSI EN 301 192[5]，提供 IP 和 DSM-CC 私有的資料協定的支援。

2.2.7 網際網路通訊協定(IP)

網際網路通訊協定定義在 IETF RFC 791[45]。

2.2.8 用戶資料訊息協定(UDP)

用戶資料訊息協定定義在 IETF RFC 786[44]。

2.2.9 DVB 資訊服務(DVB Service Information)

DVB 資訊服務定義在 ETSI EN300 468[4]和 ETSI ETR 211[11]。

2.2.10 IP 訊號服務(IP Signalling)

IP 通知表(IP Notification Table, INT)定義在 ETSI EN 301 192[5]。

2.3 互動頻道協定(Interaction Channel Protocols)

這個部分處理關於 DVB 定義的或參考的互動頻道協定。這部分並不考慮其他協定和可能提供存取它們的 APIs。其他私有的協定和可能的 APIs 並不被排除，而是超出 MHP 規範的範疇。

Figure 圖解說明一套 DVB 定義的互動頻道協定是可以被 MHP 應用程式使用某些協議(profile)或完整的協議。

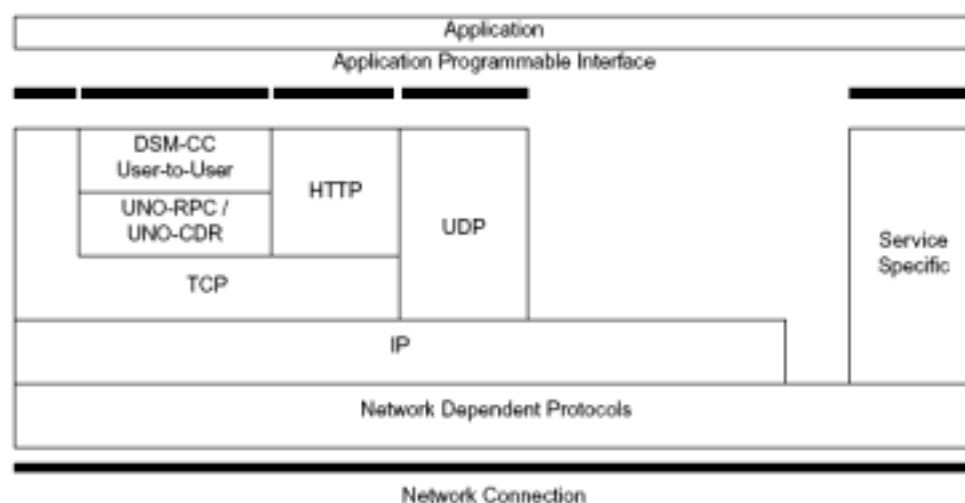


圖 2-2：互動頻道協定堆疊。

2.3.1 網路相依協定(Network Dependent Protocols)

如同在 ETS 300 800[14] 定義 CATV 網路，在 ETS 300 801[15] 定義 PSTN/ISDN 網路，在 EN301 193[6] 定義 DECT 網路，在 EN301 193[6] 定義 GSM 網路，在 EN 301 199[8] 定義 LMDS 網路，在 TR 101 201[50] 定義 SMATV 網路。

2.3.2 網際網路通訊協定(IP)

網際網路通訊協定定義在 IETF RFC 791[45]。

2.3.3 傳輸控制通訊協定(TCP)

傳輸控制通訊協定是定義在 IETF RFC 793[46]。

2.3.4 UNO-RPC

UNO-RPC 由 Internet Inter-ORB Protocol(IIOP)定義在 CORBA/IIOP[2]。

2.3.5 UNO-CDR

UNO-CDR 定義在 CORBA/IIOP[2]。

2.3.6 DSM-CC 使用者對使用者(DSM-CC User-to-User)

DSM-CC 使用者對使用者定義在 ISO/IEC 13818-6[26]。其限制規定和延伸定義在 ETSI EN 301 192[5]和 ETSI TR 101 202[49]。

2.3.7 超文件傳輸協定 (Hypertext Transfer Protocol (HTTP))

2.3.7.1 HTTP 1.1 版

超文件傳輸協定定義在 IETF RFC 2616[40]。

2.3.8 特有的服務(Service Specific)

特有的服務協定是具有專利的協定，這項服務的使用方法的指導方針定義在資料廣播規範，ETSI TR 101 202[51]。DVB 提供註冊的機制給新的、具有專利的廣播協定。

2.3.9 用戶資料訊息協定(UDP)

用戶資料訊息協定定義在 IETF RFC 786[42]。

2.3.10 網域名稱服務(DNS)

MHP終端程式至少必須實作DNS解悉協定。這個解悉協定可以提早將完全合格的網域名稱轉譯成IP位址。IETF RFC 1034 [83]和IETF RFC 1035 [84]定義DNS解悉協定，IETF RFC 1982 [85]和IETF RFC 2181 [86]則清楚闡明此協定。

以連接為基礎的返回通道，DNS伺服器位址可同時由網路和MHP終端程式提供。網路僅是建立連接的一部分，而且由此獲得位址的會被優先處理。

特有的服務協定是具有專利的協定，這項服務的使用方法的指導方針定義在資料廣播規範，ETSI TR 101 202[51]。DVB 提供註冊的機制給新的、具有專利的廣播協定。

2.3.9 用戶資料訊息協定(UDP)

用戶資料訊息協定定義在 IETF RFC 786[42]。

2.3.10 網域名稱服務(DNS)

MHP終端程式至少必須實作DNS解悉協定。這個解悉協定可以提早將完全合格的網域名稱轉譯成IP位址。IETF RFC 1034 [83]和IETF RFC 1035 [84]定義DNS解悉協定，IETF RFC 1982 [85]和IETF RFC 2181 [86]則清楚闡明此協定。

以連接為基礎的返回通道，DNS伺服器位址可同時由網路和MHP終端程式提供。網路僅是建立連接的一部分，而且由此獲得位址的會被優先處理。

3 內容格式

3.1 靜態格式

3.1.1 點陣圖影像格式

3.1.1.1 影像編碼限制

影像顯示時，任何有關像素縮放、色彩空間、或是 gamma 值的資料都將被忽略。要顯示的影像像素與當前組態的繪圖裝置像素之間，將是一對一對應，除非經過應用程式自行縮放。

請參閱「3.5 色彩表示方式」一節。

3.1.1.2 JPEG

JPEG 格式如 ISO/IEC 10918-1 [21] 所定義，該格式使用 JFIF [35] 檔案作為交換格式。

平台實做只需支援序列式 DCT (sequential DCT) 和漸進式 DCT (progressive DCT) 兩種模式的編碼。

尤其不需要支援無損失 (lossless) 與階層式 (hierarchical) 模式。

3.1.1.3 PNG

PNG 格式如 PNG [37] 所定義。

3.1.1.4 GIF

GIF 格式如 GIF 89a [17] 所定義。

3.1.2 MPEG-2 I-Frame

MPEG-2 I-frame 格式如 ISO/IEC 13818-2 [24] 所定義。

一個傳送 MPEG -2 I-frame 的檔案，其包含的內容應該：

是一個符合規格的 video_sequence() 並包含一個 sequence_extension()

只包含一個 I-frame,也就是包含一個 picture_header()、一個 picture_coding_extension() 以及一個 picture_data(), 並且必須是一個差異編碼的畫面 (intra coded frame), 同時 picture structure = "frame"。

其資料結構如下：

```
sequence_header()
sequence_extension()
extension_and_user_data(0)
optional group_of_pictures_header() and extension_and_user_data(1)
```

```

picture_header ( picture_coding_type = "I frame")
picture_coding_extension ( picture_structure = "frame picture")
extension_and_user_data(2)
picture_data()
sequence_end_code()

```

3.1.3 MPEG-2 視訊涓流 (Video Drip)

在涓流模式下，應用程式以漸進 (progressive) 方式將 MPEG-2 視訊串流一段一段 (chunk) 傳入 MPEG-2 視訊解碼器。在此模式中，解碼器只需要處理 I-frame 和 P-frame (也就是不用處理 B-frame)。每一段 (chunk) 都應該包含一個畫面和一些語法元素 (如 ISO/IEC 13818-2 [24] 所定義)，譬如 `sequence_header()` 或 `group_of_picture_header()`。

首先，傳入解碼器的每一段資料內容，必須符合下面的語法：

```

optional {
    sequence_header()
    sequence_extension()
    extension_and_user_data(0)
    optional {
        group_of_pictures_header()
        extension_and_user_data(1)
    }
}

picture_header ( picture_coding_type = "I frame" or "P frame")
picture_coding_extension ( picture_structure = "frame picture")
extension_and_user_data(2)
picture_data()
optional {
    sequence_end_code()
}

```

此外，在以時間串連各段資料時，必須遵守 ISO/IEC 13818-2 [24] 所規定的語法元素組合規則，以建立出一串符合規格的 MPEG-2 視訊串流。下圖是從 ISO/IEC 13818-2 [24] 所節錄出來的圖片，顯示該標準所訂定的規則：

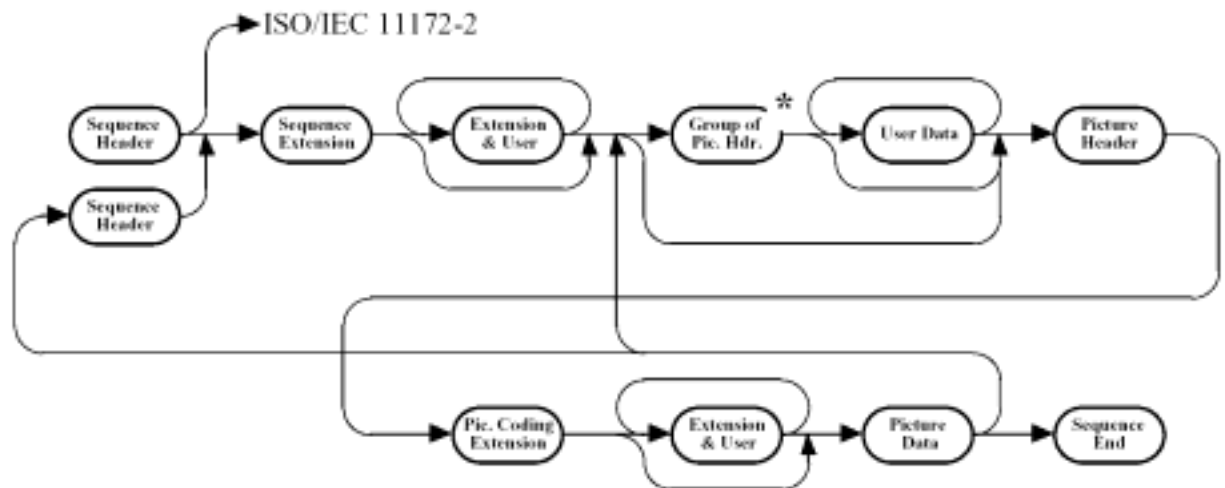


圖 3-1 在一個 GOP 之後的第一個圖必須是一個 I-picture

底下是 P-frame 的一些限制：

P-frame 不能包含預測性的資料，也就是說，在 macroblock 元素中，不能包含移動向量 (motion vector)。

P-frame 中所允許的 macroblock_types 有：

"Intra" (即 VLC code 0001 1)

"Intra, Quant" (即 VLC code 0000 01)

"No MC, Coded" (即 VLC code 01)

"No MC, Coded, Quant" (即 VLC code 0000 1)

註：P-frame 在標準中是允許使用 macroblock_escape 和 macroblock_address_increment 來標記省略的 macroblock。這讓 P-frame 可以非常希鬆，只需要包含在螢幕上某些位置的 macroblock 即可。這和 I-frame 有很大的差別，I-frame 的 macroblock 必須佈滿整個螢幕。

如果傳入 MPEG-2 視訊解碼器的內容不符合規格，內容就會被捨棄，同時，也無法保證下一段合乎規格的資料在傳入解碼器之後多久才會顯示出來（除非重新啟動解碼器）。

此模式要求解碼器必須處於 ISO/IEC 13818-2 [24] 所定義的「低延遲」(low delay) 模式。

此模式可以用來將一個 org.dvb.media.DripFeedDataSource 連接到一個播放器 (Player)，就變成了一套 MPEG-2 視訊解碼器。

3.1.4 音訊片段的單一媒體格式

音訊片段的格式為 MPEG-1 Audio (Layer 1 & 2) ES 資料，如 ISO/IEC 11172-3 [22] 所定義，

並受到 ETSI TR 101 154 [9] 所規範。

每一個音訊內容「檔案」都是二進位資料 (binary data) 檔案，其中包含音訊基礎串流資料。每一個「檔案」都包含整數個數的音訊存取單元 (audio access unit)，而且，每個檔案的第一個位元組 (byte) 同時也是音訊存取單元的第一個位元組。MPEG Audio 資料在所有其他方面都符合 ETSI TR 101 154 [9] 的規格。

在音訊片段解碼時，可以假設每一秒的位元組數目 (資料量) 大致上相同。如果遇到不相同的情況，由實做自行決定如何處理。

3.1.5 文字的單一媒體格式

Java 對 UTF-8 做了一些修改，如 Java Language Spec [32] 中第 22.2.14 節所述，而「writeUTF」就是 MHP 對文字的編碼方式。

註：這基本上是根據 ISO 10646-1 [18] 標準，但對字碼零 (character code zero) 的編碼有所修正。

3.1.5.1 內建字元集

3.2 廣播串流格式

3.2.1 音訊

採用 MPEG Audio，同時納入 ETSI TR 101 154 [9] 所定義的限制與增強功能。

3.2.2 視訊

採用 Standard Definition 25 Hz MPEG Video，同時納入 ETSI TR 101 154 [9] 所定義的限制與增強功能。

3.2.3 字幕

MHP 所支援的字幕內容格式有：

DVB 字幕

Teletext

請參閱「7.5 字幕」一節。

如果 DVB 字幕和 DVB Teletext 同時存在，那麼 DVB 字幕有優先權 (也就是如果一個串流同時包含 DVB 字幕和 Teletext 字幕，那麼就會顯示 DVB 字幕)。

Teletext 字幕也與 DVB 字幕採用相同的顯示模型。

應用程式對字幕的控制與偵測，不論是 DVB 字幕或 Teletext 字幕，都是透過 JMF 來進行。應用程式將無法獲知字幕傳送與呈現的協定。

未來將不會提供 API 來存取 Teletext 資料封包，也不會提供 Teletext 字幕解碼用的時序模型 (timing model)。文字字幕資料在到達時就會立刻解碼。

3.2.3.1 DVB 字幕

DVB 字幕 如 ETSI EN 300 743 [13] 所定義。

3.2.3.2 Teletext

文字的傳送如 ETSI EN 300 472 [12] 所定義。資料格式如 ETSI ETS 300 706 [61] 所定義，但僅限於 1.5 或更低的呈現等級 (presentation level)。Teletext 字幕頁面將透過 ETSI EN 300 468 [4] 定義的 Teletext 描述器 (descriptor) 來通知。

在 MHP 規格中，Teletext 只是字幕格式的額外選擇。

MHP 規格並不特別支援使用 Teletext 做為可以瀏覽的內容格式。

註：廠商可以自由選擇支援 Teletext 的完整功能，以符合當地市場需求或法規要求。此類功能將在 MHP 環境之外提供支援，譬如在沒有字幕的影片上重新插入 VBI，或是透過原生的 (native) Teletext 解碼器。此時，使用者介面之整合，就是廠商必須自行解決的問題。

可預見地，廣播業者將會透過 MHP 應用程式來傳送可瀏覽的文字服務，以便提高互動性與增強繪圖能力。

3.3 內建字型

3.4 可下載字型

PFR0 (Portable Font Resource version 0) 是 MHP 字型的格式，如 DAVIC 1.4.1p9 [3] 所定義。接收器只需要支援外框字型版本。

在 PFR charRecord 中的 charCode 值，必須是 ISO 10646-1 [18] 中以 UCS-2 編碼的字碼。

對於 PFR 格式的字型來說，字型名稱必須是字型檔案中的 fontID 欄位。

PFR 檔案中的每一個字型在其實際的字型記錄中都可以有自己的輔助資料 (auxiliary data)。如果有，這些輔助資料必須符合下表中的語法。同時，在其資料區塊的最後必須有一個長度 (blockLength) 為零、類型 (blockType) 為零的區塊做結尾。

表 3-1：PFR 輔助資料

	位元數	識別碼
--	-----	-----

<pre> auxDataFontRecord() { do { blockLength blockType switch (blockType) { case 2: for (i = 0; i < 10; i++) { reserved } ascent descent reserved externalLeading for (i = 0; i < (blockLength - 24); i++) { reserved } break; default: for (i = 0; i < (blockLength - 4); i++) { reserved } break; } } while (blockLength > 0 blockType != 0) } </pre>	16	uimbsf
	16	uimbsf
	8	uimbsf
	16	tcimbsf
	16	tcimbsf
	32	uimbsf
	16	tcimbsf
	8	uimbsf
	8	uimbsf

一筆字型輔助資料記錄是由一群區塊所組成，最後必須以一個長度為零、類型為零的區塊結尾。

blockLength：本區塊總共的位元組數目，包括 blockLength 和 the blockType 兩欄位。

blockType：一個定義區塊內資料類型的數字。0x0000-0x7fff 為保留區間，0x8000-0xffff 為使用者自訂類型的區間。

ascent：代表字型的上升高度，是從字型基準線到絕大部分字元字型頂端的距離。

descent：代表字型下降高度，為絕大多數字元底部到基準線的距離。

externalLeading：代表一行文字與前一行文字之間必須保留的距離。字型設計者可以將此值設為零。

MHP 終端機在計算字型大小時，應該忽略第二種類型（type 2）字型資料區塊中的保留欄位。至於第二類型以外的區塊應如何處理，則 MHP 終端機可自行決定，因此，MHP 應用程式不應使用這些區塊類型。

3.5 色彩表示方式

3.5.1 背景（參考資料）

色彩的編碼方式，對於如何在不同的系統中顯示一致的色彩，有決定性的影響。其描述不能限制在某一種最終顯示機制上。

International Colour Consortium (ICC) 已經提出一套完整的方案，可在開放的系統之間精確溝通色彩資訊。不過，ICC 色彩描述 (profile) 格式對於 MHP 來說，規格太大。在 ICC 機制中，為確保色彩能夠在輸入裝置與輸出裝置之間正確對應，因此，會在影像上附加輸入裝置的色彩描述。這對於高階系統來說非常適合，特別是印刷方面。但是，對於一個以 CRT 螢幕為主的家庭平台，並不需要、也無此運算能力和頻寬來處理色彩描述內嵌機制。如果這麼做，消費者還必須設定正確才行。

所幸，我們可以採用單一預設的色彩空間，當做大家默認的 ICC 色彩描述，如此，不但能夠享受到 ICC 規格的好處，同時將來也能擴充到完整的色彩管理系統，不僅和 ICC 色彩管理保持相容，又能降低對今日 MHP 在軟體與支援上的要求。

一個色彩空間，是一套以三度或四度座標數值，或是以三刺激值來表示顏色的一個模型。一個 RGB 色彩空間是以紅 (R)、綠 (G)、藍 (B) 三個座標來表示一個顏色。MHP 使用彩色影像編碼所用的 sRGB 做為其色彩格式，如 IEC 61966-2-1 [27] 所定義。此格式可應用到很廣的呈現環境，包括電視、電腦與網頁。譬如，此格式就與 CCIR Recommendation ITU-R BT.709 [30] 標準所定義的 HDTV 色彩編碼相容。同時，此格式也可以不需太多的額外負擔，就能擁有與裝置獨立 (device independent) 的優點。

sRGB 的目標，是要以三個數位化的色彩座標值 (每個 8 位元) 來代表一個參考螢幕 (reference monitor) 上所顯示的顏色。sRGB 色彩值的定義，是根據一個參考觀賞環境所呈現的色彩而定。

在這個參考觀賞環境中所看到的色彩，sRGB 數值是根據標準 CIE 色彩值經過一連串簡單的運算而來。

sRGB 格式非常適合絕大多數 CRT 螢幕上的 24 位元色彩。不過，在一些會大幅破壞色彩空間的裝置上，可能就無法得到一致的色彩。譬如，以遞色 (dithering) 方式對應到每一個原色 (primary color) 只有四位元的空間，就會破壞原先假設的 gamma 值。

3.5.2 規格

所有傳送的影像，都必須在 sRGB 的色域所涵蓋的範圍內。當情況允許時，應盡可能以這樣的方式編碼，如此，MHP 終端機才不用做任何轉換。若實際上不可行，此 sRGB 影像可以先以別的色彩空間來編碼，只要不違反假設的色域即可 (也就是說，為了與 JFIF 一致，所傳送的 JPEG 影像只能用到 YCrCb 與 sRGB 重疊的色域)。

註：當遇到影像中超過 sRGB 色域的顏色時，其呈現的效果將隨平台實做而定。

由 MHP 所產生的影像，預設使用 sRGB 色彩空間，當然，廠商仍可以支援其他色彩空間。所有 MHP 平台都必須支援從 sRGB 轉換到 MPEG-2 所允許的色彩空間 (例如：BT 709 和 BT 420)，反向轉換亦然，此外，廠商還可以支援與其他色彩空間之間的轉換。

3.5.2.1 sRGB 參考觀賞環境

表 3-2 列出部分 sRGB 參考顯示條件與觀賞環境。要將色彩精確定義，就必須提供一個參考觀賞環境。RGB 的參考觀賞環境是典型室內 CRT 螢幕觀賞環境，細節請參考 IEC 61966-2-1 [27]

所定義。

因此，sRGB 的參考條件提供了一套和 ITU-R BT.709 [30] 相容的參考條件。

表 3-2：sRGB 參考顯示條件

條件	sRGB
Viewing flare	1.0 %
Reference Background	20 %
Display model Offset	0.055
Display Gun/Phosphor Gamma	2.4
Display white point	x = 0.3127 y = 0.3290 (D65 Hunt, R.W.G. [52])
Ambient Lighting	64 lx
Display Luminance level	80 cd/m2

3.5.2.2 色彩定義與編碼

sRGB 三刺激值 (tristimulus value) 的計算如下 首先, 以下面線性組合關係根據 1931 CIE XYZ (CIE 15 [1]) 定義的 CIE 數值計算出線性的 sRGB 三刺激值。

$$\begin{bmatrix} R_{sRGB} \\ G_{sRGB} \\ B_{sRGB} \end{bmatrix} = \begin{bmatrix} 3,2410 & -1,5374 & -0,4986 \\ -0,9682 & 1,8760 & -0,0416 \\ 0,0556 & -0,2040 & -1,0570 \end{bmatrix} \begin{bmatrix} X_{D65} \\ Y_{D65} \\ Z_{D65} \end{bmatrix}$$

在編碼過程中，負的 sRGB 三刺激值以及大於 1.00 的三刺激值都捨棄。將 sRGB 三刺激值裁切到 0.0 到 1.0 之間，以限制亮度動態範圍和色域。不過，這樣的色域已經涵蓋絕大多數 CRT 螢幕所能呈現的色彩。

為了比較，我們將 ITU-R BT.709 和 ITU-R BT.470 紅、綠、藍參考原色以及 CIE Standard Illuminant D65 (IEC 61966-2-1 [27]) 的 CIE 色彩值 (chromaticity) 分別列於表 3-3 和表 3-4。對於這些原色所傳送的 YCbCr 數值也是以類似關係計算而來

因此，ITU-R BT.709 YCbCr 色彩空間以及類似的視訊色彩空間，都可以經由 CIE XYZ 色彩空間來與 sRGB 互相轉換。其他 MPEG 串流中允許的視訊格式，其色彩值 (chromaticity) 請參考相關的標準。

表 3-3：ITU-R BT.709 參考原色與 CIE standard illuminant

	紅	綠	藍	D65
X	0.6400	0.3000	0.1500	0.3127
Y	0.3300	0.6000	0.0600	0.3290
Z	0.0300	0.1000	0.7900	0.3583

表 3-4：TU-R BT.470-2 參考原色和 CIE standard illuminant

	紅	綠	藍	D65
X	0.6700	0.2100	0.1400	0.3100
Y	0.3300	0.7100	0.0800	0.3160
Z	0.0100	0.0800	0.7800	0.3740

接著，將線性 sRGB 三刺激值轉換到非線性的 sR'G'B' 值。此程序將產生非常接近 gamma 2.2 曲線的效果（帶一點誤差）。這樣可以讓 sRGB 與老舊（legacy）的系統與影像一致。

```

if RsRGB > 0,00304 and GsRGB > 0,00304 and BsRGB > 0,00304
then
    R'sRGB = 1,055 * RsRGB (1,0/2,4) - 0,055
    G'sRGB = 1,055 * GsRGB (1,0/2,4) - 0,055
    B'sRGB = 1,055 * BsRGB (1,0/2,4) - 0,055
else
    R'sRGB = 12,92 * RsRGB
    G'sRGB = 12,92 * GsRGB
    B'sRGB = 12,92 * BsRGB
end if

```

最後，再將非線性 sR'G'B' 值進行數位編碼。這道手續以下的公式將 sR'G'B' 進行縮放，其中 WDC 表示白色數位值，KDC 則代表黑色數位值。

$$R_{8bit} = ((WDC - KDC) * R'_{sRGB}) + KDC$$

$$G_{8bit} = ((WDC - KDC) * G'_{sRGB}) + KDC$$

$$B_{8bit} = ((WDC - KDC) * B'_{sRGB}) + KDC$$

目前的 sRGB 規格在 24 位元（每一通道 8 位元）的編碼方式中，以 0 為黑色數值、以 255 為白色數值，而 MHP 也將採用相同的方式。不過，有些數位視訊訊號可能會使用 16 作為黑色值、以 235 作為白色值，以便提供一個更廣的色域。

有關從 sRGB 轉回 CIE XYZ 的細節，請參考 IEC 61966-2-1 [27]；從 ITU-R BT.709 和 ITU-R BT.470 轉換到 CIE XYZ 的細節，請參考 ISO/IEC 13818-2 [24]。

3.6 MIME 類型

表 3-5：檔案類型識別碼

MIME 類型	副檔名 (註 1)	內容定義
"image/jpeg"	".jpg"	如「3.1.1.2 JPEG」一節所定義。
"image/png"	".png"	如「3.1.1.3 PNG」一節所定義。
"image/gif"	".gif"	如「3.1.1.4 GIF」一節所定義。
"image/mpeg"	".mpg"	如「3.1.2 MPEG-2 I-Frame」一節所定義。
"video/mpeg"	".mpg"	如「3.2.2 視訊」一節所定義。
"video/dvb.mpeg.drip"	".drip"	如「3.1.3 MPEG-2 視訊涓流」一節所定義。
"audio/mpeg"	".mp2"	如「3.1.4 音訊片段的單一媒體格式」以及「3.2.1 音訊」一節所定義。
"text/dvb.utf8"	".txt"	如「3.1.5 文字的單一媒體格式」。
"image/dvb.subtitle"	".sub"	如「3.2.3 字幕」一節所定義。
"text/dvb.subtitle"		
"text/dvb.teletext"	".tlx"	如「3.2.3.2 Teletext」一節所定義。
"application/dvb.pfr"	".pfr"	如「3.4 可下載字型」一節所定義。
"application/dvbj"	".class"	一個 DVB-J 類別檔案。請參閱「2.2.5.1 DVB-J 類別檔案」一節。
"multipart/dvb.service"	".svc"	一個符合 DVB 規範的 MPEG 節目 (DVB 服務)。
註 1：未來的格式可能會使用更長的副檔名。		

3.6.1 基本原則

此處定義的 MIME 類型是為了保留將來支援可下載 JMF 播放器之用。

副檔名應該包含在廣播內容當中，讓接收器可以辨識內容的類型。有關 DVB-J 應用程式部分，在「表 6-1：URL.getContent() 回傳值類型」有詳細說明。

本規格並未真正用到所有上述定義的 MIME 類型和副檔名。若某種 MIME 類型所對應的媒體並未列於某應用組態 (profile) 可用媒體中，則此類媒體內容便沒有定義此應用組態從檔案存取此內容的方式。

4 應用程式模型

4.1 MHP 廣播應用程式

4.1.1 基本生命週期控制

一個 MHP 廣播應用程式的基本生命週期控制，是透過廣播服務的選擇。廣播服務的選擇，可以由 MHP 終端機的使用者使用瀏覽器 (Navigator) 來起始，也可以經由提供 EPG 功能的 MHP 應用程式來起始。當 CI 模組發出 Host Control tune 請求，就會啟動服務選擇。使用 Host Control replace / clear_replace 請求與使用 javax.tv.media.MediaSelectControl 有相同的效果。

在 MHP 規格中，負責呈現與執行內容的單元是服務。MHP 當中的一個服務，代表一群必須一起呈現給使用者的內容片段。在本版本的規格中，所謂服務，就是 DVB 廣播服務的內容，包括音訊/視訊、資料串流、以及所有透過廣播傳送的服務資訊、應用程式和應用程式訊號。

每一個在 MHP 平台上呈現的服務，都必須在一個服務環境 (service context) 內呈現。這一點，是執行時期環境與執行模型的基本原則之一。所謂一個服務環境基本上就是服務呈現的環境。它定出服務的邊界 (讓平台與應用程式辨認，哪些目前呈現的內容片段代表同一個服務)。同時也讓整個服務當成一個單一個體來處理和控制。

在一個 DVB-J 應用程式中，一個服務環境是以 ServiceContext 類別的物件表示。若同一個服務環境中有許多個 DVB-J 應用程式一起呈現，此時，到底要用幾個 ServiceContext 物件來代表這個服務環境，是由實做來決定，不過，每個應用程式所看到的，仍舊是僅只一個。此時，只要任何一個應用程式對其 ServiceContext 物件做任何改變，其他應用程式所看到的 ServiceContext 物件也會發生相同的變化。DVB-J 若要獲得其所在的服務環境，可以透過 ServiceContextFactory 類別的 getServiceContext(XletContext) 方法。

一個 MHP 終端機的瀏覽器如何支援服務選擇，是隨實做而定的。不過，如果一個 MHP 應用程式正在使用遙控器的數字鍵，此時瀏覽器就不應該讓使用者透過這些數字鍵來選擇服務。因此，當使用者利用數字鍵輸入其個人識別碼 (pincode)，是不會選擇任何服務的。

一個服務環境，同一時間只能呈現一種服務。當您在一個服務環境中選擇某個了服務，該環境原來正在呈現的服務就會停止。任何先前的服務內容，若是不在這個新選取的服務中，將停止呈現。MHP 終端機可以限制同一時間可以呈現的服務環境數目。

在一個 DVB-J 應用程式中，要選擇一個服務，就使用此物件的 select() 方法。

4.1.2 啟動應用程式

當一個廣播服務被選取時，就會啟動該服務的 AIT 當中所列的「自動啟動」應用程式，不需要使用者手動操作（如「5.6 Control of application life cycle」一節所述）。在服務選取時所啟動的應用程式，將由該服務相關的通知訊號來控制。MHP 終端機必須監視這些訊號，以便獲知來自廣播業者的一些變化。這些變化包括終止某些應用程式，以及新增一些自動啟動應用程式。

在 AIT 中未標明為「自動啟動」的應用程式，將不會被 MHP 終端機自動啟動，必需手動啟動。而手動啟動的動作，可以由記憶體中的瀏覽器來執行，或是由另外一個 MHP 應用程式來執行。例如，使用者可以在畫面上看到應用程式選擇清單之後，才選擇啟動某個應用程式。由於記憶體中的瀏覽器並不一定要提供手動啟動的機制，因此，任何需要啟動應用程式的廣播服務都必須指定一個「自動啟動」的應用程式。

當一個服務環境目前所選取的服務包含多個 MHP 應用程式時，這些應用程式彼此之間都能互相啟動。被啟動的應用程式，將與服務在同一個服務環境中呈現。

一個 DVB-J 應用程式可以透過「應用程式列舉與啟動 API」來啟動其他程式。

4.1.3 支援多個應用程式同時執行

一個服務中所包含的應用程式都能同時呈現、並同時執行。

MHP 終端機必須能夠支援這些應用程式（使用相同的螢幕），至少必須達到本規格中有關最低平台能力的要求。MHP 終端機對於其允許同時呈現的每一個服務，也必須能讓其所包含的所有應用程式同時執行。

廣播業者必須確保這些同時執行的應用程式不會互相干擾或是造成使用者無法理解。

4.1.4 停止應用程式

MHP 應用程式可以利用 MHP API 自行停止，或是在某些狀況下由 MHP 終端機來停止。一些允許 MHP 終端機來終止的狀況包括：

4.1.4.1 當一個新選擇的服務取代原有服務時

如果正在執行的服務被一個新選擇的服務取代，原來正在執行的應用程式，若在新的服務當中也存在，則可以繼續執行，否則應該停止。如果一個應用程式不包含在新的服務當中，就會被 MHP 終端機所停止。若一個應用程式只限用於某一個服務，廣播業者可以將該應用程式設定為「限定服務」(service bound) 的應用程式，在應用程式的描述器 (descriptor) 中的

service_bound_flag 指定。此類應用程式將在服務切換時，立刻由 MHP 終端機停止，而不必靠 AIT 才能提供新的服務。這樣，新服務的自動啟動應用程式就可以早一點啟動。

4.1.4.2 應用程式由其他應用程式停止

一個應用程式可以請求另一個應用程式停止，但是還是要受 MHP 終端機的安全政策所規範。在此情況下，記憶體中的應用程式管理員 (Application Manager) 會先確認安全性沒有問題之後，才會停止該應用程式，否則，應用程式還是照常執行。

4.1.4.3 透過應用程式訊號來停止某個應用程式

廣播業者可以利用 AIT 當中所定義的控制碼，要求 MHP 終端機停止某個應用程式。此一動作的實際意義，需視應用程式的格式而定。

4.1.4.4 因為資源不足而被 MHP 終端機所停止

當一個 MHP 終端機的資源 (譬如記憶體、CPU、以及其他透過資源管理 API 所管理的資源等等) 不足，無法繼續執行某個應用程式時，MHP 終端機可以自行決定停止該 MHP 應用程式，不需使用者同意。

註：MHP 終端機確切的資源內容，是隨實做而定的。

4.1.5 不隨服務終止的應用程式

當一個執行中的應用程式，在原來的服務與新的服務當中都存在、且在原來的服務中沒有設定成「限定服務」時，該應用程式在切換服務的過程中，應該繼續執行，不得重新啟動。在此情況下，執行中的應用程式將由新的服務訊號所控制，而不再受舊的服務訊號掌控。因此 MHP 終端機必須監控新服務的 AIT，同時必須忽略舊服務的 AIT。

如果應用程式在原來的服務中設為「限定服務」，那麼，在新服務選取時就會正常停止。如果應用程式在新的服務中被設為「自動啟動」，那麼該程式就必須重新啟動，同時不能包含舊服務的非永久性環境內容 (volatile context)。

若一個應用程式在切換服務的過程中仍然繼續執行，它並不會自動能夠存取新服務中的新廣播檔案系統。基本上，此應用程式邏輯上還是與原先已經連接的廣播檔案系統連接著，不過這些檔案系統已經暫時與其 broadcast carousel 斷線。廣播檔案系統在這種情況下的行為表現，定義在「2.2.5.3 Loss of Carousel Behaviour」一節。

如果 broadcast carousel 暫時斷線，它會在選擇新服務的時候重新連接可用的 carousel。這也包括、但不限於應用程式原先所在的服務。要知道目前可用的 carousel，MHP 必須使用在掛載時 (mount) 從 carousel 所在的 home service 所得到的 PMT 資訊，以及當前選取服務的 PMT

資訊。

4.1.6 自動啟動管理

接收器在下列條件下，應該啟動「自動啟動」應用程式：

當訊號顯示接收器能夠支援此應用程式（比對應用程式設定檔 profile 以及應用程式描述器 descriptor 中的版本資訊）。

同一時間，應用程式識別碼（application ID）相同的應用程式，只能執行一份。

該應用程式是新加入的自動啟動應用程式，或是被重新設成自動啟動狀態。

因此：

當一個服務被選取時，接收器必須啟動它所支援的每一個自動啟動應用程式（每個最多一份）

在服務選取之後，如果有新的自動啟動應用程式加入（接收器可支援的），或是先前列出的可支援應用程式被重新設成自動啟動狀態，則該應用程式就根據正常的資源限制來決定是否啟動。

不過，如果某個自動啟動應用程式自行停止，則不應該再將它重新啟動，除非它又重新成為自動啟動應用程式。一個應用程式在下列情況下，將重新成為自動啟動應用程式：

當接收器瀏覽到當前服務之外並選取了另一個服務，而該應用程式在此服務中也被列為自動啟動程式。

應用程式從 AIT 當中移除，然後又重新加入。

應用程式的自動啟動狀態被清除之後又重新被設定。

註 1：總而言之，應用程式的自動啟動狀態，實際上是屬於邊緣觸發（edge trigger）訊號，而不是程度觸發（level trigger）的訊號。

註 2：這些自動啟動應用程式的控制，能夠避免自動啟動應用程式自行停止之後又被重複啟動。但是，並沒解決接收器將應用程式終止的狀況。在這種情況，平台可以嘗試重新啟動應用程式，但這一點由實做自行決定。

註 3：本規格並未詳細說明廣播訊號在重新設定應用程式自動啟動狀態的時序（timing）要求。

4.1.7 當選台動作不是用來選擇服務時

MHP 應用程式可能會利用選台動作來切換到另一個傳輸串流 (transport stream)，而且使用服務選擇以外的機制。這種情況並不會選擇任何服務，因此 MHP 終端機或 MHP 應用程式不應該啟動目標傳輸串流或服務內的任何應用程式。MHP 終端機應該繼續對邏輯上選擇的服務監控其 AIT (如果存在於目標傳輸串流中)。如果所選取的服務中沒有 AIT，應用程式就根據「5.4.4 Visibility of AIT」一節中的描述繼續執行。應用程式不得使用這些機制來變更服務環境中目前呈現的服務。

DVB-J 選台 API

4.1.8 DVB-J 應用程式和服務選擇

DVB-J 應用程式可以透過服務選擇 API 來選取服務。服務選擇 API 包含一個 `ServiceContext` 類別，來代表用來呈現服務的服務環境。要在服務環境中選擇一個新的服務來呈現，可呼叫 `ServiceContext` 的 `select` 方法，此時該服務環境原來呈現的服務將會停止。

當一個 MHP 應用程式利用「應用程式列舉與啟動 API」來成功啟動另外一個 MHP 應用程式，這第二個 MHP 應用程式應該視為與第一個 MHP 應用程式在相同的服務環境中執行。至於要用多少個 `ServiceContext` 物件來代表一個服務環境，是隨實做而定，但是每一個應用程式都會看到自己的一個服務環境物件。此時，只要任何一個應用程式對其 `ServiceContext` 物件做任何改變，其他應用程式所看到的 `ServiceContext` 物件也會發生相同的變化。

當服務的選擇啟動了一個 DVB-J 應用程式，則該應用程式將視為在服務環境中執行。因此，此應用程式就能利用 `javax.tv.service.selection.ServiceContextFactory` 的 `getServiceContext(XletContext)` 方法獲得其所在服務環境的參照 (reference)。

4.2 DVB-J 模型

4.2.1 啟動 DVB-J 應用程式

一個 DVB-J 應用程式可經由一般 MHP 應用程式的啟動方式來啟動。這組 API 讓一個 MHP 應用程式可以在安全政策規範下啟動另一個 MHP 應用程式。此時 `AppProxy` 介面的 `start()` 方法就會通知應用程式管理員來啟動新的 MHP 應用程式 (在正常的資源限制下)。

`Xlet` 介面定義在 Java TV [51] 的 `javax.tv.xlet.Xlet` 介面。DVB-J 應用程式需提供這個介面的實做，並在 DVB-J 應用程式位置描述器 (location descriptor) 中指定這個實做類別。為了啟動 DVB-J 應用程式，應用程式管理員會呼叫這個類別的建構函式 (constructor)，以及這個介面的 `initXlet()` 和 `startXlet()` 兩個方法。

4.2.2 停止 DVB-J 應用程式

DVB-J 應用程式可以因一般 MHP 應用程式停止的原因而停止。一個應用程式要停止時必須能夠對外通知，這是透過 `javax.tv.xlet.XletContext` 介面的 `notifyDestroyed()` 方法來通知應用程式管理員。此介面也包含一些其他方法，讓 DVB-J 應用程式來要求變更自己的狀態，或是對外通知其狀態變化。

「應用程式列舉與啟動 API」可以讓應用程式間接控制另一個應用程式的生命週期（在安全政策規範下）。這樣的控制是間接的，因為一個應用程式並無法直接呼叫 Xlet 狀態有關的方法，所以才需要透過這組 API。這樣可以確保記憶體中的應用程式管理員能夠隨時追蹤所有正在執行的應用程式。

當一個 DVB-J 應用程式被 MHP 終端機停止時，應用程式管理員就必須呼叫實做 Xlet 介面的 Java 類別中的 `destroyXlet` 方法（此類別就是應用程式初始的類別）。當應用程式因為服務的切換而被迫停止時，應用程式將無條件停止。呼叫這個方法的用意，是讓應用程式可以在停止之前先儲存狀態資料。如果應用程式不希望因為使用者在 MHP 終端機上切來切去（譬如進廣告時）而中斷，就必須在停止前先儲存狀態資料，然後在應用程式被重新啟動時（使用者又切回來），載回先前的狀態。

4.2.3 DVB-J 應用程式生命週期

4.2.3.1 概述

本節說明 DVB-J API 的 Xlet 生命週期模型。我們將說明 Xlet 在每一個狀態下的能力，以及應用程式管理員所用來控制其生命週期狀態的方法。本節與其他章節沒有直接的關聯，譬如繪圖或資源配置與管理。

註：傳統的 Java 平台已經定義了一些應用程式模型，也有自己的生命週期管理規則。一般來說，這些規則只適用於那些平台的特殊問題。譬如，Applet 就是用來支援在網頁中執行應用程式。但是截至目前為止，還沒有一個應用程式技術可以完全解決電視接收器的特殊需求。本節定義的應用程式生命週期，與現有 Java 平台以及虛擬機器 (virtual machine) 技術是相容的。

本節將定義一個 DVB-J 應用程式實例的生命週期。一個 DVB-J 應用程式與其實例的生命週期，基本上是相同的，唯一的差別是，當應用程式實例被摧毀時，應用程式本身只是暫時沒有載入 (not loaded) 而已。

4.2.3.2 DVB-J 應用程式實例的生命週期狀態機器 (state machine)

Xlet 的狀態機器確保一個 Xlet 的行為表現與電視觀賞者所期待的表現一致，也就是：

- 一個 Xlet 啟動的延遲時間要非常短。
- 一個 Xlet 有可能進入沒有提供服務的狀態。
- 一個 Xlet 隨時都可能被摧毀。

圖 4-1 顯示一個 Xlets 的狀態機器模型。更詳細的 Xlet 狀態，則列於表 4-1。

各種可能改變 Xlet 狀態的影響包括：

應用程式管理員使用 Xlet API 來促使 Xlet 狀態改變。

有各種可能的因素會促使應用程式管理員做這樣的動作，例如：

廣播訊號（譬如 AIT 中的 application_control_code 參數狀態改變，請參閱「10.4 Application Information Table」一節）。

使用者從介面上選擇了一個應用程式

Xlet 本身決定改變自己的狀態

一個應用程式必須使用 XletContext 物件來與應用程式管理員溝通或要求變更狀態。

另一個 Xlet 利用應用程式啟動 API 來改變（請參閱「11.7.2 應用程式發掘與啟動 API」一節）。

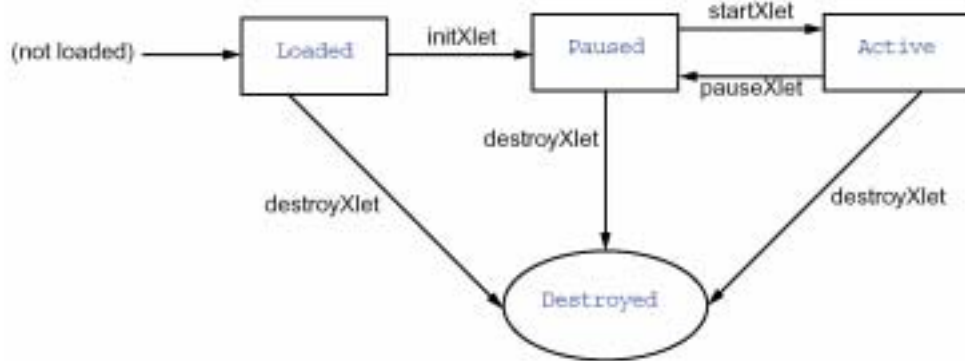


圖 4-1：Xlet 生命週期狀態機器圖

表 4-1：DVB-J 應用程式實例有效的生命週期狀態

狀態名稱	說明
Loaded	DVB-J 應用程式實例已經載入並起始化 (initialized)。

	用來起始 DVB-J 應用程式實例的 Java 類別必須實做 <code>javax.tv.xlet.Xlet</code> 介面。否則，該類別以及該應用程式實例都將被忽略。應用程式管理員將使用 <code>Class.newInstance</code> 方法來建立這個 Java 類別的實例。因此，一個 DVB-J 應用程式實例必須要有一個 <code>public</code> 的預設建構函式 (default constructor)。否則，該類別以及該應用程式實例都將立即進入 <code>Destroyed</code> 狀態。如果預設建構函式返回時沒有產生任何未處理的例外狀況 (uncaught exception)，那麼應用程式實例將進入 <code>Loaded</code> 狀態，否則應用程式將立即進入 <code>Destroyed</code> 狀態並被捨棄。一旦應用程式實例成功載入並起始化，應用程式管理員就能將該應用程式實例轉換到 <code>Paused</code> 狀態 (利用實做 <code>Xlet</code> 介面 Java 類別中的 <code>initXlet</code> 方法)。 此時若 <code>initXlet()</code> 方法產生了一個 <code>XletStateChangeException</code>，那麼應用程式實例將繼續維持在 <code>Loaded</code> 的狀態。該應用程式實例唯一可能轉換的狀態，就是進入 <code>Destroyed</code> 狀態。應用程式實例可以自行要求將狀態轉換成 <code>Destroyed</code> 或是等待應用程式管理員自己來做。 註： 應用程式起始化應該是在 <code>initXlet</code> 方法中執行，而不是在預設建構函式中執行。 每一個 DVB-J 應用程式實例只會進入此狀態一次。
Paused	一個處於 <code>Paused</code> 狀態的 DVB-J 應用程式實例必須盡量減少其資源使用量，才能提升它留在記憶體體的機率。這並不是說它不能保有任何資源，而是在這種情況下，它對資源的存取權限會比處於 <code>Active</code> 狀態時來得低。 在下列情況時將進入此狀態： 從 <code>Loaded</code> 狀態過來時：當應用程式管理員呼叫 <code>Xlet.initXlet()</code> 方法並成功返回時 (只有第一次呼叫)。其他時候呼叫這個方法並不會造成狀態改變。 請注意，應用程式管理員對一個 DVB-J 應用程式實例只能呼叫一次 <code>initXlet()</code>。 從 <code>Active</code> 狀態過來時：當應用程式管理員呼叫 <code>Xlet.pauseXlet()</code> 方法並成功返回時。其他時候呼叫這個方法並不會造成狀態改變。 從 <code>Active</code> 狀態過來時：當進入 <code>XletContext.notifyPaused()</code> 方法時。
Active	此時 DVB-J 應用程式實例正常提供服務。 在下列情況時將進入此狀態： 從 <code>Paused</code> 狀態過來時：當 <code>Xlet.startXlet()</code> 方法成功返回之後。
Destroyed	此時 DVB-J 應用程式實例已釋放其所佔用的資源並結束執行。 在下列情況時將進入此狀態： 當 <code>Xlet</code> 的 <code>destroyXlet()</code> 方法成功返回時。<code>destroyXlet()</code> 方法將釋放所有應用程式所佔用的資源，並且執行必要的清除動作，讓 <code>Xlet</code> 可以被資源回收機制 (garbage collection) 回收。

	在進入 <code>XletContext.notifyDestroyed()</code> 方法時。<code>Xlet</code> 在呼叫 <code>notifyDestroyed()</code> 方法之前，會自行執行清除動作。 註：每一個 DVB-J 應用程式實例只會進入此狀態一次。
--	--

每次一個 DVB-J 應用程式實例啟動時（也就是實做 `Xlet` 類別的建構函式被呼叫時），邏輯上會在自己的虛擬機器實例中執行。請參閱「6.2.1 基本考量」一節。

只有 DVB-J 應用程式自己才知道能不能提供它所設計的服務。因此，在某些方面，應用程式管理員無法保證一個 DVB-J 應用程式是否能夠（或是否正在）提供服務，所以，應用程式管理員只能假定 DVB-J 應用程式有此能力。一個 DVB-J 應用程式典型的執行過程如下：

表 4-2：DVB-J 應用程式生命週期逐步分解

應用程式管理員	DVB-J 應用程式
應用程式管理員產生一個新的 <code>Xlet</code> 實例。	<code>Xlet</code> 的預設（無參數）建構函式被呼叫，然後進入 <code>Loaded</code> 狀態。
應用程式管理員產生必要的服務環境物件，讓 DVB-J 應用程式在這環境中執行，然後將 <code>Xlet</code> 起始化。	DVB-J 應用程式使用服務環境物件來幫自己起始化。現在應用程式進入 <code>Paused</code> 狀態。
當應用程式管理員認為 DVB-J 應用程式應該已經備妥，可以開始提供服務時，就會通知它進入 <code>Active</code> 狀態。	DVB-J 應用程式取得必要的資源，並開始提供服務。
應用程式管理員不再需要 DVB-J 應用程式來提供服務，因此，通知 DVB-J 應用程式停止服務。	DVB-J 應用程式停止服務，同時決定是否釋放一些資源。
應用程式管理員認為不再需要該 DVB-J 應用程式，或是必須騰出記憶體來給更優先順序更高的應用程式使用，因此通知 DVB-J 應用程式將進行摧毀動作。	DVB-J 應用程式此時可以儲存其狀態資料或用戶偏好資訊，並執行清除動作。

4.2.4 Xlet API

`Xlet API` 能提供 MHP 應用程式開發人員一套生命週期通知 API。`Xlet API` 使用回呼（callback）的方式來通知有關狀態的變化。

此 API 定義在「6.7.1 DVB-J 應用程式生命週期支援 API」。

4.2.4.1 Xlet 狀態變化的意義

`Xlet` 的狀態可經由呼叫 `Xlet` 介面方法，或是先自內部改變狀態再利用 `XletContext` 物件

通知應用程式管理員。其中，狀態在何時才真正改變是很重要的：

呼叫 Xlet 介面方法：當呼叫成功返回時，才算狀態改變成功。

呼叫 XletContext：在進入 notifyDestroyed() 和 notifyPaused() 方法時，狀態就算已經改變。而呼叫 resumeRequest() 方法並不會改變狀態，只是請求改變狀態而已。

如果 Xlet 介面方法在呼叫時產生了 XletStateChangeException 例外狀況，則 Xlet 將維持在該方法呼叫前的狀態，除非另有指定。此規則在本規格中的唯一例外情況是使用 destroyXlet 方法來執行無條件摧毀時 (unconditional 旗標設為 true)，此時產生一個 XletStateChangeException 是沒有作用的。在 initXlet 只能呼叫一次的情況，應用程式管理員可以在一段時間之後（由實做決定）將 Xlet 轉換到 Destroyed 狀態（而不呼叫 destroyXlet）。

4.2.4.2 Xlet 狀態變更請求

下表列出在呼叫 XletContext 的方法來執行狀態管理之前的一些有效狀態。

註：這些方法是在 Xlet 已經變更狀態之後才呼叫的。

表 4-3：呼叫狀態管理方法之前的有效狀態

呼叫	狀態
notifyDestroyed	所有狀態
notifyPaused	只能為 active 狀態
resumeRequest	只能為 paused 狀態

當一個 Xlet 在其他狀態時呼叫這些方法將沒有任何作用。

4.2.5 多重應用程式環境支援

DVB-J 平台允許好幾個 DVB-J 應用程式同時執行。

在多個 DVB-J 應用程式可以同時執行的情況下，就必須訂出一些規則，讓 DVB-J 應用程式可以共用 MHP 資源，尤其是輸入焦點 (Input Focus) 與輸出焦點 (Output Focus) 的共用。

4.2.5.1 一個 DVB-J 應用程式控制其他的 DVB-J 應用程式

MHP 提供了一些支援，讓一個 DVB-J 應用程式可以控制另外一個 DVB-J 應用程式的生命週期。

這個功能讓廣播業者可以自行開發「應用程式啟動程式」(Launcher Application)，這樣就能控制如何將目前可用的 DVB-J 應用程式呈現在使用者眼前，並讓使用者啟動 DVB-J 應用程式。請注意，一個 DVB-J 應用程式的生命週期實際上還是掌握在應用程式管理員手中。MHP 只是提供一些 API 讓 DVB-J 應用程式可以要求應用程式管理員幫它啟動、停止、暫停、或是繼續執行 DVB-J 應用程式。

請參閱「6.7.2 應用程式發掘與啟動 API」一節。

4.2.5.2 輸入焦點 (Input Focus) 管理

輸入焦點的定義如下：

擁有輸入焦點的應用程式，原則上會收到使用者的輸入事件 (event)。

其他不擁有輸入焦點的應用程式，則可以利用專門的 API 來接收某一些使用者輸入事件。請參閱有關「org.dvb.event」的說明。

4.2.5.3 其他資源的管理

本規格中定義了一些 API 來提供一些資源配置/收回功能，以及資源收回通知。這些 API 卻沒有定義在何種情況下將允許存取資源，何種情況將收回資源。大多數的情況下，MHP 平台實做對於資源管理都有自己的政策，不過本節還是訂出一些 MHP 平台實做所應遵循的基本規則。

由於 MHP 規格訂定了一個多重應用程式的環境。因此，可能會有許多應用程式同時爭取使用某一個資源。6.7.5 一節所述的資源通知 API 提供了一個共通的方式，讓應用程式在資源存取發生衝突時，可以進行協調。此 API 讓 MHP 終端機可以通知目前佔用資源的應用程式，告知有其他應用程式想要存取其佔用的資源。此 API 也提供一個方式讓資源擁有者和資源要求者有私下溝通的管道。這個私下溝通的管道，就是讓資源要求者可以傳送一個 request_data 物件給資源擁有者。至於，這個物件內容的意義，溝通雙方都必須了解才行。

有些在 MHP 規格制定前就已經存在的通用 Java API 並不使用這種通用的資源共用機制。因此，這些 API 所產生的資源，並無法透過這樣的溝通機制來協調。例如，當一個應用程式佔用了一個 JMF 播放器，如果有另外一個應用程式要產生一個內容類型相同的 JMF 播放器，則 MHP 平台要自行決定是否將 JMF 播放器底層的資源收回，然後提供給新的這個應用程式使用。

當然，應用程式也可以透過應用程式交互通訊 API (inter-application communication API) 來建立私下溝通機制，以協調資源的存取。

4.2.5.4 VM 實做

當一個服務由多個 DVB-J 應用程式所組成時，MHP 終端機可以將這些應用程式在同一個虛擬機器上執行。不管這樣是否符合「6.2.1 基本考量」一節所述。

4.3 DVB-HTML 模型

4.3.1 DVB-HTML 應用程式

4.3.1.1 DVB-HTML 應用程式

一個 DVB-HTML 應用程式，就是從 DVB-HTML 家族元件與內容格式（如本規格所定義）中所選出來的一組文件。該組文件的範圍則是由應用程式的邊界內所定義。

4.3.1.2 使用者代理程式 (User Agent)

一個使用者代理程式，就是用來解譯某種內容格式（此處為 DVB-HTML 文件）的一個應用程式。

註：這可以是一個外掛模組。

4.3.1.3 DVB-HTML Actor

一個 DVB-HTML Actor 就是負責執行 DVB-HTML 應用程式文件的單元，再加上一些資料相關的環境。一個 Actor 必須在使用者代理程式中執行（不論是原生、外掛、或是下載的代理程式）。此 Actor 活動的本質並無法清楚定義，因為，這需視使用者代理程式的本質而定。一個使用者代理程式中可能同時會有多個 Actor 存在。

每一個執行中的 DVB-HTML 應用程式都有一個 DVB-HTML Actor，而一個 DVB-HTML 應用程式是由許多文件所組成，同一時間這些文件可能會有多个同時顯示。

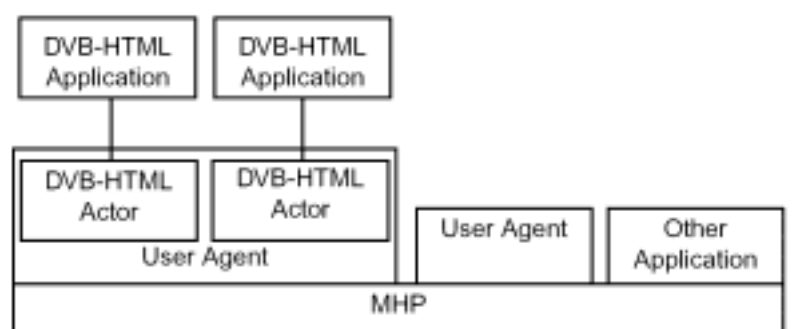


圖 4-2 : Actor 與應用程式的關係

4.3.1.4 應用程式邊界

應用程式邊界定義出一個 DVB-HTML 應用程式的範圍。任何在應用程式邊界之外的文件內容，都不算是該 DVB-HTML 應用程式的一部分，因此 DVB-HTML 應用程式也無法參照到這些內容。MHP 終

端機可以自行開發一套機制，讓終端使用者可以存取此邊界之外的資源，但這並不應放在原始 DVB-HTML 應用程式的環境中。廣播業者可以利用這樣的方式，讓使用者的瀏覽區域更容易控制，而 MHP 平台實做時，也可利用這樣的方式來更有效率地預先抓取 (pre-fetch) 應用程式。

一個 DVB-HTML 應用程式的邏輯範圍可以很大，同時，也可能無法同時全部顯示在 MHP 終端機上。其內容的某些部份可能來自於廣播、某些部分可能放在本機儲存上、某些可能又是在全球網路上，甚至，某些可能是需要時才產生。因此，我們用簡化過的規則表達式 (regular expression) 來定義其範圍。構成一個 DVB-HTML 應用程式的一組文件是採用定址器語言 (locator language) 的規則表達式來定義，大體來說，一個定址器 (locator) 包含一串下列格式的字串 (IETF RFC 2396 [41])，用來組成一個應用程式。

```
scheme://host/dir1/dirn/file#subref
```

一個規則表達式是用一個樣版 (pattern) 來定義出一個集合，這個樣版則能夠測試某一個字串是否在這個集合內，譬如下面這道規則表達式：

```
https?://www\.(dvb|etsi)\.org/[a-z0-9/]+\\.html?
```

就能用來包含位於 www.dvb.org 或 www.etsi.org 的任何檔案 (不論是使用 http 或 https 協定來連接)，而且只能是 DVB-HTML 檔案 (因為其檔名為「.htm」或「.html」結尾)，而且其路徑名稱只能使用英文字母或數字。只要很精簡的一道規則表達式，就能描述一個很大的檔案集合。

4.3.1.4.1 規則表達式語法

一個規則表達式 (RE) 定義了一個集合的字串。屬於這個集合中的字串，都可以吻合這道規則表達式。

為了要能夠吻合邊界定義規則表達式 (boundary regular expression)，整個定址器都必須與整個表達式吻合，不過，從第一個「?」或「#」字元 (含「?」和「#」) 之後的字串，則為參數部分，不屬於定址器本身，因此在做邊界比對時並不列入考慮。

我們用來定義應用程式邊界的規則表達式在 POSIX [59] 第 2.8.4 節中的定義稱為 POSIX Extended Regular Expression。

DVB-HTML 應用程式中的相對定址器 (relative locator) 在比對之前會展開成完整的 URI，如 IETF RFC 2396 [41] 所定義 (其預設的基底 URI 就在應用程式位置描述器中)。

在應用程式邊界描述器中，一個樣版 (pattern) 還能夠切成幾個子樣版 (sub pattern)。而整個的樣版則是將所有子樣版利用 OR 邏輯運算集合起來。每一個應用程式邊界描述器都可以有一個對應的標籤 (label) (請參閱「5.10.3 DVB-HTML application boundary descriptor」)。此標籤可以用在預先抓取 (pre-fetch) 的動作上 (隨傳輸方式而定)，譬如在一個 object carousel 中，所有吻合該標籤的模組都要預先載入。

例如：一個應用程式的進入點頁面為 /phase0/index.html，進去之後分成三個子區域，每一個都有對應的樣式頁 (stylesheet) 和影像目錄。

```
labelA: (/phase0/.+\.html | /phase0/images1/.+\.png | /phase0/scripts1/.+\.js)
labelB: (/phase1/.+\.html | /phase1/images/.+\.png | /phase1/scripts/.+\.js)
labelC: (/phase2/.+\.html | /phase2/images/.+\.png | /phase2/scripts/.+\.js)
```

該應用程式的進入點定址器吻合第一道規則表達式，因此，預先抓取機制就能將標有第一個標籤 (labelA) 的模組載入 (廣播業者用這些模組來放置 phase0 目錄的內容)。一旦使用者代理程式開始執行，就能利用這些資訊，來判斷目前頁面中的連結是否會跳到另一個階段 (phase)，以便進行預先抓取的動作。

4.3.2 DVB-HTML 應用程式生命週期

4.3.2.1 概述

在 DVB-HTML 應用程式生命週期模型中，有三個重要的部分：

- a) 如何讓 MHP 平台知道應用程式已經可用，同時，對於自動啟動和預先抓取的應用程式，開始的時間如何與相關的媒體串流同步。
- b) 應用程式管理員或其他應用程式啟動程式，如何 (以及何時) 讓使用者知道有一個非自動啟動的應用程式可以使用，並提供觸發機制。這部分在「應用程式發掘與啟動機制」中討論。
- c) 一個廣播業者如何控制一個啟動後的 DVB-HTML Actor。

4.3.2.2 訊號通知

DVB-HTML 應用程式訊號通知的細節在第 5 章「Application Signalling」有詳細說明。

要啟動 DVB-HTML 應用程式，必須向應用程式管理員要求，要求的方法，可以是將應用程式設為自動啟動，或是利用應用程式啟動 API。

當應用程式管理員收到啟動 DVB-HTML 應用程式的要求 (也就是在 AIT 中出現 AUTOSTART 或 PREFETCH，或是由使用者發出要求)，若沒有其他應用程式識別碼相同的程式正在執行，就必須嘗試找到一個適合的使用者代理程式。此時，應用程式管理員也可以開始預先抓取資料內容。

此時若資源不足，或是沒有適當的使用者代理程式可用，應用程式管理員可以放棄預先抓取的動作，同時通知訊號可以被忽略。

一個自動啟動的 DVB-HTML 應用程式何時啟動，是隨實做而定的。對於一個已經預先抓取下來的

DVB-HTML 應用程式，就必須有一個觸發動作，讓該應用程式可以開始提供服務。

一個 DVB-HTML Actor 可能會處於五種 DVB-HTML 應用程式狀態：

Loading

Active

Paused

Destroyed

Killed

底下各節將說明這五種狀態的意義。而會促使狀態轉換的觸發動作有：

一個瀏覽新文件的請求

DVB-HTML 應用程式主動要求變更狀態

外在環境的改變，也就是 AIT 中的 application_control_code 有改變

由於一個使用者代理程式可能含有許多個 Actor，因此，同一個時間可能會同時處於許多個狀態，不過，每一個 Actor 會有自己對應的應用程式識別碼 (application ID)。

一個 DVB-HTML 應用程式執行時會在各個文件之間移動，只要這些文件還是在 DVB-HTML 應用程式的邊界內，該 DVB-HTML 應用程式就會繼續執行。

DVB-HTML 應用程式中的連結在開啟時，通常會取代目前的文件，不過也有可能因為屬性設定的關係，新、舊文件會同時顯示。

4.3.2.3 生命週期控制

此處所描述的 DVB-HTML 應用程式生命週期控制狀態模型，反映出通知訊號的運作（請參閱「5.6 Control of application life cycle」），同時也是 DVB-HTML 應用程式運作的一個概觀，同時也考慮到使用者代理程式在正常運作時所需的一些資源：輸出（顯示）、輸入（事件）以及連線（內容）等資源。

不過，此一抽象模型僅供說明而已，並不隱含任何資源管理策略，更沒有限制使用者代理程式實做方式的用意。

4.3.2.3.1 狀態圖

底下狀態轉換圖概略說明 DVB-HTML 應用程式的狀態和轉換流程。

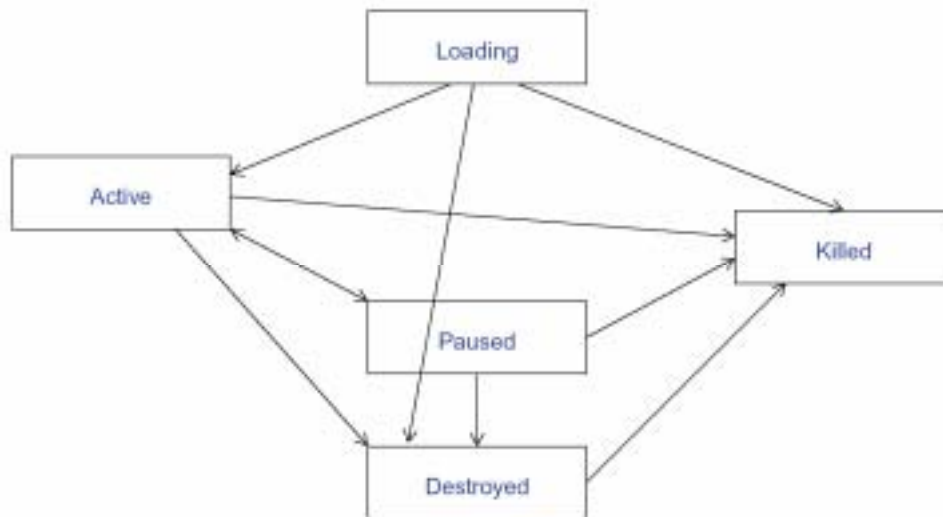


圖 4-3：DVB-HTML 應用程式生命週期狀態圖

4.3.3 狀態模型

該狀態機器的進入點是 Loading 狀態，此狀態是透過存取內容資源 (content resource) 與訊號通知資源 (signalling resource) 而引起，不過，此時並沒有、也不必輸入和輸出資源。這意味著此時 Actor 可以預先抓取內容，並在收到觸發事件時轉換到 Active 狀態，但是使用者目前還看不到。

在進入 Active 狀態時，Actor 應該對當前文件的內容以及所有 MHP 資源有完整的存取權限（當然還是受到資源管理和安全管理政策規範）。

Paused 狀態是半運作的狀態。當應用程式管理員或是使用者代理程式需要更多的資源來做事，Actor 就可能被轉換到 Paused 的狀態，在這個狀態下，Actor 可能沒有完整的（甚至完全沒有）資源存取權限。當 Actor 再度被喚起時，就會回到先前的狀態。

Destroyed 狀態是 Actor 失去內容資源 (content resource) 時的狀態。此時 Actor 可能還是能夠繼續執行 DVB-HTML 應用程式（因為有快取機制），但是還是可能隨時會無法從 DVB-HTML 應用程式內載入文件內容。這種問題的處理方式由實做決定。

這是廣播業者與 MHP 平台切斷關係的一種方法。

Killed 狀態是 Actor 失去所有資源的狀態，同時也代表 Actor 應該被清除的訊號。MHP 平台應該收回所有必要的資源。至於快取中的內容時否應該丟棄，則視平台實做而定。

如果 AIT 通知訊號是 KILL，那麼不管 Actor 正處於何種狀態，都會被強迫終止（其相關資源也全數被收回）。

一個 Actor 在沒有讓 DVB-HTML 應用程式進入 Killed 狀態之前，自己不能停止執行，也不能停止執行 DVB-HTML 應用程式。

4.3.3.1 Loading

4.3.3.1.1 名稱

Loading

4.3.3.1.2 進入動作

產生一個 Actor。

4.3.3.1.3 活動

等待文件可用，然後載入文件，但還不顯示這些文件。

4.3.3.1.4 資源

內容、訊號通知、輸出

4.3.3.1.5 轉換

Active 先決條件：

有足夠的資料可以呈現有意義的內容。

Killed 先決條件：

當 DVB-HTML 應用程式收到 KILL 通知訊號時。

Destroyed 先決條件：

當 DVB-HTML 應用程式收到 DESTROY 通知訊號時。

4.3.3.1.6 備註

這是此狀態機器的進入狀態，此狀態在一個 DVB-HTML Actor 的生命週期中只進入一次。使用者代理程式的任何起始階段也視為此狀態的一部分。當 Actor 處於這個狀態時，是不會顯示任何內容的。此狀態並不是應用程式執行之前 MHP 平台進行預先抓模組的階段，請勿搞混。

4.3.3.2 Active

4.3.3.2.1 名稱

Active

4.3.3.2.2 活動

蒐集並解析 (parse) 當前文件與相關資源、顯示文件、維護顯示的文件。隨時接收外在事件，等待外在觸發動作，來顯示載入的文件。

4.3.3.2.3 進入動作

如果應用程式收到的訊號是 pre-fetch，那就應該等到處發動作才能開始顯示內容。

4.3.3.2.4 資源

內容、訊號通知、輸出、輸入

4.3.3.2.5 轉換

Pause

使用者代理程式或應用程式管理員將 DVB-HTML 應用程式轉換到 PAUSE 狀態。

Destroyed

當 DVB-HTML 應用程式收到 DESTROY 通知訊號時。

Killed

當 DVB-HTML 應用程式收到 KILL 通知訊號時。

4.3.3.2.6 備註

此狀態為穩定狀態。

在此狀態中，使用者代理程式需自行決定是否要顯示未完全載入的文件內容，並處理輸入事件。

當應用程式切換到新的文件時，Actor 仍然維持在這個狀態中。

如果主要文件的相關資源文件改變，那麼，資源可以重新載入，這樣 DVB-HTML Actor 會收到適當的 DOM 事件，不過 DVB-HTML Actor 的狀態還是沒有改變。類似情況，在這個狀態下，DVB-HTML Actor 可能會獲得或失去輸入焦點、收到適當的 DOM 事件、甚至在沒有輸入焦點時也會收到輸入事件。

如果 AIT 不再指向此 DVB-HTML 應用程式，處於 Active 狀態的 DVB-HTML Actor 並不需要做

任何動作。

4.3.3.3 Paused

4.3.3.3.1 名稱

Paused

4.3.3.3.2 活動

DVB-HTML Actor 應該盡量減少資源的使用。

4.3.3.3.3 資源

視應用程式而定。

4.3.3.3.4 轉換

Active

當 DVB-HTML 應用程式繼續執行時。

Destroyed

當 DVB-HTML 應用程式收到 DESTROY 通知訊號時。

Killed

當 DVB-HTML 應用程式收到 KILL 通知訊號時。

4.3.3.3.5 備註

這個狀態的意義是隨使用者代理程式以及 DVB-HTML 應用程式而定。當 DVB-HTML 應用程式從 Paused 狀態返回時，整個環境可能已經改變（已經失去資源和網路連線），而且，有些事件可能也沒有收到。

4.3.3.4 Destroyed

4.3.3.4.1 名稱

Destroyed

4.3.3.4.2 活動

載入文件。顯示文件。接收事件。與使用者互動。

4.3.3.4.3 資源

輸入和輸出。

4.3.3.4.4 轉換

Killed

{如果 DVB-HTML 應用程式收到 KILL 通知訊號} 或 {本機事件強迫 Actor 終止 (可能透過應用程式管理員)}

4.3.3.4.5 備註

此狀態表示 MHP 平台可能再也無法存取 DVB-HTML 應用程式執行所需的內容資源。至於此時 Actor 還應不應該繼續執行，則視 DVB-HTML 應用程式和使用者代理程式而定，如果繼續執行，那就要決定在某個連結無法存取所指到的內容時 (或是快取中的內容也過期時) 應該如何通知使用者。DVB-HTML Actor 可能可以繼續在 Destroyed 狀態下執行，直到使用者將它關閉為止。

4.3.3.5 Killed

4.3.3.5.1 名稱

Killed

4.3.3.5.2 進入動作

釋放資源。

4.3.3.5.3 活動

終止 DVB-HTML 應用程式。

4.3.3.5.4 資源

無。

4.3.3.5.5 轉換

無。

4.3.3.5.6 備註

當此狀態下的所有活動都完成後，應用程式就不再執行。此狀態是本狀態機器的離開狀態。

4.4 跨應用程式資源管理

除了下面說明的部分之外，其餘的視實做而定。當同一個服務內的兩個應用程式在同一個服務環境中同時執行時，若遇到資源使用衝突，則必須利用每個應用程式描述器 (application descriptor) 中所指定的優先順序 (application_priority 欄位) 來決定。取捨時，資源應該保留給 application_priority 較高的應用程式。

當同一個服務內的兩個應用程式在同一個服務環境中同時執行時，若遇到資源使用衝突，則必須利用每個應用程式描述器 (application descriptor) 中所指定的優先順序 (application_priority 欄位) 或是外部應用程式授權描述器 (external application authorisation descriptor) 來決定，看目前是採用哪一個描述器。

除非另有指定，只有擁有資源的應用程式有權限可以修改資源的狀態 / 設定 / 組態。如果一個記憶體中的應用程式 (譬如 MHP 瀏覽器) 改變了某個資源的狀態 / 設定 / 組態，MHP 終端機應該通知原先擁有該資源的 MHP 應用程式，讓該程式知道已經失去資源的所有權。

註：本規則的唯一例外是服務環境的 service component handler，此資源是所有在服務環境中執行的應用程式所共同擁有，並非某一個程式所有。請參閱「11. 6.2 服務選擇 API」一節。

5 應用程式信號(Application Signaling)

5.1 簡介(Introduction)

此節涵蓋下列的主題：

- 接收機如何確認應用程式是屬於某一服務，並找尋其位置進而取回它們。
- 信號可以讓廣播系統管理應用程式的存在週期。
- 接收機如何能確認某一服務的應用程式所需的廣播資料來源。

多數的信號(signalling)是個廣泛性的。例如，應用程式描述(Application descriptor)是不相依於應用程式的表示法。其餘的信號(signaling)是指定應用程式表示法或指定傳輸協定(就像 DVB-J application descriptor 和 IP signalling descriptor)。

5.1.1 一般信號概述(Summary of common signaling)

對於任何 MHP 應用程式的信號需求，其最小概述如下：

- PMT 附有應用程式信號描述(Application Signaling Descriptor)，作為確認帶有應用程式資訊表(Application Information Table)服務的元件。
- 應用程式資訊表(Application Information Table)，在其一般描述迴圈中含有下列資訊：
 - 傳輸協定描述(Transport protocol descriptor)(在所有應用程式描述範圍之內，必須包含至少一個傳輸協定描述(Transport protocol descriptor)。這些可以被放置在一般(common)或應用程式(application)任一描述迴圈中，也可在兩者的描述迴圈中都出現。)
- 應用程式資訊表(Application Information Table)，在其應用程式資訊描述迴圈中含有下列資訊：
 - 應用程式描述(Application descriptor)。
 - 應用程式名稱描述(Application name descriptor)。

5.1.2 DVB-J 應用程式附加信號概述

(Summary of additional signalling for DVB-J applications)

DVB-J 應用程式附加信號(signaling)最小的需求可摘要如下：

- 應用程式資訊表(Application Information Table)，在其應用程式資訊描述迴圈中有下列資訊：
 - DVB-J 應用程式描述(DVB-J application descriptor)。
 - DVB-J 應用程式位置描述(DVB-J application location descriptor)。

5.1.3 DVB-HTML 應用程式附加信號概述

(Summary of additional signalling for DVB-HTML applications)

最小的 DVB-HTML 應用程式附加信號需求可摘要如下：

- 應用程式資訊表(Application Information Table)，在其應用程式資訊描述迴圈中含有下列資訊：
 - DVB-HTML 應用程式描述(DVB-HTML application descriptor)。
 - DVB-HTML 應用程式位置描述(DVB-HTML application location descriptor)。

5.1.4 由物件連續撥放攜帶的應用程式附加信號概述

(Summary of additional signalling for application carried via OC)

在“一般(common)” (最先的)描述迴圈或“應用程式”(內部的)描述迴圈中：

- 傳輸協定描述(Transport protocol descriptor)，含有的選擇位元組包含了物件連續撥放特定的資訊，定義在表 5-20。

5.1.5 由 IP 攜帶的應用程式附加信號概述

(Summary of additional signalling for applicatoin carried via IP)

應用程式資訊表(Application Information Table) 其一般描述迴圈中含有下列資訊：

- IP 信號描述(IP signalling descriptor)。

在“一般”(最先的)描述迴圈或“應用程式”(內部的)描述迴圈中：

- 傳輸協定描述(Transport protocol descriptor)，含有的選擇位元組包含了 IP 特定的資訊，定義在表 5-21。

5.1.6 如何加入新的結構(非正式的) (How to add a new scheme—informative)

信號結構著重在考慮應用程式表示和傳輸協定支援的延展性。當處理此部分時需注意如下概述：

加入額外的傳輸協定：

- 記錄在表 5-19，“選擇位元組的表示意義”。
- 盡可能的定義額外的專門描述，比如 IP 信號描述。

加入額外的應用程式表示：

- 定義額外的專門描述，如同 5.9，“DVB-J 特定描述(DVB-J Specific descriptors)”。
- 在 5.6 中定義了應用程式型態指定的存在週期控制碼，“控制的應用程式存在週期(Control of application life cycle)”。

本規範所註冊的常數值提供在表 5-31，“註冊的常數值”。

5.1.7 服務資訊(Service information)

參照 5.12，“服務資訊(Service Information)”。

5.2 程式指定資訊(Program Specific Information)

為了使 DVB 服務支援一個至多個 MHP 應用程式，PMT 的基本流(內部的)迴圈必須參考下列形式的流：

- 傳輸應用程式資訊表(Application Information Table) 之流的位置。
- 傳輸應用程式碼和資料之流的位置。

5.2.1 應用程式信號流(Application signalling stream)

PMT 記錄的基本流資訊，其所描述基本流攜帶的應用程式資訊表(Application Information Table)有以下的特性：

- stream_type 設為 0x05(ITU-Trec.H.222.0|ISO/IEC 13818-1 private sections)。

- 一個應用程式信號描述(Application Signaling Descriptor)。

一項服務中可能有一個至多個基本流攜帶應用程式信號資訊。

5.2.2 資料廣播流(Data broadcast streams)

為了傳輸協定的需求，PMT 信號最小必須結合的資料廣播元件是 PMT 中的 stream_type 值域，這是為了傳輸協定在 DVB 資料廣播規範(EN 301 192[5])提出的。資料廣播傳輸的完整細節：“首要的(principal)”元件的位置等，由 AIT(見 5.4, “[應用程式資訊表\(Application Information Table\)](#)”)提供。

PMT 可以選擇性地決定是否包含資料廣播確認描述(Data broadcast id descriptors)。

註：

包括資料廣播確認描述(Data broadcast id descriptors)能夠使接收機啟動架設檔案系統，以傳遞應用程式並同時取得 AIT，使其確認哪一個應用程式是必要的。可使這同時發生的操作允許接收機加速它們的作用於互動應用程式的反應。

資料廣播確認描述(Data broadcast id descriptor)確認資料廣播的“重要”“principal”元件。此可選擇的訊號反映出傳輸協定的詳細意義。例如，DVB 物件連續撥放(DVB Object Carousel)的例子它確認傳遞 DSI 的元件。

可能也有某種協定特有的描述在 PMT 中。例如，物件連續撥放需要包含 carousel_id_descriptor。

(不含選擇的資訊) 資料廣播確認描述 Data broadcast id descriptor 必須確認“重要”“principal”元件的存在。這種型式可以與選擇器資訊一起被任意的擴張，而選擇器資訊是被資料廣播所傳遞的，用以確認應用程式類型的自動開始應用程式。

參考 5.7.2, “資料廣播確認描述” “Data broadcast id descriptor”。

5.3 標誌(Notation)

5.3.1 保留的(reserved)

當使用在定義編碼過的位元流款項中，“reserved”項目指示這個值也許未來將被 ISO 定義的延伸使用。除非其他方面具體指定的，否則在本章節所有的“reserved”位元應被設為“1”。

5.3.2 未來保留使用(reserved_future_use)

當使用在定義編碼過的位元流款項中，“reserved_future_use”項目指示這個值也許未來將被 ETSI 定義的延伸使用。除非其他方面具體指定的，否則在本章節所有的

“ reserved_future_use ” 位元應被設為 “ 1 ”。

5.4 應用程式資訊表(Application Information Table)

應用程式資訊表(AIT)提供在資料廣播上的全部的資訊，及被它傳遞的應用程式需求的反應狀態等。AIT 允許廣播者要求接收機改變應用程式的反應狀態。每一個 AIT 次表最小的重複率是 10 秒。

5.4.1 資料錯誤(Data errors)

AITs 包括的錯誤處理如下：

一個錯誤描述將導致描述默許丟棄。處理此描述迴圈應繼續處理下一個描述（若有的話）。描述的錯誤範圍應限制在所伴隨的應用程式資訊片段。

在應用程式迴圈外的描述錯誤應導致此應用程式迴圈項默許丟棄。處理此應用迴圈應繼續下一個應用迴圈項（若有的話）。

標記：上述的結果表示當錯誤在代理描述中將導致描述默許忽略，然後導致應用程式迴圈遺失此一代理描述。即此一應用程式迴圈也應被默許忽略。

在應用程式迴圈的應用程式資訊片段錯誤將導致整個應用程式資訊片段默許丟棄。AIT 的處理方式將繼續下一個應用程式資訊片段（若有的話）。

5.4.2 AIT 的傳送與監控(AIT transmission and monitoring)

MHP 終端機應監控 AIT 基本流中 PMT 的編號變化。其變化應在一秒內被測出來。

AIT 次要表的最低重複速率為 10 秒。AIT 應伴隨不混亂的基本流。

提供 AITs 被選擇的服務是被傳遞在 3 或者比在 AIT 被更新的時刻和新版本被 MHP 偵測到之間的最大時間區間較少的基本串列應該不會超過 30 秒。

註：如果廣播使用超過 3 基本串列來傳遞 AITs，然後接收機反應時間將會以不可預測的方式降低。

MHP 終端只需監控能解碼為應用程式型態的 AIT 片段。應用程式型態列在應用程式資料庫反映出可監測的 AIT 片段。因此，應用程式型態僅為廣播伴隨主集中的次要集合形式。

5.4.3 最佳化的 AIT 信號(Optimized AIT signalling)

任意的 AIT_version_number 被 Application signaling Descriptor 攜帶，允許可能的最佳化的接收機負擔，如同它允許接收機只在它們看見 PMT 中所建議的 AIT 版本改變之後取得 AIT。

參考 5.7.1，“Application Descriptor”。

5.4.4 AIT 的可見性(Visibility of AIT)

假設應用程式調離傳訊串列它 signaling 被攜帶沒有選擇新服務的地方，它將繼續執行雖然 AIT 是不可見的。MHP 終端有多種的網路介面，若 AIT 經由這些介面所選擇的服務是可見的，則 AIT 訊號便可正常的使用。

5.4.5 AIT 定義的次表(Definition of sub-table for the AIT)

在相同 PID 有相同 AIT table_id 和相同 application_type 的值的所有的片段是相同次表的一部份。

5.4.6 AIT 的語法(Syntax of the AIT)

應用程式資訊片段敘述了應用程式及其相關資訊。每一個應用程式資訊片段之最上層描述包含一個 “common” 描述迴圈，其共享於應用程式次表與應用程式迴圈。每一個應用程式迴圈中的應用程式均有 “application” 描述迴圈，其包含了應用程式相關描述。

像 DVB SI 的表一樣，common 迴圈描述是次表。因此，任何描述出現在 common 描述迴圈將應用到所有的節的次表。典型的，除非沒有足夠的空間，common 描述將按慣例的只被呈現在節 0 的次表。像其他的 DVB SI 表，任何字串包含在這些表中不應該有空的字尾。

表 5-1: 應用程式資訊區段語法(頁 1/全 2 頁)

	No.of Bits	Identifier
application_information_section() {		
table_id	8	uimsbf
section_syntax_indicator	1	bslbf
reserved_future_use	1	bslbf
reserved	2	bslbf
section_length	12	uimsbf
test_application_flag	1	bslbf
application_type	15	uimsbf
reserved	2	bslbf

表 5-1: 應用程式資訊區段語法(頁 2/全 2 頁)

	No.of Bits	Identifier
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
reserved_future_use	4	bslbf
common_descriptors_length	12	uimsbf
for(i=0;i<N;i++){ descriptor() }		
reserved_future_use	4	bslbf
application_loop_length	12	uimsbf
for(ia0;i<N;i++){ application_identifier() application_control_code reserved_future_use application_descriptors_loop_length	8 4 12	uimsbf bslbf uimsbf
for(j=0;j<N;j++){ descriptor() } }		
CRC_32	32	rpchbf

table_id : 8 bit 整數，確認這種表的值是 0x74。

section_syntax_indicator : section_syntax_indicator 是 1 bit 應該被設為 1。

section_length : 這是 12 bit，前兩個 bit 應被設為 '00'。剩下的 10 bits 說明表示的數值是從節一開始到接下來的節的長度以 bytes 表示，並包括 CRC_32。這值在這裡不應超過 1021(0x3FD)。

test_application_flag : 此為 1bit，有設置則表示此應用程式的傳送是為了做接收端測試，並且在正常操作情況下，均不應該啟始或列出 API 或任何 MHP 接收端的使用者介面。若有值便表示 MHP 接收端處於某一模式，可偵測此應用程式訊號的位元設定。若設為 0 表示不應該發現任何的典型的終端使用者。

application_type : 16 bit，用以確認應用程式描述在 AIT 次表中的類別。

表 5-2: 應用程式型式

application_type	description
0x0000	reserved_future_use
0x0001	DVB-J application
0x0002	DVB-HTML application
0x0003...0x7FFF	subject to registration with DVB

version_number : 5 bit 是次表的版本值。當次表攜帶的資料發生改變時，version_number 應該被增加 1。當它的值達到 "31"，後又回復到 "0"。當 current_next_indicator 被設為 "1"，

然後 version_number 應該是現在的應用程式次表。當 current_next_indicator 被設為 "0" , 然後 version_number 應該是下一個應用程式次表。

current_next_indicator : 1 bit 的表示, 應該設置為 "1" 。

section_number : 8-bit 給這節的值。section_number 的第一節在次表中應被設為 "0x00" 。 section_number 應被增加 1 在每一個額外的節有相同的 table_id 和 application_type。

last_section_number : 8 bit 是子節中最後的節的特定的值(就是, 有最高的 section_number 的節)。

common_descriptors_length : 12 bit 以 byte 表示接下來的描述的全部的長度。這描述在描述迴圈中供應給包括在這個 AIT 次表中的所有的應用程式。

application_control_code : 8 bit 是控制應用程式的狀態。這部分的語法與應用程式類型無關。參考 5.6 , "Control of application life cycle" 。

application_loop_length : 12 bit 給定所有的長度, 以 bytes 表示接下來的迴圈包括應用程式資訊。

application_identifier() : 48 bit 用以確認應用程式。這範圍的架構被定義在 5.5.5 , "Application identification" 。

application_descriptors_loop_length : 12 bit 給定所有的長度以 byte 表示接下來的描述。這描述在此迴圈供應給特定的應用程式。

CRC_32 : 32 bit 是包括 CRC 的值, 在處理過所有節之後在解碼器給一個 0 輸出的暫存器的定義在 EN 300 468 的附錄 B。

5.4.7 在 AIT 中使用私有的描述(Use of private descriptors in the AIT)

私有的描述可以被包含在 AIT 中提供給 DVB-SI[4]私有資料特有的描述範圍。這私有資料特有的描述範圍的規則如下列：

．假設這描述是被放置在 AIT 的任何描述迴圈中, 則任何特有的確認在這描述迴圈中提供給所有接下來的描述和使用者定義的值, 在獨特的描述迴圈直到描述迴圈結束, 或者直到另一個 private_data_specifier_descriptor 發生。

．描述的使用在 common(第一) 描述迴圈中不提供給應用程式(第二)描述迴圈中的描述或使用者定義的值。

5.4.8 AIT 的內文編碼(Text encoding in AIT)

除非有特別指定,否則在 AIT 所有的轉譯文字字串,均應以 UTF8 編碼 (參照 " Monomedia format for text)或(" Text encoding of application identifiers ")

5.5 應用程式的識別(Application identification)

5.5.1 編碼(Encoding)

每個應用程式是被關聯到應用程式識別。這是一個由 6 byte 的字串其結構如下：

表 5-3: 應用程式識別語法

	No.of Bits	Identifier	Value
<code>application_identifier { organisation_id application_id }</code>	32 16	bslbf bslbf	

organization_id : 32bit 是全域的獨特的值用以確認為這應用程式負責的組織。這些值被註冊在 ETR 162[10]。此值不應該編碼為 0。

此值被重新產生在組織名稱裡已被授權的主題名稱，名為“ leaf ”之認證應用。

標記：此項包含的驗證提供驗證值。

application_id : 16 bit ，用以確認獨特地應用程式函式。這位置被決定在位置政策的組織以 organization_id 註冊。當 application_id 的值為 0xFFFFh 此應用程式以對應所有的應用程式之相同的 organization_id。此值不應該編碼為 0。

把application id 的值分成兩個範圍：一個為 unsigned ，一個為 signed。 這是由於安全原因。 當 Applications 傳送時若為 unsigned則對應 unsigned 的範圍,當傳送時若為 signed 則對應 signed 的範圍。

表 5-4: application_id值的範圍

application_id values	Use
0x0000...0x3fff	Application_ids for unsigned applications
0x4000...0x7fff	Application_ids for signed applications
0x8000...0xffff	Reserved for future use by DVB
0xfffe	Special wildcard value for signed applications of an organisation
0xffff	Special wildcard value for all application of an organisation

Application id 的值為 0xffff and 0xfffe 時是廣泛的卡片。將不用他們來確定應用，但是考慮到用於 External application authorization descriptor，參見5.7.5。當值為 0xffff 時，

所有 applications 匹配於相同的 organisation_id。當值為 0xffff 時所有 signed applications 匹配於相同的 organisation_id。

相同的應用識別碼可能被使用在不同的應用模式以執行相同的函式。

相同的應用識別碼應該會出現在 AIT 的次表裡的應用服務。

5.5.2 存在週期的影響(Effects on life cycle)

這裡主要的概念是：

- 當服務改變時，現在執行先前的廣播，service_bound_flag需設為"0"，應用程式被允許(主題的資源限制)繼續執行，假設它們的應用程式識別碼是列出在Application Information Table的最新選擇的服務。
- 當服務改變時，現在執行先前的廣播，應用程式被允許(主題的資源限制)繼續執行，假設它們的應用程式識別碼是符合列出地 External application authorization descriptor，甚至假設它們不是現在服務的一部份。
- 只有單一的情況的應用程式有獨特的應用程式識別碼是被允許可以任意時間執行。所以假設應用程式是已經執行，然後，另一個情況的相同的應用程式不應被啟動。這影響應用程式啟動 API 秩序和在選擇服務後自動啟動應用程式的行為。
- 如果當 service_bound_flag 被設作"1" 時，在上個服務選擇時，此應用程式將會被刪除。

5.5.3 鑑定應用程式識別 (Authentication of application identification)

5.6 控制的應用程式存在週期(Control of application life cycle)

廣播 signaling 提供一種機制給廣播者控制標準應用程式類別的存在週期。。

5.6.1 進入和離開應用程式的領域

(Entering and leaving the domain of an application)

應用程式的區域是被定義成一連串的服務，這應用程式是被列在 AIT 中。它可以被當作應用程式列在AIT的應用程式(內部)迴圈中或者當成應用程式列在 External application authorization descriptor。應用程式沒列出的，而以這兩種方式的服務，是為超出應用程式的領域。

5.6.2 應用程式存在週期的動態控制

(Dynamic control of the application life cycle)

動態控制的應用程式存在週期是被 signaling 透過 application_control_code 給 AIT 的應用程式。

這控制碼允許廣播者發信號給接收機怎麼去處理應用程式並考慮它的存在週期。這系列的碼在不同的應用程式之間有所不同而且這些被定義在應用程式類別特定的準則。假使接收者接收了一個不被辨識的碼，應用程式應繼續在它現有的狀態。

如果接收者接收了一個碼而每有被認知，應用程式應繼續在它現有的狀態。

當改變一個控制碼時將會導致執行 MHP 的應用程式也跟著改變

AppStateChangeEvent 將被產生到所有 DVB-J 已註冊給接收這樣的事件所涉及到的應用程式。

5.6.2.1 DVB-J

應用程式控制碼給 DVB-J 應用程式使用的被列在表 5-5。

表 5-5: DVB-J 應用程式控制碼的值

code	identifier	semantics
0x00		reserved_future_use
0x01	AUTOSTART	The file system element(s) (e.g. an Object Carousel module) containing the class implementing the Xlet interface is loaded, The class implementing the Xlet is loaded into the VM and an Xlet object is instantiated, and the application is started subject to usual restrictions, etc.
0x02	PRESENT	Indicates that the application is present in the service, but is not autostarted.
0x03	DESTROY	When the control code changes from AUTOSTART or PRESENT to DESTROY, the destroy method of the Xlet is called (with the unconditional parameter set to false) by the application manager and the application is allowed to destroy itself gracefully.
0x04	KILL	When the control code changes from AUTOSTART or PRESENT to KILL, the destroy method of the Xlet is called (with the unconditional parameter set to true) by the application manager.
0x05		reserved_future_use
0x06	REMOTE	This identifies a remote application that is only launchable after service selection.
0x07...0xFF		reserved_future_use

參考 4.2.4 , “ DVB-J Application Lifecycle ”。

5.6.2.2 DVB-HTML

應用程式控制碼給 DVB-HTML 應用程式使用的被列在表 5-6。

表 5-6: DVB-HTML應用程式控制碼的值

code	identifier	semantics
0x00		reserved_future_use
0x01	AUTOSTART	The Application Entry Point of the DVB-HTML application is loaded. This is loaded into the user agent, and the DVB-HTML actor is created (in the Loading state) and the DVB-HTML application is started. When these steps are complete the DVB-HTML actor is in the Active state.
0x02	PRESENT	Indicates that the DVB-HTML application is present in the service, but is not autostarted.
0x03	DESTROY	When the control code changes from AUTOSTART or PRESENT to DESTROY, the DVB-HTML actor goes to the Destroyed state.
0x04	KILL	When the control code changes from AUTOSTART or PRESENT or DESTROY to KILL the DVB-HTML actor goes to the Killed state.
0x05	PREFETCH	As for AUTOSTART except that the DVB-HTML actor holds on entry to the Active state and waits for a trigger before completely transitioning to the Active state.
0x06	REMOTE	This identifies a remote application that is only launchable after service selection.
0x07...0xFF		reserved_future_use

參考，“DVB-HTML Application Lifecycle”。

5.7 一般的描述(Generic descriptors)

5.7.1 應用程式信號描述(Application Signalling Descriptor)

應用程式訊號描述是被定義給使用在基本串列迴圈的 PMT 其基本串列的 stream_type 是 0x05。它定義在基本串列攜帶 Application Information Table。應用程式信號描述選擇性的地攜帶 application_type 和 version_number 成對的迴圈。這些允許描述任意地重新產生相關的 Application Information Table 現在版本數值狀態。這允許接收機被通知版本的 AIT 當作監測 PMT 地邊際效應(它被預期是在正常的狀況下非常接近的監測)。參考“Optimised AIT signaling”。

當MHP偵測到應用程式signaling描述內容的改變，它應取得新的版本的AIT和照著回覆。application_type and AIT_version子域是選擇性的出現。假使出現則Default AIT監測起作用。

表 5-7: 應用程式信號描述語法

	No. of Bits	Identifier
<code>application_signalling_descriptor() {</code>		
<code>descriptor_tag</code>	8	uimsbf
<code>descriptor_length</code>	8	uimsbf
<code>for(i=0; i<N; i++){</code>		
<code>application_type</code>	16	uimsbf
<code>reserved_future_use</code>	3	bslbf
<code>AIT_version_number</code>	5	uimsbf
<code>}</code>		
<code>}</code>		

`descriptor_tag` : 8 bit 整數值 0x6F 確認這描述。

`descriptor_length` : 8 bit 指示位元組的數值指示接下來的描述的長度。

`application_type` : 16 bit 用以確認應用程式類別在這個基本串列的次表的 Application Information Table。

`AIT_version_number` : 5 bit 提供 “current” 版本值被應用程式類別識別的 Application Information Table 次表。

5.7.2 資料廣播識別碼描述(Data broadcast id descriptor)

資料廣播 id 描述是定義在使用基本串列資訊的 PMT。這描述識別：

- ．傳輸格式的資料廣播的 “principal component” 是在這個基本串列。

這 “principal component” 的語法是在基本串列特有的。

- ．一系列的應用程式類別給任何經由資料廣播傳遞的自動啟動應用程式。

對單一基本流可使用多於一個的資料廣播 id 描述，列出附加的應用程式型態，然而每一個描述應該指向相同的資料廣播 id。

超過一個基本流可以有資料廣播 id 描述指向自動啟動應用程式而這自動啟動應用程式是經由不只一個傳遞機制來傳遞(例如，單一服務有不只一個物件連續撥放用以傳遞自動開始應用程式)。

5.7.2.1 通用的描述(Generic descriptor)

資料廣播識別碼描述是由 DVBSI-DAT 規範(描述在表 5-5)定義在通用的型式。那裡沒有 ‘id specific data’ 被提供給描述用來識別 ‘principal’ 元件的資料廣播。

表 5-8: 通用的資料廣播識別碼描述語法

	No.of Bits	Identifier	Value
<code>data_broadcast_id_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	
<code>descriptor_length</code>	8	uimsbf	
<code>data_broadcast_id</code>	16	uimsbf	
<code>for (i=0; i<N; i++) {</code>			
<code>id specific data</code>	8	bslbf	
<code>}</code>			
<code>}</code>			

5.7.2.2 MHP 資料廣播識別碼描述(MHP data broadcast id descriptor)

當資料廣播 id 是這份詳細規格說明(參考 表 5-29)的其中一個定義。其資料廣播 id 描述的語意表示成表 5-9 此延伸的一般性描述選擇性的列在應用程式型態其自動開始應用程式可存在在資料廣播中。此表提供的提示允許 MHP 終端當提供眾多服務時，能優先考慮資料廣播。若無列出則資料廣播 id 描述在資料廣播攜帶的自動開始應用程式型態是靜止的。若應用程式列表不是空的，則資料廣播不應包括應用程式型態中自動應用程式以外的列表。在此列表裡的所有型態，資料廣播不需要總是包括自動開始應用程式。

表 5-9: MHP資料廣播識別碼描述語法

	No.of Bits	Identifier	Value
<code>data_broadcast_id_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	
<code>descriptor_length</code>	8	uimsbf	
<code>data_broadcast_id</code>	16	uimsbf	
<code>for (i=0; i<N; i++) {</code>			
<code>application_type</code>	16	uimsbf	
<code>}</code>			
<code>}</code>			

`descriptor_tag` : 8 bit 整數值 0x66 確認這描述。

`data_broadcast_id` : 16 bit 指示資料傳輸協定的格式。這些值是被註冊在 ETR 162。

`application_type` : 16 bit 指示應用程式類別(也就是說，引擎或插入在應用程式可以被執行)參考表 5-2。

5.7.3 應用程式描述(Application descriptor)

恰好一個應用程式描述例子應被包含在每一個“application”(inner)描述迴圈的AIT。

表 5-10: 應用程式描述語法

	No.of Bits	Identifier	Value
<code>application_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimbsf	
<code>descriptor_length</code>	8	uimbsf	
<code>application_profiles_length</code>	8	uimbsf	
<code>for(i=0; i<N; i++) {</code>			
<code>application_profile</code>	16	uimbsf	
<code>version.major</code>	8	uimbsf	
<code>version.minor</code>	8	uimbsf	
<code>version.micro</code>	8	uimbsf	
<code>}</code>			
<code>service_bound_flag</code>	1	bitbf	
<code>visibility</code>	2	bitbf	
<code>reserved_future_use</code>	5	bitbf	
<code>application_priority</code>	8	uimbsf	
<code>for(i=0; i<N; i++) {</code>			
<code>transport_protocol_label</code>	8	uimbsf	
<code>}</code>			
<code>}</code>			

`descriptor_tag` : 8 bit 整數值 0x0 確認這描述。

`application_profiles_length` : 8 bit 指示以位元組表示 `application_profile` 迴圈的長度。

`application_profile` : 16 bit 是整數值代表應用程式類別特有的 profile。這只是接收者實現一個列在這迴圈的 profile 是有能力執行的應用程式。

`version.major` : 8 bit 攜帶主要子域的 profile 版本號碼的值。

`version.minor` : 8 bit 攜帶次要子域的 profile 版本號碼的值。

`version.micro` : 8 bit 攜帶微小子域的 profile 版本號碼的值。

四個上述的區域指示最小的 profile 在應用程式上將會執行。應用程式可以測試給架構發現在更高的(向下相容)profile 和利用它們。MHP 終端機可以啟動接下來的表示是應用程式至少一個 signaled profile :

```

(application_profile ∈ terminal_profiles_set)
  ∧ {(application_version.major < terminal_version.major(application_profile))
    ∨ [(application_version.major = terminal_version.major(application_profile))
      ∧ ({application_version.minor < terminal_version.minor(application_profile)}
        ∨ [{application_version.minor = terminal_version.minor(application_profile)}
          ∧ [application_version.micro ≤ terminal_version.micro(application_profile)]])]}

```

Where:

- ∈ represents "belongs to the set of"
- ^ represents "logical AND"
- ∨ represents "logical OR"

`service_bound_flag` : 假設這個區域是被設為 1，應用程式是與現在服務相關而且處理將殺除的應用程式，應開始在服務不注意的改變內容的 AIT 位置。

visibility : 2 bit 說明應用程式適合性的意思，給可能的顯示到使用者可能的啟動。表 5-11 列出這區域允許的值。

表 5-11: 應用程式可視狀態的定義

visibility	description
00	This application shall not be visible either to applications via an application listing API or to users via the navigator with the exception of any error reporting or logging facility, etc.
01	This application shall not be visible to users but shall be visible to applications via an application listing API.
10	reserved_future_use
11	This application can be visible to users and shall be visible to applications via the application listing API.

註：

例如當一個服務提供一個遊戲數字給使用者時，這些值將會被使用下列情形：

- 00 - 為自動開始一般的啟動器來提供給使用者來選擇哪個遊戲去啟動。
- 11 - 讓使用者去選擇中間的啟動。
- 01 - 為常見服務的應用適用於所有遊戲被使用在互動頻道的服務上。

application_priority : 這區域識別在這服務中的應用程式 signalled 相關的優先權。

- 不只一個應用程式有相同的 Application identification 這優先權應被使用來決定哪一個應用程式被啟動。
- 有足夠的資源繼續執行一連串的應用程式，這優先權應被使用來決定哪一個應用程式暫停或停止。
- 較大的整數值代表優先權較高。
- 如果兩個相同的應用程式的驗證則會有相同的優先權，所以 MHP 可以隨著實作去選擇開始。

transport_protocol_label : 8 bit 為識別傳遞應用程式的一種傳輸協定。參考 transport_protocol_label 在 Transport protocol descriptor。

如果超過一個協定而被當作信號時，每個協定是為可改變傳遞機器，其順序指示出廣播者的觀點將會提供最好的經驗(第一個為最好)。這個只會在應用程式的存在週期裡被評估。

當協定被 MHP terminal 選擇時，可能會依賴著很多因子包括使用者的偏好跟效能當傳輸連結到終端機。

5.7.4 使用者資訊描述(User information descriptors)

使用者資訊描述補充“Application descriptor”提供符合顯示到使用者的資訊(而“Application descriptor”提供資訊技術給接收者自動的使用)。

這些描述是定義給使用在應用程式迴圈的 AIT。

5.7.4.1 應用程式名稱描述(Application name descriptor)

這描述的正确的一個例子應該被包含在一個應用程式的應用程式資訊。應用程式名稱應區別應用程式和應被提供給使用者。

表 5-12: 應用程式名稱描述語法

	No.of Bits	Identifier	Value
<pre>application_name_descriptor() { descriptor_tag descriptor_length for (i=0; i<N; i++) { ISO_639_language_code application_name_length for (i=0; i<N; i++) { application_name_char } } }</pre>	8 8 24 8 8	uimsbf uimsbf bslbf uimsbf uimsbf	

descriptor_tag : 8 bit 整數值 0x01 用以識別這描述。

ISO_639_language_code : 24 bit 包括 ISO 639.2[19]三字元語言碼的語言，接著 bouquet name。ISO 8859[20]和 ISO 639.2/T 都可以被使用。每一個字元是被編碼成 8 bit 根據 ISO 8859[20]和依次序插入成 24-bit。

application_name_length: 8 bit 沒有符號的整數已位元表示 application name。

application_name_char : 這部分攜帶字串(非空的結束)的字元編碼與 ETS 300 468 的附錄一致。應用程式的字串名稱以一種方式提供給使用者。

5.7.4 應用程式圖像描述(Application icons descriptor)

這描述的零或一的例子應被包含在應用程式資訊的應用程式。它允許圖像被關聯到應用程式。這些可能的圖像的內容的格式應被限制 PNG。

表 5-13: 應用程式圖像描述語法

	No. of Bits	Identifier	Value
<code>application_icons_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	
<code>descriptor_length</code>	8	uimsbf	
<code>icon_locator_length</code>	8	uimsbf	
for (i=0; i<N; i++) {			
<code>icon_locator_byte</code>	8	uimsbf	
}			
<code>icon_flags</code>	16	bslbf	
for (i=0; i<N; i++) {			
<code>reserved_future_use</code>	8	bslbf	
}			
<code>}</code>			

`descriptor_tag` : 8 bit 整數值 0x0B 用以識別這描述。

`icon_locator_length` : 8 bit 整數說明在字串中的字元個數，那字串是將標準的圖像檔案名稱放在前面。

`icon_locator_byte` : 8 bit 值是一個位元組的圖像位置字串。

這圖像位置是字串的第一部份說明位置的圖像檔案。這是取決於應用程式類型。參考表 5-14。

`Icon_locator` 不應該使用斜線 " / " 作為結束字元。

表 5-14: 圖像位置的表示意義

application_type	description
0x0000	<code>reserved_future_use</code>
0x0001	For DVB-J this is a path relative to the base directory of the application as defined in 10.9.2, "DVB-J application location descriptor" on page 98.
0x0002	For DVB-HTML this is a path relative to the physical root of the application as defined in 10.10.2, "DVB-HTML application location descriptor" on page 100.
0x0003...0xFFFF	<code>reserved_future_use</code>

`icon_flags`: 這個16 bit攜帶一個數值，這是一個bitwise OR的flag位元定義了icons，提供這個flag是為了應用程式。這個flag位元定義在表 5-15。

表 5-15: 不同圖像旗標的定義(頁1/全2頁)

Icon flag bits	Description of icon size and pixel aspect ratio
0000 0000 0000 0001	32 x 32 for square pixel display
0000 0000 0000 0010	32 x 32 for broadcast pixels on 4:3 display (note 1)
0000 0000 0000 0100	24 x 32 for broadcast pixels on 16:9 display
0000 0000 0000 1000	64 x 64 for square pixel display
0000 0000 0001 0000	64 x 64 for broadcast pixels on 4:3 display

表 5-15: 不同圖像旗標的定義(頁2/全2頁)

Icon flag bits	Description of icon size and pixel aspect ratio
0000 0000 0010 0000	48 x 64 for broadcast pixels on 16:9 display
0000 0000 0100 0000	128 x 128 for square pixel display
0000 0000 1000 0000	128 x 128 for broadcast pixels on 4:3 display
0000 0001 0000 0000	96 x 128 for broadcast pixels on 16:9 display
xxxx xxxx 0000 0000	reserved_future_use
NOTE 1: approx. 15/16 pixel aspect ratio on 50 Hz system	

icon檔案的檔案名稱使用標準的方法來編碼：

filename = icon_locator " /dvb.icon." hex_string

hex_string = 4*4hex

hex = digit | "A" | "B" | "C" | "D" | "E" | "F" | "a" | "b" | "c" | "d" | "e" | "f"

digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"

一個圖示檔案應恰包含一個圖示。圖示檔案的圖示應指定在檔案名稱後成4個16位元的補數。

標記：這表示若application_icons_descriptor的icon_flags有一個值表示多重圖示的顯示，每一個指示圖示會有他自己的圖示檔案。例如，如果icon_flags 有一個值是0x0005，則目錄指定icon_locator 會包含兩個檔案名稱 dvb.icon.0004 (64x64 的方形像素成像) 和 dvb.icon.0001(32x32 的方形像素成像)

5.7.5 外部應用程式授權描述

(External application authorisation descriptor)

這“common”(最先)應用程式信號表的描述迴圈可能包含零或更多external_application_authorisation_descriptors。每個描述包含有關外部應用程式的訊息被允許繼續執行列在這個應用程式信號表之sub-table但不從這個service開始之應用程式。藉由此描述所包含的AIT次表，外部授權可提供應用程式定義application_identifier裡的application_identifier()。

表 5-16：外部應用程式授權描述語法

	No.of Bits	Identifier	Value
external_application_authorisation_descriptor() {			
descriptor_tag	8	uimsbf	
descriptor_length	8	uimsbf	
for(i=0; i<N; i++) {			
application_identifier()			
application_priority	8	uimsbf	
}			
}			

descriptor_tag: 這8-bit整數，當值為0x05 時為這個描述的識別。

application_identifier(): 這48-bit用來鑑定應用程式。這個filed的結構定義在5.5，"Application identification"。

application_priority:這8-bit整數說明優先順序，如此這個應用程式取得現行服務之背景。

參考5.7.3之application_priority，"Application descriptor"。

5.8 傳輸協議描述(Transport protocol descriptors)

5.8.1 傳輸協議描述(Transport protocol descriptor)

這傳輸協議描述定義了關於服務元件的傳輸協議，並且可能提供協議相關之資訊。

這描述可能被使用在"common" (.rst)描述迴圈或"application" (inner)描述迴圈。當在"common" 迴圈，它適用於在這sub-table之所有應用程式。任何這類的描述在"application"迴圈敘述額外的傳輸協議可用於特定的應用程式。

在這section所敘述每個應用程式應該在至少一個傳輸協議描述的範圍內。

表 5-17: 傳輸協定描述語法

	No.of Bits	Identifier	Value
<pre>transport_protocol_descriptor() { descriptor_tag descriptor_length protocol_id transport_protocol_label for(i=0; i<df; i++) { selector_byte } }</pre>	8 8 16 8 8	uimsbf uimsbf uimsbf uimsbf uimsbf	 N1

descriptor_tag: 這8 bit整數0x02用來辨識這個描述。

protocol_id: 一個協議的辨識用來運送應用程式。這個protocol_id的數值被登記在此與ETR162。

表 5-18: Protocol_id

protocol_id	description
0x0000	reserved_future_use
0x0001	MHP Object Carousel as defined in annex B, "(normative): Object carousel" on page 295.
0x0002	IP via DVB multiprotocol encapsulation as defined in ETSI EN 301 192 [5], ETSI TR 101 202 [49]
0x0003...0x00FF	reserved_future_use
0x0100...0xFFFF	Subject to registration in ETSI TR 101 162 [10]

transport_protocol_label: 如此8 bit field唯一地辨識在AIT section範圍內之傳輸協議。這應用程式描述參考這個數值來辨識運送應用程式之傳輸連結。

selector_byte: 額外的協議特定資訊。

表 5-19: 選擇位元組的表示意義

protocol_id	selector byte data
0x0000	reserved_future_use
0x0001	See 10.8.1.1 "Transport via OC".
0x0002	See 10.8.1.2 "Transport via IP".
0x0003...0xFFFF	Not defined in this version of the specification

5.8.1.1 透過物件連續撥放傳輸(Transport via OC)

當協定ID是0x0001，則傳輸協定描述之selector bytes應該如表 5-20所示。

表 5-20: 物件連續撥放傳輸的選擇位元組之語法

Syntax	Bits	Mnemonic
remote_connection	1	bslbf
reserved_future_use	7	bslbf
if(remote_connection == "1") {		
original_network_id	16	uimsbf
transport_stream_id	16	uimsbf
service_id	16	uimsbf
}		
component_tag	8	uimsbf

component_tag: 辨識傳送應用程式之"principal"服務成員 這個辨識成員是elementary stream運送object carousel之DSI。

remote_connection: 這個單位元旗標設定為 ' 1 ' 時指示傳輸連結是由transport stream提供不同於運送AIT之傳輸連結。當設定以下三個fields (original_network_id, transport_stream_id and service_id)是被包含在selector bytes。

這個辨識一個遠端的應用只在tuning之後開始。這樣的應用程式不應該有接收端autostarted但可能可以看見藉由application listing API對於可能由service selection開始(但不藉由

application launching API)。

單一位元旗標設定為 ' 0 ' 時，則傳輸連結由現在的服務來提供。

應用程式伴隨這個flag set應該有它們應用程式控制碼設定為REMOTE (參考表 5-2、5-3)。

參考6.7.2, "Application discovery and launching APIs"。

original_network_id: 這16 bit作為辨別transport stream之DVB-SI原始網路ID並提供傳輸連結。

transport_stream_id: 這個16 bit用來辨別transport stream之MPEG transport stream id並提供傳輸連結。

service_id: 這個16 bit用來辨別提供傳輸連結之DVB-SI service id。

5.8.1.2 透過 IP 傳輸(Transport via IP)

當協議ID為0x0002則在傳輸協議描述之selector bytes應該如表 5-21所示。

這個結構包含了兩個data_broadcast_descriptor之重要成員定義在EN 301 192 [5]。它提供全部必須資訊使得MHP獲得應用程式及應用程式資料成員由IP protocols傳送為了enhanced broadcast與Internet profiles。

表 5-21: IP傳輸的選擇位元組之語法(頁1/全2頁)

Syntax	Bits	Mnemonic
remote_connection	1	bsbf
reserved_future_use	7	bsbf
if(remote_connection == "1") {		
original_network_id	16	uimbsf
transport_stream_id	16	uimbsf
service_id	16	uimbsf
}		
alignment_indicator	1	bsbf
reserved_future_use	7	bsbf

表 5-21: IP傳輸的選擇位元組之語法(頁2/全2頁)

Syntax	Bits	Mnemonic
for(i=0; i<N; i++){		
URL_length	8	uimbsf
for(j=0; j<URL_length; j++){		
URL_byte	8	uimbsf
}		
}		

remote_connection: 這個與相關的三個(original_network_id, transport_stream_id與service_id)有著相同的語法與語意，這些fields在5.8.1.1有這相同的名稱，"Transport via OC"。

alignment_indicator: 這1-bit 指示alignment 存在 datagram_section 位元組與Transport Stream位元組之間(相同於有這個名稱定義在EN 301 192 [5] MPE data_broadcast_descriptor 的場域)。

URL_length: 這8-bit指示URL的位元組數目。

URL_byte: 這些位元組構成URL符合IETF RFC 2396 [43]。

對於URL使用"server"包含host:port標記法如IETF RFC 2396 [43]所定義，只有數字IP位址應該被使用來辨識IP傳輸被運送在廣播頻道中，當沒有Domain Name Service在broadcast-only方案被用來解決名稱。

IP到MAC的mapping應該被執行如IETF RFC 1112 [47]所描述。

註：

這份文件說明沒有特意的定義或要求任何URL的格式來被支援，因此他不能被使用在inter-operable way

5.8.2 IP 信號描述(IP signalling descriptor)

這訊號被定義為了在AIT之"common"描述迴圈使用。它們指示特定多重傳送IP位址之服務成員位址封包。路由封包應該提供一個完整的全部多重傳送IP位址列表，AIT section之應用程式可能需要。

當多重傳送IP被使用在一個服務時，路由描述是必要的。多重描述可能被使用。

表 5-22： IP信號描述語法

Syntax	Bits	Mnemonic
ip_signalling_descriptor () {		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
platform_id	24	uimsbf
}		

descriptor_tag: 這8-bit數值為0x11時辨識這個描述。

descriptor_length: 這8-bit用來辨識接下來位元組數目的長度。

platform_id: 這24 bit 包含一個組織的platform_id 提供IP/MAC streams on DVB transport streams/services.

platform_id: 的配置位址可以被發現在 ETSI TR 101 162 [10].

5.8.3 預先取得信號(Pre-fetch signaling)

5.8.3.1 簡介(Introduction)

若訊號定義為可開始取得檔案，將需要有先前較早的應用程式的部分存在。稍後應用程式存在的部分便會主動地使用API機制的需求事先取得檔案。此規範的版本，描述在這節與API-based pre-fetching沒有關聯。對單一應用程式，事先取得描述和AIT中DII的位置描述應指向相同的傳輸連線。如果transport_protocol_labels 在這些描述呈現不同時，參考傳輸協定描述應指向相同的傳輸連線（連續傳輸）。此訊號對於廣播是非必要的並且實做考量亦是必要的。

5.8.3.2 預先取得描述(Pre-fetch descriptor)

零或一個事先取得描述可被包含在AIT 之"application" (inner)描述迴圈。這被定義當傳送時 protocol_id 為 0x0001 時（MHP 物件連續傳送，MHP Object Carousel）使用。藉由 transport_protocol_label，每個描述有其對應相關的傳輸協定描述。

MHP終端可能使用這個描述來改善應用程式的start-up時間藉由有指示標籤之事先取得模組(參考 "Label descriptor")。

表 5-23: 預先取得描述語法

Syntax	Bits	Mnemonic
prefetch_descriptor () {		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
transport_protocol_label	8	uimsbf
for(i=0; i<N; i++) {		
label_length		
for(j=0; j<label_length; j++) {		
label_char	8	uimsbf
}		
prefetch_priority	8	uimsbf
}		

descriptor_tag: 這8-bit值為0x0C時定義此描述。

descriptor_length: 這8-bit用來定義接下來位元組數目的長度。

transport_protocol_label: 這8-bit定義傳輸協議描述，說明傳輸連結傳送這個事先取得描述涉及的模組。參考"transport_protocol_label"。

label_length: 這8-bit定義module label的位元數目。

label_char: 這些8-bit載送module label之一個陣列的位元組。這個標籤符合在DIIs之 moduleInfo結構之userInfo內，由標籤描述所傳送之一個或多個模組的標籤(參考"Label descriptor")。相同的module label可能附屬於數個模組。

prefetch_priority: 其值介於1與100之間(並包含兩者)。這表示具有使用特定優先權(100 highest, 1 lowest)的對應標籤的模組，其事先取得的提示。

5.8.3.3 DII 位置描述(DII location descriptor)

對於每個應用程式可以提供零或一個DII位置描述。它可以位於AIT之"common" (first)或"application" (inner)描述迴圈。它被定義當傳輸之protocol_id為0x0001(MHP Object Carousel)使用。藉由transport_protocol_label，每個描述有其對應相關的傳輸協定描述。

這模組是DSM-CC object carousel的一部份作為DownloadInfoIndication (DII)訊息信號。這object carousel在單一位置沒有列出全部存在的DII訊息。為了找到全部符合特定事先取得標記的模組(參考 "Pre-fetch descriptor")，必須得到全部DII的相關訊息。DII位置描述列在這些DII位置。假使DII位置描述部沒有包括，則當尋找符合特定標籤之模組時只有包含ServiceGateway且發出模組信號的DII應該被列入考慮。

在迴圈內的DII定義應該被歸為重要。DII包含label(s)還有最高事先取得優先權的應該被列為第一。實行module-based pre-fetching的接收者應該依序檢查DII對應的label，這些labels列在DII位置描述。

表 5-24: DII位置描述語法

Syntax	Bits	Mnemonic
DII_location_descriptor () { descriptor_tag descriptor_length transport_protocol_label for(i=0; i<N; i++) { reserved_future_use DII_identification association_tag } }	8 8 8 1 15 16	uimbsf uimbsf uimbsf bslbf uimbsf uimbsf

descriptor_tag: 這8-bit為0x0D時定義此描述。

descriptor_length: 這8-bit用來定義接下來位元組數目的長度。

transport_protocol_label: 這8-bit定義傳輸協議描述，說明傳輸連結傳送這個事先取得描述涉及的模組。參考"transport_protocol_label"。

DII_identification: 這15-bit定義DII訊息。這相當於transactionId的識別部份。

association_tag: 這16-bit定義連結(i.e. elementary stream)當DII訊息被播放。

5.9 DVB-J 特定描述(DVB-J Specific descriptors)

5.9.1 DVB-J 應用程式描述(DVB-J Specific descriptor)

這各描述的一個例子應該被包含在AIT之"application" (inner)描述迴圈為了每個DVB-J的應用

程式。這提供start-up參數資訊。

表 5-25: DVB-J應用程式描述語法

	No.of Bits	Identifier	Value
<pre>dvb_j_application_descriptor() { descriptor_tag descriptor_length for(i=0; i<N; i++) { parameter_length for(j=0; j<parameter_length; j++) { parameter_byte } } }</pre>	8 8 8 8	uimsbf uimsbf uimsbf uimsbf	

descriptor_tag: 這8-bit為0x03時定義這個描述。

parameter_length: 這8-bit整數說明parameter_byte字串的位元組數目。

parameter_byte: 這參數位元組包含一陣列之字串傳遞至應用程式當作參數。

5.9.2 DVB-J 應用程式位置描述 (DVB-J application location descriptor)

這個描述的一個例子應該被包含在AIT之"application" (inner)描述迴圈為了每個DVB-J的應用程式。它提供了不同的路徑資訊細項，允許DVB-J應用程式的獲得並且運作。

表 5-26: DVB-J應用程式位置描述語法

	No.of Bits	Identifier	Value
<pre>dvb_j_application_location_descriptor { descriptor_tag descriptor_length base_directory_length for(i=0; i<N; i++) { base_directory_byte } classpath_extension_length for(i=0; i<N; i++) { classpath_extension_byte } for(i=0; i<N; i++) { initial_class_byte } }</pre>	8 8 8 8 8 8 8	uimsbf uimsbf uimsbf uimsbf uimsbf uimsbf uimsbf	

descriptor_tag: 這8-bit為0x04時辨識這個描述。

base_directory_length: 這8-bit整數說明base_directory_byte字串的位元組數目。

base_directory_byte: 這些位元組包含一字串說明一個目錄名稱從carousel的根源開始，並且目錄由斜線字元 ' / ' (0x2F)界定。這個目錄被視為一個基本目錄為了相對路徑名稱。這個基本目錄自動地被考慮形成類別路徑的第一個目錄。假使基本目錄其根目錄字串為 ' / '。

classpath_extension_length: 這8 bit整數說明classpath_extension_byte字串的位元組數目。

classpath_extension_byte: 這些位元組包含一個字串說明DVB-J類別路徑的進一步延伸，如此應用程式的類別被搜尋除了基本目錄之外。類別路徑延伸字串包含路徑名稱，在此路徑的成員由分號字元 ' ; ' (0x3B)界定。路徑的成員可能包含絕對路徑字串來自carousel的根源或它們可以相對於基本目錄。這目錄由斜線字元 ' / ' (0x2F)界定並且絕對路徑名稱加上斜線字元 ' / '。

initial_class_byte: These bytes這些位元組包和一個字串說明在carousel的物件名稱，這是類別實現Xlet介面。

這個字串是一個DVB-J類別名稱可由類別路徑得到(e.g. "com.broadcaster.appA.MainClass")。

5.10 DVB-HTML 特定描述 (DVB-HTML Specific descriptors)

5.10.1 DVB-HTML 應用程式描述 (DVB-HTMP application descriptor)

這個描述的一個實例將分別被包含在AIT之"應用程式(application)"內部(inner)描述迴圈裡的DVB-HTML的應用程式。這個描述指示應用程式參數的值與廣播業者所用來發出表示應用程式狀態的控制信號。

表 5-27: DVB-HTML應用程式描述語法

	No.of Bits	Identifier	Value
<pre>dvb_html_application_descriptor(){ descriptor_tag descriptor_length appid_set_length for(i=0; i<N1; i++){ application_id } for(j=0; j<N; j++){ parameter_bytes } }</pre>	8 8 8 16 8	uimsbf uimsbf uimsbf b1b1f uimsbf	N1

descriptor_tag: 這個8位元整數，以0x08作為這個描述的識別值。

appid_set_length: 這個8位元整數明確指出application_ids串列的長度。

application_id: 這個16位元值域形成一個application_ids的集合(參照5.5, "Application identification")。這個值域資訊語意在這份規範中未加以定義。

parameter_bytes: 這參數位元組內含附加應用程式啟始路徑當作參數的字串。這參數位元組是用簡單連鎖方式組合在一起的。製作者有責任確保為它串接合法的前置字元(例如?或#), 以形成一個在語句構造上正確的URLs。

5.10.2DVB-HTML 應用程式位置描述

(DVB-HTML application location descriptor)

這個描述的一個實例將分別被包含在AIT之"應用程式(application)"內部(inner)描述迴圈裡的DVB-HTML的應用程式。

表 5-28: DVB-HTML應用程位描述語法

	No.of Bits	Identifier	Value
<pre>dvb_html_application_location_descriptor () { descriptor_tag descriptor_length physical_root_length for(i=0; i<N1; i++) { physical_root_bytes } for(i=0; i<N; i++) { initial_path_bytes } }</pre>	8 8 8 8 8	uimsbf uimsbf uimsbf uimsbf uimsbf	N1

descriptor_tag: 這個8位元整數, 以0x09作為這個描述的識別值。

physical_root_length: 這個8位元整數明確指出physical_root_byte字串的長度。

physical_root_bytes: 這些位元組包含字串詳細說明應用程式進入點之實體根源。這字串的語意是如表, "Transport specific semantic of physical root bytes"所表列的傳輸協定。

表 5-29: 實體根源位元組(root bytes)的傳輸特定語意

protocol_id	semantic
0x0000	reserved_future_use
0x0001	A directory specification
0x0002	One of the base URLs defined in the Transport protocol descriptor signalled for the application (see 10.8.1.2, "Transport via IP" on page 94).
0x0003...0xFFFF	TBD

initial_path_bytes: 這些位元組包含一個指定進入點文件之URL路徑元件的字串。

此路徑與定義在physical_root_bytes值域的根源有關。

5.10.2.1 範例(Example)

下面的例子描述DVB-HTML應用程式位置描述的使用方法。

一個應用程式製作者以下列的方法設計一個HTML應用程式：

- 此應用程式資料被散佈在數個目錄中，假設有一個"image"目錄與另一個"main"目錄。
- 此應用程式進入點是一個叫做"index.htm"的HTML文件，並且儲存在"main"目錄。

5.10.2.2 應成程式進入點(Application Engrly Point)

從這個應用程式製作者的觀點，路徑"main/index.htm"指向此應用程式進入點。這個路徑被儲存在應用程式位置描述之initial_path_bytes字串。

假使廣播業者將這個應用程式在插入於某個檔案系統中的"application"子目錄，應用程式位置描述之physical_root_bytes內容將為"application/"字串。

假使廣播業者透過IP傳輸應用程式，他們應該利用信號機制告知使用中的協定與IP位址，關於這個應用程式的傳輸協議描述，並且physical_root_bytes應該包含相符的URL字串。

5.10.3 DVB-HTML 應用程式界限描述

(DVB-HTML application boundary descriptor)

這個描述被定義的目的，是為了AIT應用程式迴圈的使用。它提供規則性表示法來敘述形成應用程式之資料組成單元。這個描述是非必需的。當不使用這個描述時，應用程式界限預設為來自傳輸信號的全部內容的完整集合，這個信號內容是與應用程式相關的傳輸協定描述。

多重界限描述可以用於相同的應用程式。在這種情況，個別的規則性表示法聯集(OR combination)為或稱合併(union)為等效的全域的規則性表示法。

表 5-30：DVB-HTML應用程式邊界描述語法

	No.of Bits	Identifier	Value
<pre>dvb_html_application_boundary_descriptor { descriptor_tag descriptor_length label_length for(i=0; i<N1; i++) { label_bytes } for(i=0; i<N2; i++) { regular_expression_bytes } }</pre>	8 8 8 8 8	uimsbf uimsbf uimsbf uimsbf uimsbf	N1

descriptor_tag: 這個8位元整數，以0x0A作為這個描述的識別值。

label_length: 這個8位元整數，明定label_bytes字串的長度。

label_bytes: 這些位元組包含一個由一套規則性表示法來辨識資料的標籤字串。這個標籤在特定的傳輸方式，可以被用作pre-fetching。

regular_expression_bytes: 這些位元組包含一個可用規則性表示法產生所有應用程式領域內的URLs的字串。

參考4.3.1.4.1, "Regular Expression Syntax"。

5.11 常數值(Constant values)

[表 5-31 : 註冊的常數值\(頁1/共2頁\)](#)

Where used	Type	Value	Where Defined	Scope
private data specifier descriptor	descriptor tag	0x5F	PSI & SI tables	SI
Data broadcast id descriptor		0x66	PMT	
Application Signalling Descriptor		0x6F	PMT	
Service identifier descriptor		0x71	SDT	
Label descriptor	descriptor tag	0x70	DII moduleInfo userInfo	SI-DAT
Caching priority descriptor		0x71		
Content type descriptor		0x72	BIOP objectInfo (note 1)	
reserved to MHP for future OC descriptors		0x73-0x7F	OC	
reserved to MHP for future use	table ID on AIT PID	0x00-0x73		MHP
Application Information Table		0x74		
reserved to MHP for future use		0x75-0x7F		
reserved for private use		0x80-0xFF		
Application descriptor	descriptor tag	0x00	AIT	MHP
Application name descriptor		0x01		
Transport protocol descriptor		0x02		
DVB-J application descriptor		0x03		
DVB-J application location descriptor		0x04		
External application authorisation descriptor		0x05		
reserved to MHP for future use		0x06, 0x07		
DVB-HTML application descriptor		0x08		
DVB-HTML application location descriptor		0x09		
DVB-HTML application boundary descriptor		0x0A		
Application icons descriptor		0x0B		
Pre-fetch descriptor		0x0C		
DII location descriptor		0x0D		
reserved to MHP for future use		0x0E-0x10		
IP signalling descriptor		0x11		
reserved to MHP for future use		0x12-0x5E		
private data specifier descriptor (note 2)		0x5F		
reserved to MHP for future use		0x60-0x7F		
User defined (note 3)		0x80-0xFE		
MHP Object Carousel	data broadcast id	0x00F0	PMT, AIT	SI
reserved for MHP Multi Protocol Encapsulation		0x00F1		
reserved to MHP for future use		0x00F0 - 0x00FE	PMT, AIT	SI

表 5-31 : 註冊的常數值(頁 2 / 共 2 頁)

Where used	Type	Value	Where Defined	Scope
MHP Application Service (see 10.11.1 on page 103)	service type	0x10	SDT	SI
NOTE 1: Strictly MessageSubHeader::ObjectInfo in the file message and the bound object info in a file binding of a directory or service gateway message. NOTE 2: The DVB SI private data specifier descriptor is defined for use in the Application Information Table to introduce private descriptors. NOTE 3: All user defined descriptors shall be within the scope of a private data specifier descriptor (see 10.4.7, "Use of private descriptors in the AIT" on page 83).				

5.11.1 MHP 應用程式服務(MHP Application Service)

這個服務型式必須使用內含至少一個自動啟始的 MHP應用程式的服務，當中的MHP應用程式即是這個服務的主要元件。若包含任意的廣播音訊或視訊的服務被付與這個型式的信號，瀏覽器不須要開始播放它們，只須留給自動啟始的應用程式。

5.12 服務資訊(Service Information)

5.12.1 服務識別碼描述(Service identifier descriptor)

一個服務的SDT描述可能不含或包含很多服務識別碼。各個描述為其服務定義單一文字式的識別碼。

單一服務的識別碼能被指派給不同實體網路上的服務，即使他們擁有不同的 original_network_id 和 service_id。一個已知的服務識別碼只將與一個服務有關，即使有眾多服務也被視為同一個服務。

註：眾多服務中，那些是被視為 " 相同的 "，那些是 " 不同的 "，是由服務提供者決定。舉例來說，在不同網路上的兩個服務，這些服務擁有相同的節目內容，但分屬於不同的地區的廣告，一般來說，可能會被當作是 " 相同的 " 服務。然而，是否用這個方法，原全取決於服務提供者。

超過一個以上的服務識別碼會被配置在一個服務實例。

表 5-32：服務識別碼描述

	No. of Bits	Identifier	Value
<code>service_identifier_descriptor () {</code>			
<code>descriptor_tag</code>	8	uimsbf	
<code>descriptor_length</code>	8	uimsbf	
<code>for (i = 0; i < descriptor_length; i++) {</code>			
<code>textual_service_identifier_bytes</code>	8	uimsbf	
<code>}</code>			
<code>}</code>			

descriptor_tag: 這個8位元整數，以0x71作為這個描述的識別值。

textual_service_identifier_bytes: 這些位元組內含唯一的服務識別碼，其編碼方式是使用 DVB SI對文字字串標準編碼法。

6 DVB-J 平台

6.1 虛擬機器

DVB-J 虛擬機器如 Java VM [34] 和 JVM Errata [68] 所定義，再加上 Inner Classes [69] 中所定義的 Inner Class specification。

Java 虛擬機器必須支援版本在 45.3 到 45.65535 之間的 Java 類別檔案。

6.2 一般問題

6.2.1 基本考量

除非本規格中另有指定，否則一些本規格中參照到的 API 中，若有一些類別和方法已經被標明為已淘汰 (deprecated) 者，MHP 平台實做便不需要支援。當本規格中定義的一個類別是實做某一個介面，而該介面卻要求必須提供一個已淘汰的方法，此時，介面的規定高於已淘汰標記的效力，因此，該方法在本規格中也將視為必須。

每一個 DVB-J 應用程式實例在邏輯上都將在自己的一份虛擬機器中執行。因此，應用程式不能假定在它結束時會執行應用程式類別中定義的終結程序 (finalizer)。當應用程式管理員終止應用程式所在的虛擬機器時，可以不執行應用程式的終結程序，如 Java VM [34] 第 2.17.9 節所定義。

DVB-J 應用程式不能與系統類別或其他系統靜態物件 (static object) 同步，否則會出現未知的行為。

只有當方法宣告中指定了 `SecurityException`，或是本規格因所參照到的封裝 (package) 沒有指定而指定使用 `SecurityException` 時，才能產生 `SecurityException` 例外狀況。

只要意義不變，實做時可以不要蓋過 (override) 原來的的方法。這意味著在使用 `java.lang.reflection` 封裝時，不同的實做會有不同的做法。

在納入 (include) 一個 Java 封裝時，不代表同時會納入其他的子封裝 (sub package)，譬如，在納入 `javax.tv.media` 封裝時，並不會自動納入 `javax.tv.media.protocol` 子封裝。本規格將會明列需要納入的子封裝。

在 `org.dvb` 名稱空間 (namespace) 中的 `public` 或 `protected` 方法與欄位，若沒有列於本規格中，則 DVB-J 應用程式不應該使用。如果應用程式可以支援本規格的各個版本，在執行時應

該只用那些適合於執行平台版本的 `org.dvb` 方法。

應用程式不應該在本規格所定義的封裝名稱空間內定義類別或介面。MHP 終端機應該透過 `SecurityManager.checkPackageDefinition` 機制來強制執行這項規定。

若 DVB-J 應用程式在設計時，就能夠同時支援多個 MHP 應用組態 (profile)、應用組態版本、或一些選擇性功能，就可以參照到只有 MHP 終端機支援這些應用組態、組態版本、或選擇性功能時才會有的 API 類別。MHP 終端機必須確保一件事，就是當 DVB-J 應用程式執行時，只有在應用程式參照的類別 bytecode 不存在時，才能產生類別無法連結的錯誤。當載入一個類別時，不能因為類別檔案中參照到不存在的類別，就產生錯誤。

註：只要應用程式利用條件分支 (conditional branch) 先確定某個 API 功能是否存在，才呼叫這個 API 方法，就算 MHP 終端機沒有某些 API 功能，該應用程式還是可以執行才對。

6.2.2 子集的處理

若本規格中某個類別所包含的方法、欄位、或建構函式宣告參照到實做應用組態所沒有的類別，實做就不需要提供這些方法。當然，實做也可以自行包含這些方法或類別。

當本規格只用到某個封裝的子集時，將整個封裝納入當然也是可以，但卻不需要。而多出來的功能，其行為表現對於廣播應用程式來說並沒有規範。

6.2.3 類別的載入

對於所有不屬於平台內的類別，DVB-J 應用程式環境的設計，應該讓每個應用程式都好像在自己的類別載入器 (classloader) 或類別載入器階層架構 (classloader hierarchy) 中執行。如此，兩個應用程式就不會存取同一份應用程式所定義的靜態變數 (static variable)。

在一個經過簽署的應用程式中，所有從 classpath 載入的類別或檔案，至少都必須使用應用程式當初用來簽署初始 xlet (initial xlet) 類別的憑證來簽署。此規定適用於組成該應用程式的類別檔案，以及透過 `java.lang.Class.getResource()` 機制所載入的影像或其他資料。MHP 終端機在認證一個經過簽署的應用程式時，可以選擇這些憑證中的一個來認證所有後續載入的類別或檔案。至於選擇的機制如何，則是隨實做而定。

註：如果 MHP 終端機對於好幾個用來簽署初始 xlet 的憑證都信任的話，則必須選擇最信任的憑證。

對於不適當、無意義、或是無效的定址器 (locator)，都會被自動略過。

6.2.4 卸載

類別的卸載如 Java Language Spec [32] 的第 12.8 節所定義，同時也支援 Java VM [34] 第 2.16.8 節之內容。

6.2.5 事件監聽器 (Event Listener)

如果一個事件監聽器沒有註冊，則 `org.dvb` 和 `org.davic` 中所有用來移除事件監聽器的方法都將沒有作用。

註：到底要用多少執行緒 (thread) 來負責將事件傳送給事件監聽器，是故意隨實做而定的。應用程式不應該阻擋事件監聽器，否則可能會影響其他事件的傳送。

6.2.6 DAVIC API 中的事件模型 (Event Model)

目前在 DAVIC API (DAVIC 1.4.1p9 [3]) 中用來延伸 `java.lang.Object` 的事件，現在應該改成延伸 `java.util.EventObject`。而 DAVIC 1.4.1p9 [3] 中定義的事件監聽器則應該延伸 `java.util.EventListener`。

6.2.7 DAVIC & DVB API 中的事件模型

每個在 `org.dvb` 和 `org.davic` 中從 `java.util.EventObject` 繼承的類別，都只是這些欄位的一個容器而已，同時其建構函式也不會對參數做任何有效性檢查 (validity check)。這些類別的實例基本上應該由平台實做來建構，而不是由應用程式來建構。平台實做只會利用適當的傳入資料來建構這些事件。

在 `org.dvb` 和 `org.davic` 中，除非另有明確指定，所有用來新增事件監聽器的方法，當遇到不同的呼叫所使用的參數都一樣時，則每一個監聽器只會加入一次。這意味著同樣的事件，每一個事件監聽器只會收到一次，就算將一個監聽器加入兩次也是沒用。

6.2.8 因副作用而造成選台動作

任何 MHP API 都不應該造成選台動作，除非明確註明。譬如，如果 `DVBClassLoader` 接收到一個需要選台的定址器，則應該當成所指到的檔案不存在一樣的處理。

6.2.9 應用程式內的媒體資源管理

當應用程式對所發出的媒體解碼資源請求不夠清楚時，則假設最近一次請求的媒體解碼資源為應用程式最想要的。此規則適用於 MPEG-2 I-frames、MPEG-2 視訊涓流以及串流視訊。同樣地，此規則也適用串流音訊與記憶體中的音訊。

在同一個應用程式中，當一個非廣播媒體畫面的呈現（記憶體中的音訊或視訊，亦或是靜態影像）因為失去資源而被打斷時，其畫面呈現將被取消並且不再回復。

在同一個應用程式中，當一個廣播媒體畫面因為失去資源而被中斷時，其畫面在中斷結束之後繼續回復。

6.2.10 應用程式執行緒優先順序

一個應用程式所有執行緒的 ThreadGroup，其 MaxPriority（最高優先等級）的值應該是 `java.lang.Thread.NORM_`

`PRIORITY`。

註：因此，應用程式將無法產生優先等級大於 `java.lang.Thread.NORM_PRIORITY` 的執行緒，因為應用程式沒有 `java.lang.RuntimePermission( "modifyThread" )` 權限。應用程式所產生的執行緒可以執行繁重的運算工作，也不會讓應用程式變得沒有回應。

6.2.11 文字編碼

MHP 平台預設的文字編碼方式為「UTF8」，如「3.1.5 文字的單一媒體格式」所定義，當 Java API 中提到平台預設文字編碼時，就是指這個。當然，「latin1」的編碼方式也將支援，如 ISO 8859-1 [20] 所定義。

在一個 Java String 當中，標記文字檔案所使用的字碼應該以 Java char 來編碼，其最高位元組為零，最低位元也有特定的值。此外，還將支援「DVBMarkupUTF8」編碼方式，此編碼方式除了下列幾點之外，基本上與 UTF8 是相同的：

當從 byte 轉到 char 時，標記序列必須依照前述的方式轉成 char。

當從 char 轉到 byte 時，一串符合上述編碼方式的位元組，必須轉成標記編碼序列。

6.2.11.1 服務資訊 (SI) 的文字編碼

當 DVB SI API 或 JavaTV API 在存取 SI 表格中的編碼字串，然後以 String 物件傳回時，必須支援下列編碼方式：

ISO 6937 (預設)

ISO 8859-5 到 ISO 8859-9 (SI 字串的第一個位元組碼為 0x01...0x05)

ISO 8859-1 到 ISO 8859-9 (SI 字串的第一個位元組碼為 0x10)

16 位元 ISO/IEC 10646-1 UCS-2 (SI 字串的第一個位元組碼為 0x11)

這些編碼方式在 `org.dvb.si.SIUtil` 類別的方法中也將支援。MHP 終端機不需要支援 DVB SI 規格中所訂的其他文字編碼方式。

6.3 DVB-J 基礎 API

6.3.1 Java 平台 API

下列封裝定義在 JAE 1.1.8 API [31]。請參閱「6.8.1.3 其他類別」。

6.3.1.1 java.lang 封裝

本規格支援 `java.lang` 封裝，但有下列變動。

- a) 當「交互運作式應用程式」(inter-operable application) 呼叫下列方法時，這些方法將永遠會產生 `SecurityException` 例外狀況：

`Runtime.exec`

`Runtime.load`

`Runtime.loadLibrary`

`Runtime.exit`

`System.exit`

`System.load`

`System.loadLibrary`

- b) MHP 平台實做不需要包含下列方法：

`Runtime.runFinalizersOnExit`

`System.runFinalizersOnExit`

`Thread.destroy`

`Thread.stop`

`Thread.suspend`

`Thread.resume`

`Thread.countStackFrames`

`ThreadGroup.stop`

`ThreadGroup.suspend`

`ThreadGroup.resume`

`ThreadGroup.allowThreadSuspension`

- c) 交互運作式應用程式不應使用下列欄位：

`System.in`

- d) 交互運作式應用程式不應使用下列類別：

`java.lang.Process`

- e) 應用程式應該可以使用：

`System.out`

`System.err`

`Runtime.traceInstructions()`

`Runtime.traceMethodCalls()`

做為偵錯 (debugging) 支用，而不會對應用程式造成不良影響。其輸出不應該顯示在一般終端使用者面前，同時也不應與其他 API 衝突。

- f) 對於 `java.lang.Compiler` 類別以及下列方法，只能當作應用程式對系統的提示，不過，沒有保證會發生什麼事情。

`Runtime.gc()`

`System.gc()`

- g) `System.getProperty()` 和 `System.getProperties()` 只要支援下列屬性 (property) 即可：

`file.separator`

`path.separator`

`line.separator`

`dvb.returnchannel.timeout` (請參閱「6.5.3 在返回通道上支援 IP」)

`dvb.persistent.root` (請參閱「6.5.6 永久性儲存 API」)

除了 `dvb.persistent.root` 之外，所有這些屬性對所有應用程式來說，都是能夠存取的。`dvb.persistent.root` 的存取，只能如「6.10.2.1 `java.util.PropertyPermission`」所定義。

以「`dvb.`」開頭的屬性都保留給未來使用。

- h) 若一個下載的 DVB-J 應用程式呼叫 `System.setProperties()` 和 `System.setSecurityManager()` 方法，將永遠會產生例外狀況。
- i) `SecurityManager` 類別的定義如 `PersonalJAE` [36]。
- j) 在 JDK 1.2.2 [78] 中提到，
`java.lang.SecurityManager.checkPackageDefinition` 和
`java.lang.SecurityManager.checkPackageAccess` 兩個方法會將封裝的名稱與從
`java.security.Security` 的一個方法所獲得的封裝清單做比對。由於
`java.security.Security` 類別在 MHP 當中並非必要，因此對於 MHP 來說，這兩個方法所用到的封裝限制清單，是從隨平台而定的清單取得。

MHP 終端機的開發人員，最好利用這個機制來限制應用程式定義的類別不能存取實做中的類別，以免破壞 MHP 終端機的安全機制。

MHP 終端機不應該使用空的 ("") 封裝名稱空間。

註：同時開發 MHP 終端機與 MHP 應用程式的開發人員，必須要確保應用程式所用到的名稱空間不會與 MHP 平台實做所用到的名稱空間衝突。

`System.currentTimeMillis()` 方法所傳回的時間值，其刻度不能大於 10 ms。
`Object.wait(long timeout)` 方法中的逾時數值 (timeout) 所指定的是等候的最長時間。
MHP 同時保證，如果沒有收到通知，物件最少會等 10 ms。

6.3.1.2 java.lang.reflect package

MHP 支援 `java.lang.reflect` 封裝。

6.3.1.3 java.util

MHP 支援 `java.util` 封裝。其 `Locale` 類別中的常數並非代表 MHP 所支援的地區設定 (locale)。MHP 所支援的地區設定是在應用組態 (profile) 中指定

而 `java.util.Properties.save()` 和 `java.util.Properties.load()` 方法中所用的格式必須是 JDK 1.2.2 [78] 所指定的格式。

一個 MHP 應用程式呼叫 `java.util.Timezone.setDefault` 方法所產生的結果，將只限於呼叫該方法的應用程式而已。

6.3.1.4 java.util.zip

MHP 支援 `java.util.zip` 封裝，不過下列類別例外：

- `Deflater`
- `DeflaterOutputStream`
- `GZIPOutputStream`
- `ZipOutputStream`
- `Adler32`

6.3.1.5 java.io

MHP 支援 `java.io` 封裝。雖然交互運作式應用程式不應呼叫 `java.io.ObjectInputStream.readLine()` 方法，但是這個方法還是必要，因為要跟先前版本的 MHP 規格相容。

此封裝中與檔案和檔案系統相關的類別與介面，在 MHP 規格中有額外的意義，列舉如下。

對於 broadcast carousel 的意義，請參閱「6.5.1.1 `java.io.File` 方法在 broadcast carousel 上的限制」。

對於永久性儲存的意義，請參閱 「6.5.6 永久性儲存API」。

6.3.1.6 java.net

在 java.net 封裝中，MHP 只支援 java.net.URL 和 java.net.InetAddress 兩個類別以及 MalformedURLException 和 UnknownHostException 兩個例外狀況。除非平台實做將此封裝納入特別的應用組態中，否則，本封裝其他內容都不在 MHP 的支援範圍，如「6.2.2 子集的處理」所定義。這些支援的類別中的宣告若參照到與該封裝其他部內容，則依「11.2.2 子集的處理」所定義的方式切開。

MHP 至少必須支援一種與實做相關的 java.net.URL 通訊協定。MHP 平台中，會傳回一個資源 URL 的方法，譬如 Class.getResource、ClassLoader.getSystemResource 和 ClassLoader.getResource，當本規格中沒有定義適當的通訊協定時，都應該傳回一個採用這種通訊協定的 java.net.URL 實例。

在一個簽署過的應用程式中，仍然要像未經簽署的應用程式一樣，傳回這個資源的 URL，不論其底層的檔案是否經過簽署。當存取該檔案時（譬如從透過 java.net.URL.openStream() 所取得的輸入串流存取該檔案），如果該檔案沒有通過認證（如「6.2.3 類別的載入」所述），則系統將當做該檔案沒有包含任何資料來處理（也就是將檔案長度當成零）。

註：由於簽章認證很花時間，因此，不是很必要認證的資產檔案（asset files）應該使用 java.io.File 或 org.dvb.dsmcc.DSMCCObject 來載入，並設為未簽署。

MHP 必須支援「file:」通訊協定。應用程式應該能夠使用「file:」URL 的字串來建構一個 java.net.URL 實例，如 IETF RFC 1738 [67] 所定義，其中 <host> 部分為空白字串，<path> 部分則是一個絕對路徑的檔案名稱。

而且 URL.getContent 方法應該依照其原定規格運作，就算是其參照的 URL 連線並不是在每一個 MHP 應用組態上都有效。

URL.getContent() 的回傳值類型定義在「表 6-1：URL.getContent() 回傳值類型」所列出的資料類型到 Java 類別名稱對應當中。

表 6-1：URL.getContent() 回傳值類型

資料類型	回傳值類型
未知或不支援的資料類型	java.io.InputStream
text/plain	java.io.InputStream
text/dvb.utf8	java.io.InputStream
text/Generic	java.io.InputStream

image/png	java.awt.image.ImageProducer
image/jpeg	java.awt.image.ImageProducer
image/mpeg	org.havi.ui.HBackgroundImage

URL.getContent() 的行為表現隨資料類型而定，依序參考下列資訊：

a) 內容類型的通知訊號，譬如：

OC 中的 content type descriptor

HTTP 標頭 (header) (若此應用組態支援的話)、Content-Type 標頭欄位 (如果有的話)

b) 副檔名 (如果知道的話) (請參閱「表 3-5：檔案類型識別碼」)

c) 開啟該檔案來了解

getContentType 會傳回 OC 中選擇性的 content type descriptor 所包含的值 (如果有的話)。

getFileNameMap 會傳回「表 3-5：檔案類型識別碼」中所對應出來的資訊。

在 java.net.URLStreamHandler 類別中，setURL 這個 protected 方法是不一定要有的。

在 java.net.URLConnection 類別中，getDefaultRequestProperty 和 setDefaultRequestProperties 這兩個 static 方法都不是一定要有的。

6.3.1.7 java.beans

在這個封裝中，MHP 支援下列類別和介面：

- Beans
- PropertyChangeEvent
- PropertyChangeListener
- PropertyChangeSupport
- PropertyVetoException
- VetoableChangeListener

- `VetoableChangeSupport`
- `Visibility`

在 Beans 類別中，只有下列方法是必須的：

- `Beans.instantiate(ClassLoader cl, String beanName)`
- `Beans.isDesignTime()`
- `Beans.isGuiAvailable()`

最後，下面的方法還有規定：

`Beans.isDesignTime()` 必須傳回「false」。

6.3.1.8 java.math

MHP 支援 java.math 封裝中的 `java.math.BigInteger`。

6.3.2 MHP 平台 API

6.3.2.1 org.dvb.lang

若建構時沒有指定父物件 (parent)，則執行呼叫的應用程式將由 `DVBClassLoader` 的代理父類別載入器 (delegation parent classloader) 做為其原始類別載入器 (original classloader)。

ClassLoader 的代理定義在 PersonalJAE [36] 中有關 `java.lang.ClassLoader` 的部分。

6.3.2.2 org.dvb.event

當 `org.dvb.event.EventManager` 傳送一個事件到監聽器 (listener) 時，平台實做應確保平台所產生的一個事件，每一個監聽器只能收到一次。如果監聽器還在處理前一個事件而尚未返回時，若有新的事件產生，MHP 終端機應該先等到它處理完前一個事件，才呼叫監聽器。

至於當平台內註冊了許多個事件監聽器時應如何表現，則隨實做而定。實做可以用多個執行緒來負責傳送同一個事件給不同的監聽器。實做應確保每一個應用程式同一時間只處理一個事件。

6.4 畫面呈現 API

6.4.1 圖形使用者介面 API

6.4.1.1 核心圖形使用者介面 API

底下封裝定義在 JAE 1.1.8 API [31]。

MHP 支援下列 `java.awt` 封裝中的類別與介面：

Adjustable	Graphics
• AWTError	GridBagConstraints
AWTEvent	GridBagLayout
AWTEventMulticaster	• GridLayout
AWTException	IllegalComponentStateException
BorderLayout	Image
CardLayout	Insets
Color	• ItemSelectable
Component	LayoutManager
Container	LayoutManager2
Cursor	MediaTracker
Dimension	Point
EventQueue	Polygon
FlowLayout	Rectangle
Font	• Shape
FontMetrics	Toolkit (as detailed below)

整個 `java.awt.event` 封裝也包含在內。

不過在 `java.awt.Image` 與其「`dvb.`」開頭的子類中，`getProperty` 方法所使用的屬性名稱，都保留給未來使用。

請參閱「6.8.1.3 其他類別」。

從 `Component` 到 `PopupMenu` (`add` 方法) 和 `MenuComponent` (`remove` 方法) 的宣告依存性 (signature dependency) 則依「6.2.2 子集的處理」所定義的方式切開。所有參照到 `java.awt.peer` 封裝的部分也比照辦理。在 `Component` 和 `Container` 中的 `list` 方法不是必須的。

在 Toolkit 類別中，下列方法是必須的：

`getDefaultToolkit`

`getFontList`

`getFontMetrics`

`sync`

`getColorModel`

所有與影像相關的方法

下列兩個方法從 PersonalJAE [36] 加到 Toolkit 類別中。這兩個方法不應快取 `java.awt.Image` 物件。

- `public abstract Image createImage(String filename);`
- `public abstract Image createImage(URL url);`

下列兩個方法的影像快取行為，現在已經明確定義在 PersonalJAE [36]。這兩個方法會快取 `java.awt.Image` 物件，呼叫時若傳入相同的參數，就會傳回同一個物件。

- `public abstract Image getImage(String filename);`
- `public abstract Image getImage(URL url);`

應用程式應可以使用 `Toolkit.bEEP`，而不會對應用程式有不良的副作用。其輸出不一定要讓一般終端使用者聽得到，同時也不應與其他 API 衝突。

MHP 應該支援 `getScreenResolution` 和 `getScreenSize` 兩個方法，同時還要包含 HAVI [50] 所描述的內容。

`java.awt.image` 封裝是必須的。至於 `java.awt.image` 所使用的影像內容的編碼，則如「3.1.1 點陣圖影像格式」所定義。MHP 平台所支援的格式，則隨應用組態而定。

當使用 `java.awt.FontMetrics` 類別時，`charsWidth` 或 `stringWidth` 方法所傳回的一串字元寬度必須將字間微調 (kerning) 和字型顯示時的次像素定位 (sub-pixel positioning) 都考慮進去。若計算時是採用個別字元寬度加總的方式，則可能會算出不一樣的值，因為，只是將字元寬度加總，並沒有考慮到字間微調和字型顯示時的次像素定位。

從廣播網路下載字型將透過 `org.dvb.ui.FontFactory` 的方法來支援。這些方法在無法下載字型時，必須通報。`java.awt.Font` 的建構函式只需要知道平台上既存的字型。

6.4.1.2 電視使用者介面

MHP 應支援 HAVi [50] 中所定義的 `org.havi.ui` 和 `org.havi.ui.event` 兩個封裝。而 `org.havi.ui.event.HRcEvent` 的實例則透過正常的 `java.awt` 事件機制來傳送，因為它是繼承 `java.awt.event.KeyEvent`。

除了 `HSound.load` 方法之外，HAVi [50] 中的任何方法都不能在 MHP 環境中產生安全性例外狀況。而 `HSound.load` 所允許的範圍，也僅限於針對 `java.io` 所定義的部分。

在對 `HVideoDevice` 使用 `getVideoController` 方法時，此方法的行為應該如下：

該方法應該只傳回處於 `Prefetched` 和 `Started` 狀態下的 JMF 播放器（請參閱 Java Media Player Specification [33]），同時該播放器應該是使用 `HVideoDevice` 為其稀少資源之一。否則，應該傳回 `null`。該方法不應傳回其他應用程式的 JMF 播放器（此播放器正使用 `HVideoDevice` 做為底層的視訊裝置）。

除了下面說明的狀況之外，該方法只能傳回應用程式已經透過 `javax.media.Manager.createPlayer` 呼叫產生的 JMF 播放器，或是從 `javax.tv.service.selection.ServiceContext.getServiceContentHandlers` 所傳回的 JMF 播放器（請參閱 Java TV [51]）。

上述規則唯一的例外，就是當應用程式環境的背景已經有視訊正在播放，但是卻尚未呼叫過 `javax.tv.service.selection.ServiceContext.getServiceContentHandlers`。此時，`getVideoController` 應該傳回後續使用 `getServiceContentHandlers` 呼叫所會傳回的 JMF 播放器。

`HComponent` 和 `HContainer` 兩個類別的宣告應該加上「implements `org.dvb.ui.TestOpacity`」。

`HGraphicsConfiguration.getPunchThroughToBackgroundColor` 方法不應該用在交互運作式應用程式。

`org.havi.ui.HfontCapabilities` 類別中的 `fontAvailable()` 和 `downloadFont()` 不應該用在交互運作式應用程式。

MHP 應支援 Java TV [51] 所定義的 `javax.tv.graphics`。

應用程式在呼叫 `javax.tv.graphics.TVContainer.getRootContainer` 所傳入的參數，應該與 MHP 終端機呼叫其 `initXlet` 方法時所傳入的 `xletContext` 是同一個。若傳入 `getRootContainer` 方法的 `xletContext` 不是同一個，其行為將隨實做而定，而且該方法可以傳回 `null`。除此之外，此方法不得傳回 `null`。

`HComponent.isDoubleBuffered` 不應該用在交互運作式應用程式。

6.4.1.3 繪圖延伸

6.4.1.4 處理輸入事件

MHP 包含兩種方式，讓應用程式收到輸入事件：`java.awt.Component` 所用的正常 AWT 方法，以及 `org.dvb.event` 封裝（請參閱「6.3.2.2 `org.dvb.event`」）。在正常 AWT 方法中，當元件獲得輸入焦點時，應用程式就會正常收到輸入事件。

通常，MHP 終端機記憶體內的瀏覽器軟體，會使用遙控器上的一些按鍵來做為瀏覽之用。瀏覽器不應該處理傳送給 MHP 應用程式的輸入事件。尤其，瀏覽器應該忽略那些傳送給 MHP 應用程式的數字鍵。反過來說，如果擁有輸入焦點的 MHP 應用程式已經表明不想收到數字鍵（利用 `HScene.setKeyEvents` 方法，然後傳入一個不包含數字鍵的 `HEventGroup`），此時瀏覽器就可以處理這些數字鍵輸入。至於 MHP 應用程式應該何時提出要求以表明想收到的輸入事件、何時不能提出要求，請看下面說明。

為確保使用者的感覺維持一致性，務必遵守下列原則：

應用程式若建立 `HScene` 物件並置入一些元件，預設情況不應該取得這些元件的輸入焦點

應用程式可以呼叫 `Component.requestFocus()` 來取得輸入焦點。如果成功，則輸入焦點將移到提出請求的元件。

即使應用程式未擁有輸入焦點，也可以利用 `org.dvb.event` API 來要求獲得部分的輸入事件。

對於一般電視節目所提供的應用程式，請將下列各點考慮進去：

當螢幕上正在播放電視節目視訊，而且使用者的認知是處於觀看電視節目的狀態，而不是正在和電視互動（底下稱「電視觀賞模式」），此時應用程式不應該請求獲得 AWT 輸入焦點，而是應該讓瀏覽器處理大部分的輸入事件（譬如，數字鍵、方向鍵、Enter 鍵等等應該如使用者的預期，由瀏覽器處理）

在「電視觀賞模式」中，應用程式應該只要求最低限度的輸入事件，譬如，只有會產生進一步行動的事件，或是會讓使用者從「電視觀賞模式」切換到使用者與應用程式互動狀態的事件。

如果應用程式要求了一些事件，以便讓使用者可以在「電視觀賞模式」中喚起應用程式，建議您使用 `VK_COLOURED_KEY(0 3)` 和 `VK_TELETEXT`（請看下面備註）輸入事件來做此用途，而且應用程式應該透過 `org.dvb.event` API 來提出請求。在實際應用上，這意味著在「電視觀賞模式」中，瀏覽器不應該將重要的功能對應到 `VK_COLOURED_KEY(0 3)` 事件。

MHP 應用程式若希望離開「電視觀賞模式」並接收 `VK_COLOURED_KEY(0 3)` 和 `VK_TELETEXT`

以外的輸入事件，則必須在螢幕上顯示一個不透明的繪圖物件，此物件必須至少涵蓋可見區域的 3 % (在 720x576 繪圖模式下，一個 634x518 的可見區域中，3% 就是 9852 像素)，該物件要清楚標明目前電視接收器在處理輸入事件方面，是處於「非瀏覽器」模式。

註 1：電視節目中如果有傳送 DVB Teletext 字幕，在「電視觀賞模式」下的瀏覽器在遇到這個事件時，通常會是啟動 teletext 解碼器（如果該終端機支援的話）。如果應用程式要求收到這個輸入事件，就會讓使用者無法方便地啟動 DVB Teletext 解碼器。因此，當節目包含 DVB Teletext 字幕時，本規格建議只有提供 simulcast 版本 teletext 字幕的應用程式才在「電視觀賞模式」下要求獲得 VK_TELETEXT 輸入事件，這樣就能提供替代 DVB Teletext 字幕的資訊。

在 WST Teletext 字幕很普遍的環境中，視訊服務中就不會有 DVB Teletext 字幕，此時 MHP 應用程式應該提供類似 teletext 的資訊服務。當再次收到這個事件時，就終止該資訊服務，讓使用者回到正常觀賞模式。

在 MHP 應用程式中，VK_TELETEXT 事件不應該做其他用途。

註 2：上面幾點只適用於「電視觀賞模式」。當使用者正處於與應用程式互動的狀態時，應用程式當然可以使用正常的 AWT 輸入事件機制，並請求獲得輸入焦點，同時並使用 org.dvb.event API 來獲得任何平台支援的輸入事件。

註 3：對於資料服務來說，一個在服務選擇之後自動啟動的應用程式，其起始狀態也是「電視觀賞模式」。這是為了讓使用者可以使用瀏覽器來順利瀏覽各項服務，而不會被搞混。

6.4.1.5 字型關係

6.4.1.5.1 PFR0

對於使用 PFR0 格式的字型（3.4 可下載字型）來說，字型尺寸相關的 Java API 與字型格式之間的關係如下：

FontMetrics.getMaxAscent 和 FontMetrics.getMaxDescent 兩個方法所傳回的數值，必須分別為 yOffsetTop 和 yOffsetBottom。

對於包含輔助資料 (auxiliary data) 的 PFR 字型來說：

FontMetrics.getAscent 應該傳回 ascent 欄位。

FontMetrics.getDescent 應該傳回 descent 欄位。

FontMetrics.getLeading 應該傳回 externalLeading 欄位。

所有在輔助資料記錄中的數值，都應該先從字型尺寸單位轉換成像素再傳回。此轉換過程若

遇到小數點，應該無條件進位。

對於沒有輔助資料的 PFR 字型：

`FontMetrics.getAscent` 應與 `getMaxAscent` 相同。

`FontMetrics.getDescent` 應與 `getMaxDescent` 相同。

`FontMetrics.getLeading` 應該為零。

一個字元的「advance width」(以字型尺寸為單位) 定義如下：

對於有設定「proportionalEscapement」旗標的 PFR 字型，字元的 advance width 就是字元記錄的「charSetWidth」欄位。

對於沒有設定「proportionalEscapement」旗標的 PFR 字型，字元的 advance width 就是實體字型記錄中的「standardSetWidth」欄位。

`java.awt.FontMetrics.charWidth()` 應該傳回轉換成像素的 advance width。此轉換過程若遇到小數點，應該無條件進位。

`java.awt.FontMetrics.stringWidth()`、`java.awt.FontMetrics.bytesWidth()` 和 `java.awt.FontMetrics.charsWidth()` 都是將先字串中所有字元的 advance width 加總，然後再根據字間微調來調整。字間微調必須在字型尺寸單位上進行，然後，才將結果轉成像素單位。此轉換過程若遇到小數點，應該無條件進位。

`java.awt.FontMetrics.getMaxAdvance()` 傳回

`java.awt.FontMetrics.charWidth()` 可傳回的最大值。請注意，對於調和字型 (proportional font)，平台實做必須從字型中的每一個字的 advance width 計算出來，PFR 檔案中沒有任何欄位包含這個數值。

6.4.2 串流媒體 API

6.4.2.1 解決方案架構

MHP 平台應該實做 `javax.media` 和 `javax.media.protocol` 兩個 Java Media Framework 封裝，如 Java Media Player Specification [33] 所定義，並參考下面釐清、延伸與限制各節的說明。

6.4.2.2 釐清

JMF 在播放透過 MPEG 傳輸串流所傳遞的 MPEG 內容時所使用「時間基準時間」(time base time)

是一個不斷在跑的時間，其頻率可以和 MPEG 中的 Program Clock References / System Time Clock 同步，或是透過其他合適的方式同步。其數值與 MHP 接收器中的 MPEG System Time Clock 沒有直接的關聯。

JMF 在播放 MPEG 傳輸串流所傳送的 MPEG 內容的媒體時間，只是媒體串流播放時不斷在跑的時間而已，其實際的數值則隨實做而定。

註：JMF 媒體時間，與任何 MPEG-2 / DSM-CC Normal Play Time 所提供的時間基準 (time base) 之間，沒有隱含或預期的關聯。

當一個 JMF 播放器在建構時，那些參照到 DVB 服務、事件或基礎串流 (elementary stream) 的定址器 (locator) 和 URL，都會產生一個能夠直接從廣播網路播放相關內容的播放器。而那些參照到檔案的定址器和 URL 所產生的播放器，則會在其 Prefetching 狀態時，就先從廣播網路下載相關的內容。如果內容無法下載，則該播放器永遠都不會進入 Prefetched 狀態。

交互運作式應用程式不應該呼叫 `javax.media.MediaHandler.setSource()`。

在 `javax.media.PackageManager` 中：

- a) 使用 `set<xxx>PrefixList` 方法的效果僅限於呼叫該方法的應用程式。
- b) 在 MHP 內，`commit<xxx>` 方法應該永遠產生 `SecurityException`。

`javax.media.ControllerEvent` 和 `javax.media.GainChangeEvent` 兩個事件，應該繼承自 `java.util.EventObject`。

`javax.media.ControllerListener` 和 `javax.media.GainChangeListener` 兩個介面應該繼承自 `java.util.EventListener`。

`SubtitlingLanguageControl.isSubtitlingOn()` 應傳回目前字幕顯示的狀態，而不是 `setSubtitling` 最近一次設定的數值。尤其當 `setSubtitling` 會被設為 `true`，而廣播網路中卻沒有字幕，或是未選擇字幕服務元件時，`isSubtitlingOn` 就應該傳回 `false`。

任何在 `DVBLocators` 之後的元素 (包含 `event_id` 元素) 都會被 JMF 播放器略過。

6.4.2.3 預設媒體播放器行為表現

對於正在呈現 DVB 服務的一個 JMF 播放器來說，當服務當中包含多個音訊、視訊、或字幕元件時，要遵循下列規則 (按下列順序) 來決定應該呈現哪一個服務元件：

- a) 對於不同語言的音訊和字幕串流，就利用使用者偏好來決定選擇哪一個串流。

b) 在廣播訊號通知資訊（也就是 PMT）中第一個串流，就是要呈現的串流。

如果服務中的媒體元件變更時，那麼平台實做應該儘量考慮使用者偏好，自行選擇最適當的替代元件。

6.4.2.4 視訊涓流所需的控制元件

下列控制元件支援視訊涓流（請參閱「3.1.3 MPEG-2 視訊涓流（Video Drip）」一節）：

- `javax.tv.media.AWTVideoSizeControl`
- `org.dvb.media.BackgroundVideoPresentationControl`

6.4.2.5 架構延伸功能

6.4.2.5.1 DVB 規定的延伸功能

如果一個連結到 `DripFeedDataSource` 的 `Player` 收到一個 `ResourceWithdrawnEvent` 然後又收到一個 `ResourceReturnedEvent`，該播放器所連結到的視訊解碼器的視訊輸出內容，將隨實做而定。而使用該 `Player` 的 MHP 應用程式應該在收到 `ResourceReturnedEvent` 之後，儘快呼叫 `DripFeedDataSource.feed` 方法，並傳入一個 I-frame。

註：應用程式開發人員在建構視訊涓流序列時，應該將這點考慮進去。如果在 I-frame 之間出現大量的 P-frame，則可能會造成顯示內容嚴重的延遲，這是因為新資料送入解碼器的速率上有所限制。

6.4.2.5.2 org.davic 中的延伸功能

本規格將包含下列 `org.davic.media` 封裝中的類別和介面（如 DAVIC 1.4.1p9 [3] 的附錄 L 所定義）。

<code>MediaPresentedEvent</code>	<code>LanguageNotAvailableException</code>
<code>MediaLocator</code>	<code>NotAuthorizedException</code>
<code>MediaTimePositionControl</code>	<code>NotAuthorizedMediaException</code>
<code>FreezeControl</code>	<code>MediaFreezeException</code>
<code>ResourceWithdrawnEvent</code>	<code>LanguageControl</code>
<code>ResourceReturnedEvent</code>	<code>AudioLanguageControl</code>
<code>MediaTimePositionChangedEvent</code>	<code>SubtitlingLanguageControl</code>

而下列 `org.davic.media` 封裝中的類別 (如 DAVIC 1.4.1p9 [3] 的附錄 L 所定義) 也將包含在本規格中, 但將包含下列修正:

對 `MediaTimePositionControl` 呼叫 `setMediaTimePosition()` 所開始的每一個動作在完成都會收到一個 `org.davic.media.MediaTimePositionChangedEvent`。

只有在使用 `org.davic` package 中的控制元件而導致變化時, 還有在 JMF 播放器轉換到 Started 狀態的過程時, 才產生 `org.davic.media.MediaPresentedEvent`。第二種情況只是增加 JMF 狀態轉換的事件而已, 並沒有取代原有事件。

註: 使用 `javax.tv.media.MediaSelectControl` 而導致的變化, 不會產生 `org.davic.media.MediaPresentedEvent`。

註: 本規格針對 `ResourceWithdrawnEvent` 和 `ResourceReturnedEvent` 訂定了比原始 DAVIC 規格更細的條件。請參閱「6.6.2 服務選擇 API」。

6.4.2.5.3 javax.tv 中的延伸功能

MHP 必須實做 Java TV [51] 中所定義的 `javax.tv.media` 封裝。

在 `javax.tv.media.MediaSelectControl` 中 `InsufficientResourcesException` 是由 `select()` 方法因為系統資源不足而無法選擇任何服務元件時所產生。

在 `javax.tv.media.AWTVideoSize.equals()` 方法中, 其資料成員 (data member) 應該是用來建構該物件時, 所傳入的「source」和「dest」兩個參數。

從 `javax.tv.media.MediaSelectControl.getCurrentSelection` 傳回來的, 就是目前所選取的元件, 這有可能跟該控制元件先前所選取的不同。至於可能會改變目前選取元件的機制包括:

控制字幕和音訊語言

CA 系統, 包括 DVB Common Interface, 例如 Host Control 的 `replace / clear_replace`

使用者利用瀏覽器改變

底下釐清有關 `javax.tv.media.AWTVideoSizeControl` 中 `getDefaultSize` 方法的一些細節:

a) 此方法應該傳回目前預設的 `AWTVideoSize`。當 JMF 播放器的視訊縮放和定位是由 MHP 平

台所掌控時，預設值應該是 MHP 終端機實做決定（請參閱
`org.dvb.media.VideoFormatControl.DFC_PLATFORM`）。當視訊縮放與定位處於
`DFC_PLATFORM` 模式時，此方法的回傳值應該變成 `DFC_PLATFORM` 下的政策所設定的值。

- b) 呼叫 `AWTVideoSizeControl.setSize()` 時若傳入此方法所傳回的 `AWTVideoSize` 物件做為參數，會將視訊縮放設為 `getDefaultSize()` 呼叫當時的預設值。因此，這應該和呼叫 `BackgroundVideoPresentationControl.setVideoTransformation` 然後傳入
`VideoFormatControl.getVideoTransformation(getDecoderFormatConversion())` 所傳回的視訊轉換設定有相同的效果，而和傳入
`getVideoTransformation(DFC_PLATFORM)` 所傳回來的視訊轉換設定不同。

註：MHP 終端機處於 pan & scan 模式（請參閱
`org.dvb.media.VideoFormatControl.DFC_PROCESSING_PAN_SCAN`），則幾乎在
`AWTVideoSizeControl.getSize()` 呼叫返回的瞬間，其傳回的值就已經過時。

6.4.2.5.4 廣播組態所需的控制元件

底下控制元件用來支援「3.2 廣播串流格式」一節所訂定的廣播串流格式：

`org.davic.media.LanguageControl`

`org.davic.media.AudioLanguageControl`

`org.davic.media.SubtitlingLanguageControl`

`org.davic.media.FreezeControl`

`javax.tv.media.MediaSelectControl`

`javax.tv.media.AWTVideoSizeControl`

`org.dvb.media.VideoPresentationControl`

`org.dvb.media.BackgroundVideoPresentationControl`

`org.dvb.media.SubtitlingEventControl`

`org.dvb.media.VideoFormatControl`

`org.dvb.media.DVBMediaSelectControl`

底下控制元件用來支援從「3.1.4 音訊片段的單一媒體格式」所解碼出來的媒體：

`org.davic.media.MediaTimePositionControl`

6.4.2.5.5 釐清

在 `SubtitlingLanguageControl` 中，當前顯示的字幕應該是當前執行的字幕解碼器所輸出字幕。此時，螢幕上並不一定要顯示字幕。請參閱「7.5 字幕」一節。

當一個 `PresentationChangedEvent` 產生時，其 `Player` 應該重新整理從 `getControls()` 方法傳回的控制元件清單。同樣地，支援 `MediaSelectControl` 的 `Player` 應該在 `MediaSelectControl.select()` 方法被呼叫之後，重新整理從 `getControls()` 方法傳回的控制元件清單。

應用程式在 `PresentationChangedEvent` 或 `MediaSelectSucceededEvent` 產生之後所參照到的任何控制元件，都可以當成已經無效。至於新内容的控制元件仍然持續有效。應用程式在這些事件產生之後，應該重新獲得所需的最新控制元件。

如果目前 `JMF` 播放器正在呈現的内容因為選台動作的關係而無法讀取，則應該傳送 `org.davic.media.ResourceWithdrawnEvent` 給已註冊要接收該事件的監聽器。如果内容再度回復，則應傳送一個 `ResourceReturnedEvent` 給所有此時已註冊的監聽器。

6.4.2.6 架構在廣播上的限制

那些不支援的控制元件，在 `MHP` 中就不需要有對應的 `GUI` 元件。而呼叫 `Control.getControlComponent()` 就可能傳回 `null`。

`MHP` 應用程式開發人員不應該假設會有下列類別或介面存在：

`javax.media.CachingControl` (除非由 `JMF` 方法傳回)

`javax.media.CachingControlEvent`

`javax.media.GainControl` (除非由 `JMF` 方法傳回)。

`MHP` 平台在實做 `Player.getControls()` 方法時，不需傳回一個 `CachingControl` 或 `GainControl`，當然這樣做也並非不行。如果 `MHP` 平台實做真的傳回一個 `GainControl`，那麼，使用此控制元件所設定的音量，不得大於系統音量大小。也就是說，此控制元件只能在靜音與應用程式啟動時的音量之間做調整。系統音量大小將以 1.0 來表示。

`JMF Player` 的 `getVisualComponent` 方法不需要傳回一個 `java.awt.Component`，而且可以傳回 `null`。傳回 `null` 的 `Player` 只能用來播放背景視訊。而傳回一個 `java.awt.Component` 的 `Player`，則必須完全支援 `java.awt.Component` 有關定位與縮放的行為。

播放器啟動時，都是背景播放器。如果 `getVisualComponent` 會傳回一個元件，那麼該播放器也可以當成「元件式播放器」(component based player)。在 `getVisualComponent` 所傳回元件的 `paint` 方法第一次被呼叫之前，視訊仍將繼續在背景播放。在轉換過程中，視訊畫面會配合此 `visualComponent` 的大小與位置來調整。在此元件之外的區域，如須重新更新，則遵照「7.3 繪圖」一節的規定。

在轉換之後，應用程式應重新取得該播放器的 JMF 控制元件清單。至於是否要繼續使用先前的 JMF 控制元件，則隨實做而定。應用程式不應使用不在新的 JMF 控制元件清單中的舊控制元件。當一個 JMF 播放器是以元件式播放器執行時，則不應該支援下列的控制元件，同時必須調整 `getVisualComponent` 方法所傳回的控制元件大小與位置。

- `org.dvb.media.BackgroundVideoPresentationControl`
- `javax.tv.media.AWTVideoSizeControl`

應用程式不應假定 MPEG 廣播內容會有 `PushDataSource` 和 `PullDataSource` 存在，而且在 DVB 中的 JMF 實做必須提供這些功能。實際使用的 `DataSource` 是隨實做而定的，而且只規定必須繼承 `javax.media.protocol.DataSource` 類別。

`URLDataSource` 的建構函式對於廣播 MPEG 內容可以產生 `IOException`。

為了與 `javax.tv.media.MediaSelectControl` 整合，只要新、舊服務元件都參照同一個 `PCR_PID`，就沒有所謂的「重新同步」(resynchronization)。

JMF 實做應該不用選台就能允許選擇傳輸串流中不必選台 (tuning) 就能選擇的媒體元件 (即使這些元件不在目前選取的服務中)。

底下說明當使用了一個不選台就無法指到媒體元件的定址器所會發生的失敗狀況：

如果 JMF 播放器是一個 `ServiceMediaHandler`，其失敗狀況在 `MediaSelectControl` 敘述中有完整說明。

如果一個不是 `ServiceMediaHandler` 的 JMF 播放器在建構時傳入了這樣的一個定址器，則該播放器將不能進入 `Realized` 狀態。

同時在離開 `Realizing` 狀態時應該產生一個 `ResourceUnavailableEvent`。

對於不是 `ServiceMediaHandler` 的 JMF 播放器來說 `MediaSelectControl` 必須限用於當前可用的傳輸串流中不必選台就能使用的服務元件。

註：此版本的規格中，沒有一種媒體類型可以讓 `javax.media.Player.addController` 不產生 `IncompatibleTimeBaseException` 的。

6.4.2.7 MediaSelectControl 和

SubtitlingLanguageControl / AudioLanguageControl 功能重疊部分

`org.davic.media.LanguageControl.listAvailableLanguages()` 方法應該列出服務中所有可用的語言。這包括目前尚未選取的服務元件中所包含的語言，至於為何這些元件沒有被選取，可能是因為已經被 `MediaSelectControl` 取消選取，或是一開始就沒有選取（因為沒有列在定址器的元件標記 `component tag` 清單中）。

`org.davic.media.LanguageControl.selectLanguage()` 方法的效力應該蓋過任何透過 `javax.tv.media.MediaSelectControl` 或透過服務定址器（內含元件標記）所選擇的服務元件。如果 `selectLanguage()` 改變了所選取的服務元件，`javax.tv.media.MediaSelectControl` 上的方法應該當作這些改變是如同透過 `MediaSelectControl` 改變一樣，並傳回正確的服務元件（必須能夠正確反映出 `selectLanguage()` 所做的變更）。

如果使用 `javax.tv.media.MediaSelectControl` 上的方法來取代或移除目前提供音訊或字幕的服務元件，那麼該服務元件應該停止呈現其服務。如果使用 `javax.media.MediaSelectControl` 上的方法來新增一個提供音訊或字幕的服務元件（且目前還尚未有此類服務元件正在執行），那麼該服務元件應該開始呈現其服務。要知道新選擇服務元件的語言，應用程式應該呼叫 `AudioLanguageControl` 或 `SubtitlingLanguageControl` 的 `getCurrentLanguage` 方法來取得。如果嘗試使用 `MediaSelectControl` 上的方法來同時選擇好幾個媒體類型相同的服務元件來一起呈現，除非 MHP 終端機有足夠的資源來同時呈現，否則這些呼叫將會失敗而無任何訊息。其失敗狀況與 `MediaSelectControl` 在資源不足時所定義的狀況一樣。

使用 `MediaSelectControl` 或是 `SubtitlingLanguageControl` 從 `LanguageControl` 繼承來的 `selectLanguage` 和 `selectDefaultLanguage` 方法來控制字幕的顯示，不應該影響「Application Control」的數值，如「圖 7-19：如何決定字幕語言以及顯示設定」所示，反之亦然。

6.4.2.8 串流媒體 API 和電視使用者介面 API 功能重疊部分

6.4.2.8.1 基本原則

有些 `org.dvb.media.VideoFormatControl` 預先定義的解碼器格式控制，在視訊、繪圖物件以及背景合併之後，可能還必須控制最終輸出裝置的像素寬高比，如圖 7-2 所示。為了達到這點，一個 JMF 播放器應該嘗試保留其輸出顯示裝置的 `HVideoDevice` 實例。如果沒有保留此 `HVideoDevice`，對 JMF 播放器來說也不是什麼嚴重的錯誤，但是可能在電視畫面呈現上會比較差。

註：HVideoDevice 可用與真正使用的組態，需視 MHP 終端機的確切組態而定。機上盒、整合式數位電視以及電腦，會有一些差別。至於電視畫面呈現怎麼個差法，要視 MHP 終端機的組態、輸入視訊、以及最終輸出裝置的特性而定。

如果從一個正處於 prefetched 或 started 狀態的 JMF 播放器取得一個 HVideoComponent，其關聯的 HVideoDevice 應該就是 JMF 播放器視訊顯示的裝置。如果該 JMF 播放器正處於其他狀態，那可能就找不到關聯的 HVideoDevice。

6.4.2.8.2 電視行為控制

在「圖 7-16：JMF 播放器存在時的格式控制」中的電視行為控制 (TV Behaviour Control)，就是 JMF 播放器預設的視訊轉換 (video transformation) 行為 (如果沒有設定的話)。將 VideoTransformation 設定為 DFC_PLATFORM 就會將視訊格式控制設定在電視行為控制模式。

註 1：在此模式中，若視訊輸入訊號有任何會引起「電視行為控制」改變最終輸出訊號 (即合併了視訊、繪圖、字幕、背景之後的訊號) 像素寬高比的變化，也會改變繪圖、字幕、背景的像素寬高比。

註 2：如需「電視行為控制」在機上盒或整合式數位電視上的範例，請參閱 E-Book [1]。

6.4.2.8.3 應用程式行為控制

若 MHP 應用程式在控制視訊呈現方式時，是透過 BackgroundVideoPresentationControl (針對背景視訊) 或 VideoPresentationControl 和 java.awt.Component 上的方法 (針對控制元件中的視訊)，則該應用程式就必須負責監控視訊格式然後適當改變視訊呈現方式。

使用 VideoTransformation 類別的建構函式所建構的實例，不應該會影響 HVideoDevice 的組態設定。而透過 VideoFormatControl.getVideoTransformation 傳回的 VideoTransformation 實例對 HVideoDevice 的影響，應該與「電視行為控制」模式中所套用的轉換相同。如果 JMF 播放器並沒有保留其 HVideoDevice，則可能不會達到完全的效果。

6.4.2.8.4 動態行為

一個 JMF 播放器應該在進入 Prefetched 狀態之前 (除非其先前狀態是 Started 狀態) 嘗試保留其 HVideoDevice 物件。一個正處於 Prefetched 或 Started 狀態的 JMF 播放器，若先前沒有保留其 HVideoDevice，則應該只有在建立此 JMF 播放器的應用程式呼叫 org.dvb.media.BackgroundVideoPresentationControl 的 setVideoTransformation 方法時，才嘗試保留 HVideoDevice 物件。如果 JMF 播放器保留了 HVideoDevice 物件，則在離開 Started 或 Prefetched 狀態而進入 Realized 狀

態之前，應先將它釋放。

一個保留了 HVideoDevice 的 JMF 播放器若後來失去對該裝置的控制權，則應該發出一個 `org.davic.media.ResourceWithdrawnEvent` 給所有已註冊希望收到該 Player 的 `ControllerEvent` 事件的監聽器。如果播放器後來在嘗試保留該裝置時又成功獲得該裝置的控制權，則應該發出一個 `org.davic.media.ResourceReturnedEvent`。

6.4.2.8.5 資源管理細節

當一個 JMF 播放器保留了一個 HVideoDevice，那麼一個交互運作式應用程式就不應該呼叫 `HVideoDevice.getClient` 所傳回的 `ResourceClient` 實例上的方法。

交互運作式應用程式可以直接呼叫 `HVideoDevice.release` 或 `HVideoDevice.setConfiguration`。JMF 實做應該允許這樣的行為。

6.5 資料存取 API

6.5.1 廣播傳輸協定存取 API

在 carousel 中用來存取物件的相對檔案名稱，是相對於基底目錄 (base directory)，如「5.9.2 DVB-J application location descriptor」所述。只要呼叫 `new DSMCCObject(".")` 或 `new java.io.File(".")` 就會產生一個物件指到「5.9.2 DVB-J application location descriptor」所說的基底目錄。

Caching 的快取規則，必須在 `DSMCCObject.synchronousLoad()` 或 `DSMCCObject.asynchronousLoad()` 載入資料後進行比對，或是因為下一節 (6.5.1.1 `java.io.File` 方法在 broadcast carousel 上的限制) 所說的其他動作而載入資料後進行比對。在 DVB-J 物件載入之後，任何 object carousel 對物件內容所產生的改變，在應用程式使用此 DVB-J 物件來存取其內容時，都將看不到。

6.5.1.1 `java.io.File` 方法在 broadcast carousel 上的限制

應用程式應該可以使用標準的 `java.io.File` 類別，來存取 broadcast carousel (譬如一個 carousel unaware application)。此時將適用下列規定：

`File` 類別的建構函式只要建立一個抽象路徑名稱 (abstract pathname) 的實例，同時不應以同步方式存取 broadcast carousel 以免造成執行緒暫時中斷 (blocked)。

在 DVB-J `File` 物件建構函式執行之後，此物件的目錄資訊可能還是處於卸載 (unloaded) 狀態。與此物件相關的資訊 (譬如其長度) 則來自於其父目錄，但此時建構函式不需要載入其父目

錄。不過，如果平台實做將這些資訊放在快取中，此時還是可以取得。

如果一個 DVB-J File 物件的目錄資訊正處於卸載狀態，那麼在下列方法被呼叫時，應該同步載入這些資訊：`canRead()`、`exists()`、`isDirectory()`、`isFile()`。如果載入時失敗，所有這些方法都應該傳回 `false`。

如果內容是處於卸載狀態，當目錄的 `list()` 方法被呼叫時，應該將它同步載入。如果在載入時會造成服務的轉換，就不應該執行，那麼 `list()` 就應該傳回一個空的清單。

對於包含該檔案的模組，`lastModified()` 方法會傳回 DLL 模組的 `moduleVersion` (將此 8 位元值當成一個正整數)。

在 `FileInputStream`、`FileReader`、`RandomAccessFile` 的建構函式呼叫之後，就算檔案有任何改變，從該檔案物件讀出的資料也反映不出來。

如果將 `list` 方法用在一個非目錄的物件上，則該方法應該傳回 `null`。

如果 `FileInputStream`、`FileReader`、`RandomAccessFile` 建構時所參照到的物件不是一個檔案，則應該產生一個 `FileNotFoundException`。

一個簽署過的檔案若無法通過認證，則應採取通用原則呈報。

一個簽署過的目錄若無法通過認證，則該目錄物件的 `list` 方法應該傳回 `null`。

從 `java.io.File.length()` 所傳回的值可能會不準，因為這個數值是從目錄資訊中取出，而不是計算檔案實際大小而得來。

除非廣播網路訊號指明某檔案需要 `transparent access`，否則，不能保證會傳回最新版本的檔案。

6.5.1.2 需要寫入權限的方法

在 `java.io.File` 類別中也有一些方法會假設對檔案系統擁有寫入權限。由於廣播的特性使然，廣播接收器對於 `carousel` 自然是沒有寫入權限。不過要注意，一個 `broadcast carousel` 並非是一個唯讀的檔案系統，也就是並非永遠不會改變。當然，只有廣播業者才能改變 `carousel` 的內容，接收器是不行的。因此，這就好像在 Unix 檔案系統中，某個使用者只有讀取權限，而沒有寫入權限或擁有權。

有幾個 `java.io.File` 的方法會需要目錄的寫入權限：`canWrite()`、`mkdir()`、`makedirs()`、`renameTo()`。

應用在 `broadcast carousel` 中的抽象路徑名稱，應採用下列做法：

`canWrite()` 應該傳回 `false` 來顯示該檔案無法被寫入

`mkdir()`、`makedirs()` 和 `renameTo()` 應該傳回 `false` 來表示動作失敗

6.5.1.3 失去 broadcast carousel 之後的行為表現

當應用程式失去 broadcast carousel 時 (如「2.2.5.3 Loss of Carousel Behaviour」所述), 無法讀取的資料應該採如下做法:

如果某一個檔案從未出現過, 嘗試使用該檔案內容格式專屬的 API 來讀取資料時, 應該會失敗。這應該依個別 API 來執行。

在應用程式一開始時由平台所產生的類別載入器, 在失去起始的廣播檔案系統時, 不要自動回復。

嘗試載入一個從未出現過的類別應該失敗。

在 `InputStream` 成功呼叫之後, 平台就應該維護該 `InputStream` 的資料, 直到該 `InputStream` 被關閉為止。這點和「6.5.1.1 `java.io.File` 方法在 broadcast carousel 上的限制」一節對於 `InputStream` 和檔案版本變更的說明一致。

嘗試為一個失去資料的檔案建立一個 `FileInputStream` 應該失敗, 並產生一個 `FileNotFoundException`。

嘗試為一個失去資料的檔案建立一個 `RandomAccessFile` 應該失敗, 並產生一個 `IOException`。

嘗試使用失去資料的 `FileInputStream` 的 `FileDescriptor`, 將導致 `FileInputStream` 或 `InputStreamReader` 所有的方法都會產生 `IOException`。

嘗試呼叫一個失去資料的 `File` 物件的方法, 將與該 `File` 物件從來不曾存在一樣造成呼叫失敗。

在 broadcast carousel 失去時正在進行的作業應該停止。被暫時中斷 (blocked) 的 Java I/O 作業應該產生 `InterruptedException`。

6.5.2 在廣播通道上支援 Multicast IP

當廣播通道上支援 IP 時, 除了上述沒有 IP 支援時所列出的之外, 還應該支援下類別和封裝:

a) 應支援 Java TV [51] 定義的 `javax.tv.net` 封裝。

- b) 在 `java.net.MulticastSocket` 中的 `send` 方法若遇到 socket 所連結 (bind) 的是一個單向 IP multicast 資料來源時，應該產生一個 `IOException`。
- c) 在 `java.net.DatagramSocket` 中的 `send` 方法若遇到 socket 所連結 (bind) 的是一個單向 IP multicast 資料來源時，應該產生一個 `IOException`。
- d) 應該支援下列 `java.net` 中的類別：

`DatagramPacket`

`SocketException`

`UnknownHostException`

`ProtocolException`

`BindException`

- e) 應該支援 `org.dvb.net.DatagramSocketBufferControl` 類別。

- f) 如果不支援，應該如何：

當應用程式嘗試 `joinGroup()` 到一個 Multicast 位址時，應該要產生一個 `ProtocolException` 來表示無法加入該群組。

不需支援 `java.net.MulticastSocket` 的 `getTTL` 和 `setTTL` 兩個方法。

6.5.3 在返回通道上支援 IP

如果在返回通道支援 IP，則整個 `java.net` 封裝都應支援。若平台不支援某些選擇性的返回通道協定，應該遵照本 API 的定義來讓呼叫失敗。

在一些返回通道能夠連線、也能斷線的裝置上，將一個 `java.net.Socket` 或一個 `java.net.URLConnection` 連結到返回通道上的一個主機位址時，也應該自動建立一個連線到預設連線目標，此目標由擁有預設 ISP 返回通道權限的應用程式指定，而返回通道可以連接或不連接到另外的一個目標，但返回通道資源要保留給其他優先順序較低的應用程式。這樣的連線應該在超過一段時間沒有活動之後自動斷線，逾時時間應該可以透過 MHP 終端機的瀏覽器來設定。應用程式可以利用「`dvb.returnchannel.timeout`」系統屬性來獲得此逾時時間，以帶小數的秒數來計算。

請參閱「6.10 Java 權限」一節。

請參閱「6.8.1.3 其他類別」。

不需支援 `java.net.HTTPURLConnection.HTTP_SERVER_ERROR` 常數。

不需支援 `java.net.DatagramSocketImpl` 類別。

不需支援 `java.net.MulticastSocket` 的 `getTTL` 和 `setTTL` 兩個方法。

在 `java.net.URLStreamHandler` 類別中, `setURL` 這個 `protected` 方法是不一定要有的。

在 `java.net.URLConnection` 類別中, `getDefaultRequestProperty` 和 `setDefaultRequestProperties` 這兩個 `static` 方法都不是一定要有的。

應該支援 `org.dvb.net.DatagramSocketBufferControl` 類別。

註:在「2.3 Interaction Channel Protocols」一節中,是不需在返回通道上支援 `multicast IP`。不支援此功能的 MHP 終端機,相關的方法也就自然依其所定義的方式失敗。

6.5.4 MPEG-2 Section Filter API

MPEG-2 section filter API 定義在 DAVIC 1.4.1p9 [3] 的附錄 E。

11.5.5 中層通訊 API

在一些返回通道能夠連線、也能斷線的裝置上,當 `java.net.Socket` 或 `java.net.URLConnection` 自動產生連線時,應該透過此 API 來通知其他監聽此連線的程式。

在平台實做中,此 API 所建立的連線在一段時間沒有活動之後應該自動斷線,逾時時間跟「6.5.3 在返回通道上支援 IP」一節所自動建立的連線一樣。

如果佔用了一個 `ConnectionRCInterface` 的應用程式在建立連線之後沒有先斷線就釋放該資源,則由實做決定此時 MHP 終端機應該怎麼辦。

如果還有一些 `socket` 正在使用該連線, MHP 終端機應該讓連線在「6.5.3 在返回通道上支援 IP」一節所述的逾時時間到達時自動斷線。

6.5.6 永久性儲存 API

永久性儲存的 API 應該使用 `java.io` 封裝,以及 `org.dvb.io.persistent` 的延伸功能。

可經由 `java.lang.System.getProperty()` 獲得的「`dvb.persistent.root`」屬性，代表永久性儲存檔案名稱空間中的根目錄。這個值，對於所有應用程式來說，應該都是相同的。

若嘗試存取此根目錄的父目錄或更上層目錄中的目錄或檔案，都將產生一個 `SecurityException`，如 `java.io` 封裝的規格所定義。

經過簽署的應用程式若通過認證，並允許存取永久性儲存，則擁有下列權限：

- 對根目錄（定義如上所述）有唯讀權限
- 自動獲得一個「組織」子目錄的讀寫權限，子目錄命名方式為 `<organisation_id>`。
- 自動獲得一個「應用程式」子目錄的讀寫權限，子目錄命名方式為 `<organisation_id> + <file.separator> + <application_id>`。

當一個應用程式的權限申請檔案（`permission request file`）要求存取永久性儲存並獲得允許，MHP 終端機就應負責建立應用程式可以讀寫的目錄（如果這些目錄還未存在）。這些動作應該在此應用程式實例開始存取這些目錄之前完成（包括一些存在測試的方法，例如 `File.isDirectory()`）。

對於取得檔案存取權限的應用程式來說，其必要的目錄就是「組織」和「應用程式」兩個子目錄。對於取得資格證明（`credentials`）的應用程式來說，其必要的目錄就是該證明中的檔案名稱元素（`filename element`）中，到第一個包含萬用字元的目錄為止的上一層目錄。

例如：下面這個檔案名稱元素

```
org0_id/appA_id/external/B_?/-
```

會建立下面這個目錄

```
org0_id/appA_id/external
```

當 MHP 終端機在自動建立「應用程式」子目錄時，應該將此目錄的所有人設為其對應的應用程式，並賦予該應用程式讀、寫的存取權限，同時不讓其他應用程式可以存取。MHP 終端機應該將「應用程式」子目錄下其他自動建立的目錄所有人設為其對應的應用程式，並賦予該應用程式讀、寫的存取權限，同時不讓其他應用程式可以存取。

如果在 MHP 終端機要建立一個「應用程式」子目錄但發覺該目錄已經存在，並由其他應用程式所持有時，MHP 終端機應該將此目錄及其下的所有內容移除，並建立一個空的「應用程式」子目錄，並依上述說明設好所有人和存取權限。

「組織」子目錄的所有人應該是 MHP 平台，其對應的組織應該擁有讀、寫的權限，而所有其他人應該有讀取權限。

在永久性儲存中的所有檔案,應該用建立檔案的應用程式的 48 位元識別碼 (application ID) 當做該檔案的所有人。只有檔案的所有人可以變更檔案的屬性以及檔案的存取權限。檔案的所有人在檔案建立之後就無法改變。

應用程式只能在擁有寫入權限的目錄中建立檔案和目錄。

java.io 封裝中的原有規定仍然有效。

請參閱「8.5 應用程式識別碼的文字編碼」一節。<organisation_id> 和 <application_id> 這兩個欄位,是使用 16 進位數字的文字編碼 (前面不加零),其數值來自於應用程式 48 位元應用程式識別碼中的欄位,請參閱「5.5 Application identification」一節。

註 1: 如「6.5.1 廣播傳輸協定存取 API」所定義,用來存取永久性儲存的相對路徑,無法在應用程式之間交互使用。註 2: 未經簽署的應用程式,無法存取永久性儲存。

存放在永久性儲存中的內容,除非遇到下列狀況,否則將永久保存。

- a) 當永久性儲存的空間不足以滿足正在執行的應用程式需求。
- b) 永久性儲存中的檔案過期。
- c) 檔案被建立的應用程式或是有足夠權限的應用程式清除 (也就是被刪除)。
- d) 檔案被終端使用者清除。
- e) MHP 應用程式所使用的永久性儲存超過其他資源要符合條件的最低要求列數值的 75%。

註: 這可以讓 MHP 終端機的資源管理系統可以依照實做的資源管理政策來釋放檔案空間。

在每一對 (organization ID, application ID) 所定義的範圍中,對於優先順序較高的檔案,終端機應該給予較高的保存期限。有關優先順序的資訊,請參閱 `org.dvb.io.persistent.FileAttributes`。

至於關於清除 (也就是刪除) 和過期的檔案應如何釋放,則隨實做而定。

MHP 終端機至少應該清除其他資源要符合條件的最低要求。此機制應該可用於條件符合測試 (conformance test), 而不受上述規則限制。

有些 MHP 終端機可能會限制一個檔案同一時間只能由一個 `FileOutputStream` (或其他檔案寫入物件) 開啟。在這種情況下,`FileOutputStream` `RandomAccessFile` 和 `FileWriter` 的建構函式在遇到檔案已經被開啟時,就會失敗。不過,存取相同永久性儲存檔案的應用程式不應該依賴這樣的機制,而應該自行協調好永久性儲存的存取。

6.6 服務資訊與服務選擇 API

6.6.1 DVB 服務資訊 API

6.6.2 服務選擇 API

服務選擇 API 是定義在 Java TV [51] 的 `javax.tv.service.selection` 封裝當中。

當 `ServiceContext.getServiceContentHandlers` 第一次被呼叫時,任何傳回的 JMF 播放器,都必須已經處於 `Started` 狀態,而且,若該播放器已經在播放視訊,則會在背景視訊裝置上播放。對於共用同一個時鐘的所有即時媒體元件,應該傳回同一個 JMF 播放器。MHP 終端機應該監控正在呈現的服務元件,並隨時更新服務內容處理器 (service content handler) 的清單。

應用程式應該在服務選擇完畢之後,重新取得最新的服務內容處理器清單。至於,先前服務中的內容處理器是否沿用,則隨實做而定。不沿用的 JMF 播放器就會被停止,並且可以由平台清除。當一個 JMF 播放器被沿用時,應該要重設回預設狀態,就好像新的一樣。尤其,平台實做應該取消所有先前註冊的監聽器,以及一些播放器特性,例如視訊縮放與位置設定。

MHP 終端機永遠不能產生

`javax.tv.services.selection.ServiceContextException`。

在服務選取後的預設媒體播放器行為,定義在「6.4.2.3 預設媒體播放器行為表現」一節。

在 `ServiceContextFactory` 中的 `protected` 建構函式,是給平台實做 (而不是應用程式) 使用的。MHP 應用程式不應該從 `ServiceContextFactory` 建立子類別 (subclassing)。不過,如果應用程式真的這麼做,平台實做不需為其後果負責。

若新選擇的服務已在 MHP 終端機目前所知道的傳輸串流中,但卻不是目前所選的電台,在實做 `ServiceContext.select` 方法時,不需要做選台 (tuning) 的動作。

所有在同一個服務環境中執行的 MHP 應用程式,都能取得並修改這個 `ServiceContext` 中所有服務元件的服務元件處理器 (service component handler) 狀態。若某個服務元件處理器是一個 JMF 播放器,則改變其狀態將不只改變播放器本身的狀態,也會改變其 JMF 控制元件的狀態。

至於不在該服務環境中執行的應用程式 (例如:MHP 瀏覽器) 不應該能夠修改服務元件處理器的狀態,除非事先通知服務環境內的應用程式。要進行這項通知,若服務元件處理器就是 JMF 播放器,可傳送一個 `org.davic.media.ResourceWithdrawnEvent` 給所有目前已註冊監聽

器、希望收到 JMF 播放器 ControllerEvent 的應用程式。

一旦 ResourceWithdrawnEvent 送出之後，服務環境中的應用程式若再對服務元件處理器的狀態進行變更，都將被 MHP 終端機所忽略。同時，查詢狀態的方法應確實傳回目前狀態，而變更狀態的方法則將失敗而無任何顯示。至於與 JMF 相關的事件是否還要傳送給應用程式，則視實做而定。

如果在服務環境之外執行的應用程式不再需要對服務元件處理器的狀態做任何改變，則應該將這項權利完全交還給服務環境內的應用程式。在服務環境內的應用程式此時應該收到通知權利已經返還。要進行這項通知，若服務元件處理器就是 JMF 播放器，可傳送一個 `org.davic.media.ResourceReturnedEvent` 給所有目前已註冊監聽器、希望收到 JMF 播放器 ControllerEvent 的應用程式。

在服務環境內執行的 MHP 應用程式，在收到 ResourceReturnedEvent 之後，應該負責將服務元件處理器的組態還原到應用程式需要的狀態。MHP 終端機沒有必要自動幫服務元件處理器還原到先前的組態。若服務元件處理器就是 JMF 播放器，其組態就是一組可以透 Player 的 Control 來存取的可寫入屬性。

註：譬如視訊縮放、視訊位置、音訊語言選擇等屬性。

註：一種可能發生這種情形的狀況，就是當 MHP 終端機使用者打開瀏覽器畫面、做一些動作、再關閉瀏覽器，此時控制權就會交回 MHP 應用程式。

註：這些事件的產生與 Xlet 的狀態模型不需要有任何關聯。MHP 終端機可以在使用者打開瀏覽器時，將所有處於 Active 狀態的 MHP 應用程式設為 Paused 狀態，但本規格對此並沒有要求。

一個執行中的 DVB-J 應用程式，應該看成是應用程式所在的 ServiceContext 中所呈現的服務元件。它應該有一個 ServiceContentHandler，但本規格對此刻意不做規定，不過，這個 handler 不能是一個 ServiceMediaHandler。至於呼叫本 ServiceContentHandler 上的 getServiceContentLocators 方法會產生什麼結果，則視實做而定。如果 DVB-J 應用程式所在的 ServiceContext 被停止，DVB-J 應用程式就應該和該服務環境中的服務元件一樣，也被停止。當不只一個 DVB-J 應用程式在執行時，其 ServiceContentHandler 是不能在不同的 DVB-J 應用程式間共用。

註：一個 Xlet 並不是一個 `javax.tv.service.navigation.ServiceComponent`，傳送 Xlet 的 object carousel 所使用的基礎串流，應該至少由這些其中之一來表示。

註：只有 DVB-J 應用程式而沒有媒體元件的服務，就不會有 ServiceMediaHandlers，而只會有代表 DVB-J 應用程式的 ServiceContentHandlers。

註：本規格刻意不定義任何需要產生 `SelectionFailedEvent(MISSING_HANDLER)` 的狀況。

註：如果停止一個服務的所有元件（包括媒體和 `Xlet`），就會變成一個沒有任何元件呈現的服務，就好像一開始就選了一個沒有任何元件的服務一樣。

有下列串流類型（來自 `javax.tv.service.navigation.StreamType`）可供選取，並且應該有對應的 `ServiceContentHandlers`：AUDIO、VIDEO、SUBTITLES。

如果所選取的服務是一個 DVB NVOD reference service，MHP 終端機應該自行挑一個與該 reference service 關聯的 NVOD time shifted service，然後嘗試選擇此 time shifted service。其挑選方式應由平台實做自動決定，並且不應讓使用者選取。若是無法執行，則應按照本 API 所指定的方式回報。

註：應用程式若希望選擇某特定的 time shifted service，則應該明確指定該服務，而不要讓接收器代為選擇。

6.6.3 選台 API

選台 API 是定義在 DAVIC 1.4.1p9 [3] 的附錄 H，除了 H.4 (Locator 和 `DvbLocator` 類別) 以外，後者定義在「6.7.6 內容參照」。

當應用程式沒有 `org.dvb.net.tuning.TunerPermission` 時（只限此時），本封裝內的下列方法應該產生一個 `java.lang.SecurityException`：

- `NetworkInterfaceController.reserve`
- `NetworkInterfaceController.reserveFor`

本規格並不需要 `org.davic.net.tuning.dvb.DvbNetworkInterfaceSIUtil` 類別。

請參閱「6.8.3 額外的權限類別」。

若其他 MHP 中的 API 自動執行選台動作，則應該透過選台 API 通知其他正在監聽選台動作的應用程式。

6.6.4 條件性存取 API

條件性存取 API 是定義在 DAVIC 1.4.1p9 [3] 的附錄 I。

有幾個類別無法依照「6.2.2 子集的處理」一節所說的方式支援：`CA1Module`、`CA0Module`、

CA1Message、CA0Message、CA1ModuleResponseEvent 和 CA0ModuleResponseEvent。

實體 CI 模組 (physical CI module)，或是遵循 CI 協定的嵌入式系統 (embedded system) 可以產生 MMI 訊息。如果應用程式希望收到，API 實做在安全規範下，應該傳遞那些必須由應用程式所呈現的。否則，就應產生 CA 對話方塊。

本 API 中的下列方法可以產生 `java.lang.SecurityException`：

`CAModuleManager.addMMIListener`

- `CAModule.queryEntitlement`
- `CAModule.listEntitlements`
- `CAModule.buyEntitlement`
- `CAModule.openMessageSession`

在傳遞 CI 訊息時所使用的 `sessionID` 只能在單一應用程式中使用，不得跨應用程式分享。

請參閱「6.8.3 額外的權限類別」。

當 CI 模組發出 Host Control tune 請求，就會啟動服務選擇。使用 Host Control replace / clear_replace 請求與使用 `javax.tv.media.MediaSelectControl` 有相同的效果。

6.6.5 非協定專屬的 SI API

非協定專屬的 SI API 定義在下列 Java TV [51] 的封裝中：

- `javax.tv.service`
- `javax.tv.service.guide`
- `javax.tv.service.navigation`
- `javax.tv.service.transport`

如果發出的請求已經過了 `pending` 的狀態，將無法取消。在其他情況下是否能夠取消，則由實做決定。至於在請求的 SI 資料已經到達裝置之後，到 `notifySuccess` 方法開始執行之前，實做不需支援取消該請求。

本規格中定義的任何物件，都不需實做 `javax.tv.service.ServiceMinorNumber` 介面。

6.7 一般基礎架構 API

6.7.1 DVB-J 應用程式生命週期支援 API

本 API 由 Java TV [51] 所定義的 `javax.tv.xlet` package 中的類別和介面所組成。

6.7.1.1 Xlet 屬性

下列具名的 Xlet 屬性必須支援：

`dvb.org.id`

- `dvb.app.id`
- `dvb.caller.parameters`

這些屬性，只要呼叫 `getXletProperty` 然後傳入上面所列的名稱字串，就能從 Xlet 的 `XletContext` 中取得。

所有 `XletContext.getXletProperty` 所用到的名稱，只要是以「`dvb.`」開頭的，都將保留給 DVB 規格使用。

從 `XletContext.getXletProperty(XletContext.ARGS)` 所傳回來的字串陣列，應該是 DVB-J 應用程式描述器中所定義的字串陣列（請參閱「5.9.1」一節），順序也必須跟通知訊號中的一致。該描述器中的每一個項目，在本方法傳回的字串陣列中，應以一個字串表示，並使用「10.4.8 Text encoding in AIT」中所定義的編碼方式來解讀。通知訊號中長度為零的字串，應該以一個空字串表示。

`XletProperty` 中的「`dvb.caller.parameters`」屬性包含一個啟動本程式的應用程式所傳入 `AppProxy.start(String[])` 方法的 `String` 陣列。如果本應用程式是由系統啟動，或是由另外一個應用程式透過 `AppProxy.start()` 方法來啟動（沒有任何參數），那麼這個 `XletProperty` 屬性就會傳回一個空的陣列，陣列長度為零。

表 6-2：屬性名稱 / 類型 / 編碼方式

屬性名稱	<code>getXletProperty</code> 回傳值類型
<code>dvb.app.id</code>	根據「8.5 應用程式識別碼的文字編碼」來編碼的 <code>String</code>
<code>dvb.org.id</code>	根據「8.5 應用程式識別碼的文字編碼」來編碼的 <code>String</code>
<code>dvb.caller.parameters</code>	<code>String[]</code> ，如果沒有資料，就以長度為零的陣列表示。

6.7.1.2 DVB-J 應用程式在 destroy 方法中要採取的動作

Xlet 在其 Xlet.destroyXlet 方法呼叫 XletContext.notifyDestroyed 之前，應該至少採取下列動作：

促使 Xlet 所產生的所有執行緒都自行結束。請參閱「java.lang 封裝」一節。

將 Xlet 所產生的 JMF 播放器停止、釋放記憶體、並關閉。

將 Xlet 所產生 JavaTV 服務選擇 ServiceContext 物件停止並摧毀 (destroy)。

釋放其他 Xlet 所產生的其他稀有資源 (如果這些資源執行選台動作的話)，譬如 NetworkInterfaceControllers。

使用 Image.flush() 方法清空所有影像。

Xlet 在 Xlet.destroyXlet 方法中，不應該造成任何延遲。

解除任何監聽器註冊

如果應用程式沒有釋放這些資源，則平台應該代為釋放。

不論是由應用程式或是由平台來釋放資源，都應發出適當的通知給其他已註冊相關監聽器的 MHP 應用程式。對於那些經由 org.davic.resources 封裝的 API 來管理的資源，相關的通知也在該 API 中指定。對於本身也是 ServiceMediaHandler 的 JMF 播放器，如果 MHP 終端機實做變更其相關資源的組態 (譬如進行整理)，則應該遵循「6.6.2 服務選擇 API」中，針對服務環境外執行的應用程式所做的通知規定。

6.7.2 應用程式發掘與啟動 API

下列屬性與 AppAttributes.getProperty 搭配使用：

表 6-3：應用程式屬性

屬性名稱 (註 1)	回傳值
dvb.j.location.base	傳回包含 DVB-J 應用程式位置描述器 base_directory_bytes 欄位的 String。
dvb.j.location.cpath.extension	傳回從 DVB-J 應用程式位置描述器的 classpath_extension_bytes 欄位所產生 String[]，陣列中每一個字串代表一個

	classpath_extension_bytes 當中的路徑名稱。
dvb.transport.oc.component.tag	傳回一個 Integer，該數值就是傳輸協定描述器 (transport protocol descriptor) 中 selector_byte 的 component_tag。如果一個遠端應用程式的 AIT 中有多個傳輸協定描述器，則由實做自行決定要傳回哪一個。
註 1：以「dvb.」開頭的屬性皆保留給將來使用。	

對於一個 AIT 通知訊號中的應用程式，任何會傳回這些應用程式資料庫中欄位的方法，都應該使用下表所定義的資訊來源。本規格此版本並不要求支援透過其他方法通知的應用程式。

表 6-4：AppAttributes 的方法所應使用的資訊來源

方法	資訊來源
getName()	在 application name descriptor 中的名稱。
getName(String ISO639code)	在 application name descriptor 中的應用程式名稱，對應於所指定的語言。
getNames()	在 application name descriptor 中的所有應用程式名稱，以及這些名稱對應的 ISO 639 語言代碼。
getProiles()	在 application descriptor 中的所有 application_profile 欄位。
getPriority()	在 application descriptor 中的 application_priority 欄位。
getVersions(String profile)	在 application descriptor 中，所指定的 application_profile 的 version.major、version.minor 和 version.micro 欄位。
getIsServiceBound()	如果 application descriptor 中的 service_bound 欄位是 1，則傳回 true，否則傳回 false。
isStartable()	此方法沒有資訊來源，其傳回的值需按該方法說明中的方式獲得。遠端應用程式的定義，是 transport protocol descriptor 中所標記為遠端應用程式的程式。
getIdentifier()	在 application information section 中的 application_identifier 欄位。
getServiceLocator()	只有在通知訊號內的 application_control_code 為 REMOTE 時，應用程式才算是從遠端應用程式。一個遠端應用程式的定址器，應該包含 transport protocol descriptor 中的 selector_bytes。
isVisible()	如果 application descriptor 中的 visibility 欄位是「11」，則傳回 true，否則傳回 false。

表 6-5 : Applcon 的方法所應使用的資訊來源

方法	資訊來源
getLocator()	在 application_icons_descriptor 中的 icon_locator_byte 資料，這些資料要加在 DVB-J 應用程式基底目錄之後 (dvb_j_application_location_descriptor 中的 base_directory_byte)，或是加在 DVB-HTML 應用程式的實體根目錄之後 (dvb_html_application_location_descriptor 的 physical_root_bytes)。
getIconFlags()	在 application_icons_descriptor 中的 icon_flags 欄位。

當應用程式在 AIT 中只有 external application authorisation descriptor，將被視為由外部授權的應用程式。

6.7.3 應用程式交互通訊 API

本 API 由 java.rmi.Remote 和 java.rmi.RemoteException 介面，以及 java.rmi.NotBoundException 和 java.rmi.AlreadyBoundException 類別所組成，如 JAE 1.1.8 API [31] 和 org.dvb.io.ixc 封裝所定義。

本規格加入兩個具名的 Xlet 屬性:dvb.org.id 和 dvb.app.id。要從 Xlet 的 XletContext 獲得這兩個屬性，可以分別呼叫 getXletProperty("dvb.org.id") 和 getXletProperty("dvb.app.id")。

6.7.3.1 遠端呼叫的運作

有兩種方法可以將一個物件傳給另外一個 Xlet：

傳遞一個遠端物件的參照

或是

傳遞整個遠端物件的副本

這兩種方式分別稱為「以遠端參照傳遞」(pass by remote reference) 和「以遠端副本傳遞」(pass by remote copy)。如果某個物件已經由 org.dvb.io.ixc.IxcRegistry 的方法連結到 (bound to) IXC Registry，要讓另外一個 Xlet 匯入該物件，就應該以遠端參照傳遞。

6.7.3.1.1 以遠端參照傳遞的物件

一個物件若要以遠端參照傳遞，就必須實做一個遠端介面 (remote interface)。所謂遠端介面，就是直接或間接延伸 (extend) java.rmi.Remote 標記介面 (marker interface) 的介面。

要進行遠端呼叫的方法，其參數或回傳值所宣告的類型必須是一個遠端介面，或是一個可序列化 (serializable) 的物件類別。如果一個這個遠端介面是由應用程式所定義，則傳送端的 Xlet 和接收端的 Xlet 都要納入此介面的定義。如果兩端的 Xlet 含有相同名稱的遠端介面，但是，兩邊的介面宣告卻有所不同，在 Xlet 呼叫這個介面來通訊時，會產生什麼結果，將由實做決定。

當一個物件以遠端參照傳遞到另外一個 Xlet，接收端 Xlet 所收到的並不是該物件的直接參照 (direct reference)，而是一個 stub class。這個 stub class 並不是該遠端物件執行時期類型的子類別，而是一個平台產生的中介類別，用來實做參數或回傳值的遠端介面類型。此類別將實做該遠端介面的所有方法，除此之外就沒有其他應用程式能夠存取的類別成員。這些方法叫做「遠端方法」(remote methods)，當這些遠端方法被呼叫時，會將該呼叫轉到原始 Xlet 中的物件，然後在此 Xlet 中執行。

註：一個遠端介面包含該介面上所列出的所有遠端方法，再加上從 superinterface 所繼承的方法。這甚至包括 superinterface 當中沒有直接或間接延伸 java.rmi.Remote 介面的方法。

stub class 的定義應該由平台自動產生。

註：這與傳統的網路 RMI 有些不同，在 RMI 中，stub class 必須由程式開發人員自行透過 rmic 等工具產生。在 Xlet 交互通訊時，如果平台找不到 rmic 或其他離線工具所產生的 stub class，平台應該自動將它們忽略。

一個產生出來的 stub class 應該包含遠端介面類型宣告中的所有方法。所謂遠端介面，就是直接或間接延伸 (extend) java.rmi.Remote 標記介面 (marker interface) 的介面。這些遠端方法必須以下列方式宣告：

除了應用程式自己的例外狀況 (Exception) 之外，每個方法都必須在 throws 子句中宣告一個 java.rmi.RemoteException。

凡是以遠端參照傳遞的參數或是回傳值物件，必須宣告成一個延伸 java.rmi.Remote 的介面，而不是一個實做此遠端介面的應用程式類別。

每個方法的參數類型，必須是一個遠端介面、或是有實做 java.io.Serializable 介面的類別或介面、或是一個單純類型 (primitive type)。

每個方法的回傳值類型，必須是一個遠端介面、或是有實做 java.io.Serializable 介面的類別或介面、或是一個單純類型 (primitive type)、或是 void。

如果遠端方法沒有遵照這些規定，平台就無法產生一個 stub class。當找不到 stub class 時，應該傳回一個 RemoteException 給方法的呼叫者。

6.7.3.1.1.1 遠端物件生命週期的相關問題

當一個 Xlet 被摧毀時，其他的 Xlet 可能還握一些遠端參照指向這個 Xlet 當中的物件。如果有 Xlet 呼叫這些被摧毀的遠端物件方法，平台將無法執行這些方法，因此會產生一個 `RemoteException`。如果在一個遠端呼叫的過程中，接收端的 Xlet 被摧毀了，呼叫這個遠端方法的 Xlet 就會收到一個 `RemoteException`，然後呼叫就被立即終止。如果一個遠端呼叫已經由遠端物件開始執行，而呼叫端 Xlet 卻被摧毀，則這個方法要繼續執行完畢，除非接收端的 Xlet 也被摧毀。

6.7.3.1.1.2 遠端方法呼叫所產生的例外狀況

如果一個遠端方法產生一個例外狀況，該例外狀況就應該在呼叫端的 Xlet 環境中也複製一份。然後將這個副本傳給呼叫者。

6.7.3.1.1.3 重新匯出物件 (Re-exported Object)

從一個 Xlet 傳到另外一個 Xlet 的物件，有可能會再度被傳回原來的 Xlet。這種情況尤其會發生在居中牽線的 Xlet。此時，原來發送端的 Xlet 將收到其當初匯出的物件實例。Xlet 可以用 Java 語言的 `==` 運算來比較當初匯出與現在收到的這兩個物件實例，其比較結果將為 `true`。因此，我們不必提供強制版本 (override) 的 `java.lang.Object.equals()` 和 `java.lang.Object.hashCode()` 方法。

註：這樣的行為與傳統 Java 實做中的網路 RMI 有所不同。在 Sun 的網路 RMI 實做中，原來發送端的 Xlet 會拿到一個遠端 stub 物件，但是 stub 和遠端物件兩者都必須有一個特別版本的 `equals()` 和 `hashCode()` 兩個方法。

6.7.3.1.2 以遠端副本傳遞的物件

若一個沒有實做 `java.rmi.Remote` 介面的類別用於遠端方法的參數或是回傳值，則必須以遠端副本方式傳遞。此外，遠端方法呼叫時所產生的例外狀況物件，也是利用遠端副本的方式，傳遞接收例外狀況物件的 Xlet，如「6.7.3.1.1.2 遠端方法呼叫所產生的例外狀況」一節所述。

當一個物件以遠端副本的方式傳遞，該物件將會經過序列化 (serialize)，變成匯出端 Xlet 環境中的一串位元組 (byte stream)，然後在匯入端環境中反序列化 (deserialize)。序列化的方式如 `java.io.Serializable` 所定義。應用程式所定義的類別也是可以序列化的，不過，在傳送和接收兩端的 Xlet 都必須包含該類別的定義，而且該類別在兩端的外在型態 (external form) 上必須相容。如果在序列化和反序列化的過程中發生錯誤，就必須產生一個 `java.rmi.RemoteException`。

6.7.3.1.2.1 單純類型的處理

若方法的參數和回傳值為單純類型，在傳遞時直接複製其數值。

6.7.3.1.3 類別載入的相關問題

雖然 Xlet 之間可以交互通訊，但是仍舊無法直接在一個 Xlet 中載入另外一個 Xlet 中的類別。沒有一種類別載入器可以從遠端 Xlet 載入一個類別（這與網路 RMI 不同，網路 RMI 會為遠端物件建立一個特別的 `RMIClassLoader`）。因此，在遠端呼叫中所用到的遠端介面和可序列化物件，都必須包含在兩端的 Xlet 當中。如果沒有，平台實做就應該產生一個 `RemoteException` 然後傳給呼叫遠端物件方法的執行緒。

6.7.3.1.4 執行緒使用率

一個遠端方法不一定要在另外一個底層的執行緒中執行。如果一個應用程式呼叫另一個應用程式內的物件方法，而這個方法卻回呼（callback）第一個應用程式當初做遠端呼叫的執行緒，此時，第一個應用程式不一定會發覺這兩個執行緒，其實是同一個 `java.lang.Thread` 物件實例。

如果一個應用程式同時在好幾個執行緒中做遠端呼叫，那麼，遠端執行的過程要看起來是平行處理。

註：這並不是說不能使用 `thread-pooling` 的技巧。尤其，實做可以將這樣的遠端呼叫序列化，只要第一個呼叫在夠短的時間內完成（與 Java 執行緒正常的排程時間相比）。

6.7.3.1.5 遠端物件的資源回收

如果一個沒有被摧毀（non-destroyed）的 Xlet 握有一個可連接到遠端物件的 stub，則該遠端物件就不能被資源回收，除非遠端物件的 Xlet 被摧毀。如果一個匯出的物件，在所有沒有被摧毀的 Xlet 當中已經沒有可以連接到它的遠端 stub 物件，那麼該遠端物件應該視為無法連接（unreachable），因此也就可以將它資源回收。

當一個 Xlet 被摧毀時，其所包含的物件，可能還有其他 Xlet 握有這些物件的遠端參照。此時，這些被參照到的物件，可以解除參照（dereferenced）然後最終被資源回收。如果真的被回收了，那麼這些物件的遠端方法再被呼叫時，呼叫者就會收到一個 `RemoteException`。

6.7.4 MPEG 基本概念

`ApplicationOrigin`

`ElementaryStream`

`Service`

`TransportStream`

`DvbElementaryStream`

DvbService

DvbTransportStream

那些傳回基礎串流、服務或傳輸串流的方法，應該傳回對應的 dvb 子類別（譬如 DvbElementaryStream 等等）。

6.7.5 資源通知

資源通知 API 定義在 DAVIC 1.4.1p9 [3] 的附錄 F。

當一個資源在應用程式沒有釋放之前就被移除，就應該呼叫該資源所有的 ResourceClient 的 notifyRelease() 方法。如果一個資源應該清除的時間到了，就應該呼叫該資源的 release() 方法。本規格中唯一屬於這類資源的就是連線導向 (connection oriented) 的返回通道 (return channel)。

requestRelease 方法的 requestData 參數，應該實做 java.rmi.Remote 介面。在使用了 ResourceProxy 介面的 API 當中，若用來保留資源的方法含有 requestData 參數，該參數也必須實做 java.rmi.Remote 介面，不然就要是 null。如果不是這樣，則在呼叫 requestRelease() 時，就必須使用 null 做為參數。如果請求資源的程式碼與資源當前的所有人是在同一個應用程式內，那麼，實做 Remote 介面並不會變成透過 java.rmi 來執行。

resourceProxy.getClient() 應該永遠傳回應用程式在 ResourceProxy 建立或保留時提供給平台的 ResourceClient。只有當擁有該資源的 ResourceProxy 的該方法被呼叫時，此方法才傳回資源所有人的 ResourceClient。

此處釐清 DAVIC javadoc 有關多個應用程式同時執行時的問題。DAVIC javadoc 中的說明若與此處違背，則以此處說明為準。

6.7.6 內容參照

本 API 由 DAVIC 1.4.1p9 [3] 附錄 H.4 一節中的類別所組成：Locator 和 DvbLocator 類別。同時也包含 Java TV [51] 定義的 javax.tv.locator 封裝。

而 org.davic.net.Locator 類別的宣告將加上：

「implements javax.tv.locator.Locator」

javax.tv.locator.LocatorFactory 的 createFactory() 方法將永遠傳回 org.davic.net.Locator，該類別實做 javax.tv.locator.Locator 介面，並接受 DVB URL 做為輸入（如「8.1 名稱空間對應 (DVB Locator)」一節所定義）。

在本規格中，那些回傳值類型是 `org.davic.net.Locator` 或 `javax.tv.locator.Locator` 的方法，應該傳回一個 `org.davic.net.dvb.DvbLocator` 實例（或是該類別在平台中定義的子類別），所傳回的定址器可以用 DAVIC 1.4.1p9 [3] 中的 DVB locator 語法來表示。此時，傳回的 `DvbLocator` 將包含一個 DVB 服務的識別碼（請參閱「8.9 服務識別碼」）。

在 `javax.tv.locator.Locator.toExternalForm()` 當中，一個 DVB 定址器的正規形式 (canonical form) 定義如下：

對於 `org.davic.net.dvb.DvbNetworkBoundLocator` 的實例來說，這是隨實做而定的。但是，最後產生的字串，仍需可以重建立出一個相同的定址器（譬如透過 JavaTV 的 `LocatorFactory`）。此字串若由 `Xlet` 先儲存起來（譬如存入永久性儲存），在後續使用時仍然必須有效。

對於不屬於上述子類別、並參照了一個服務或是服務內更詳細部分的 `org.davic.net.dvb.DvbLocator` 實例來說，這將是 MHP 規格所訂的格式。如果該實例在建立時有指定選擇性的 `transport_stream_id`，它就必須是外部形式 (external form) 的一部分，否則就不必。

對於不是上述子類別、並只參照了一個傳輸串流而不是一個服務的 `org.davic.net.dvb.DvbLocator` 實例來說，這將是 MHP 規格所訂的格式，再包括 `transport stream id`。

註：由於選擇性的延伸資料（譬如元件標記或事件識別碼）也是外部形式的一部份，因此，在比對兩個定址器時也會用到，所以若一個定址器有延伸資料，而另一個沒有，比對就會失敗。

對於這種定址器，「儘可能」(best effort) 的比對方式，就是完完全全相同。

在 `LocatorFactory` 中的 `protected` 建構函式，是給平台實做（而不是應用程式）使用的。MHP 應用程式不應該從 `LocatorFactory` 建立子類別 (subclassing)。不過，如果應用程式真的這麼做，平台實做不需為其後果負責。

6.7.7 一般錯誤呈報

此 API 是由 DAVIC 1.4.1p9 [3] 附錄 G 中的介面和例外狀況組成：

`NotAuthorizedInterface`

`NotAuthorizedException`

ObjectUnavailableException

ResourceException

TuningException

6.8 安全性

6.8.1 基本安全性

MHP 支援下列在 PersonalJAE [36] 中定義的封裝和類別。

6.8.1.1 java.security

來自於 java.security 封裝：

- AccessControlContext
- AccessControlException
- AccessController
- AllPermission
- BasicPermission
- CodeSource
- GeneralSecurityException
- Guard
- InvalidKeyException
- Key
- KeyException
- NoSuchAlgorithmException
- NoSuchProviderException
- Permission
- PermissionCollection
- Permissions
- Policy
- Principal
- PrivilegedAction
- PrivilegedActionException
- PrivilegedExceptionAction
- ProtectionDomain
- PublicKey
- SignatureException
- SecurityPermission
- SecureClassLoader (註 1)
- UnresolvedPermission

註 1: java.security.SecureClassLoader 類別延伸了 JAE 1.1.8 API [31] 所定義的 java.lang.ClassLoader 版本，因此，java.security.SecureClassLoader 就不需要支援 java.lang.ClassLoader 後續版本所增加的 API。實做可以決定將 SecureClassLoader 設為 abstract (抽象類別)。

MHP 平台在啟動任何 DVB-J 應用程式之前，都應該先安裝一個 `SecurityManager`。

MHP 應該支援下列類別：

- `java.security.KeyFactory`

`KeyFactory.getInstance(String algorithm)` 對於「RSA」演算法，應該傳回一個有效的 `KeyFactory`。

註：由於並不一定要支援 `KeyFactorySpi`，因此，其 `protected` 建構函式將被排除在外（參閱子集的處理規則）

- `java.security.spec.KeySpec`
- `java.security.spec.EncodedKeySpec`

`EncodedKeySpec` 建構函式對 X.509 key encoding 來說必須保證運作。

- `java.security.spec.X509EncodedKeySpec`
- `java.security.spec.InvalidKeySpecException`

6.8.1.2 `java.security.cert`

- `Certificate`
- `CertificateEncodingException`
- `CertificateException`
- `X509Certificate`
- `CertificateExpiredException`
- `CertificateNotYetValidException`

對於一個符合規格的 MHP 終端機來說，在實做 `X509Extension` 時並不需要 `X509Certificate`。

6.8.1.3 其他類別

MHP 也應支援下列其他類別：

- `java.io.FilePermission`

- `java.io.SerializablePermission`
- `java.lang.RuntimePermission`
- `java.util.PropertyPermission`
- `java.net.SocketPermission`
- `java.awt.AWTPermission`

6.8.2 返回通道安全性的 API

本 API 定義在下列 JSSE [60] 封裝中：

`javax.net`

- `javax.net.ssl`
- `javax.security.cert`

註：由於本規格並不要求必須支援伺服器端 socket，實做時

`javax.net.ssl.SSLServerSocketFactory.getDefault()` 方法可以產生適當的 `RuntimeException`，譬如 `ClassNotFoundException`。

對於下列類別中的 `getSupportedCipherSuites` 方法：

- `javax.net.ssl.SSLServerSocket`
- `javax.net.ssl.SSLSocketFactory`
- `javax.net.ssl.SSLSocket`
- `javax.net.ssl.SSLServerSocketFactory`

應該將實做必須支援的 cipher suite 的 cipher suite 欄位中的名稱，從「TLS_」改成「SSL_」。

如果實做還支援其他 TLS IETF RFC 2246 [63] 中所定義的 cipher suite，則名稱是以「TLS_」或「SSL_」開頭則不重要。

6.8.3 額外的權限類別

6.8.4 一般安全問題

除非另有指定，`java.security.Permission` 的子類別在其建構函式當中，不需要檢查輸入的參數。當一個 MHP 平台在建構這些類別物件時，MHP 平台必須傳入有效的數值。如果一個應用程式在建構這些物件時，傳入了無效的數值，此時 MHP 平台應如何處理，本規則刻意不規範。

6.9 其他 API

6.9.1 計時器 (Timer) 支援

本 API 是由 Java TV [51] 的 Timer API 所組成，在 `javax.tv.util` 封裝內。

平台實做必須滿足下列規格：

最小的重複間隔必須小於或等於 40 ms。

刻度需小於或等於 10 ms。

本規格唯一的條件是，當 MHP 終端機再也無法提供任何計時器時，`TVTimer.scheduleTimerSpec` 可以產生 `TVTimerScheduleFailedException`。

6.9.2 使用者設定和偏好 API

下列偏好應該是所有應用程式都能存取的。

User Language

Parental Rating

Country Code

Default Font Size

其他偏好，則需要「讀取」的 `UserPreferencePermission` (請參閱「6.10.2.8 `org.dvb.user.UserPreferencePermission`」)。

6.9.3 應用組態與版本屬性

應用程式可以讀取應用屬性及其版本屬性，來知道平台所支援的應用組態以及該組態的版本 (這樣就知道平台支援什麼功能)。如果平台不支援某個應用組態，則相關的版本屬性將傳回 `null`。

更確切一點，在表 6-6 中所列的屬性，將包含在 `java.lang.System` 類別的屬性集內。因

此，這些屬性就能夠透過 `java.lang.System.getProperty()` 來讀取。既然此 API 傳回的是一個字串，數字就必須依照 `java.lang.Integer.toString(int)` 的方式來編碼。

表 6-6：應用組態和版本查詢相關的系統屬性

屬性	意義	可能數值	範例
<code>mhp.profile.enhanced_broadcast</code>	顯示是否支援增強型廣播應用組態。	「YES」、「NO」	YES
<code>mhp.profile.interactive_broadcast</code>	顯示是否支援互動式廣播應用組態。	「YES」、「NO」、null	「NO」
<code>mhp.profile.internet_access</code>	顯示是否支援 Internet 存取應用組態。	「YES」、「NO」、null	「NO」
<code>mhp.eb.version.major</code>	所支援的增強型廣播應用組態的大版本 (major version)。	正整數	「1」
<code>mhp.eb.version.minor</code>	所支援的增強型廣播應用組態的小版本 (minor version)。	正整數	「0」
<code>mhp.eb.version.micro</code>	所支援的增強型廣播應用組態的微版本 (micro version)。	正整數	「0」
<code>mhp.ib.version.major</code>	所支援的互動式廣播應用組態的大版本 (major version)。	正整數	「1」
<code>mhp.ib.version.minor</code>	所支援的互動式廣播應用組態的小版本 (minor version)。	正整數	「0」
<code>mhp.ib.version.micro</code>	所支援的互動式廣播應用組態的微版本 (micro version)。	正整數	「0」
<code>mhp.ia.version.major</code>	所支援的 Internet 存取應用組態的大版本 (major version)。	正整數	「1」
<code>mhp.ia.version.minor</code>	所支援的 Internet 存取應用組態的小版本 (minor version)。	正整數	「0」
<code>mhp.ia.version.micro</code>	所支援的 Internet 存取應用組態的微版本 (micro version)。	正整數	「0」
註 1：在本規格先前的版本中，不支援的應用組態可能會傳回「NO」而不是 null。			

如果 MHP 終端機根本不支援某一個應用組態，就不應該支援其版本屬性的查詢。因此，這時候 `System.getProperty()` 應該傳回 null。

6.9.3.1 選項資訊

為了讓應用程式知道某個 MHP 平台實做支援哪些選項，每一個選項都定義了一個屬性字串。

MHP 平台實做若支援某個選項，就必須認得其對應屬性，而且其屬性值必須為「SUPPORTED」。

這些屬性都在 `java.lang.System` 的屬性集內。

這些用來顯示某些功能是否支援的屬性，其語法大致為：

```
mhp.option.<選項功能>
```

表 6-7 列出目前已經定義的選項功能。

表 6-7：選項功能查詢相關的系統屬性

選項	屬性
在廣播通道上支援 IP Multicast	<code>mhp.option.ip.multicast</code>
在互動通道上支援用戶對用戶 DSMCC	<code>mhp.option.dsmcc.uu</code>
DVB-HTML	<code>mhp.option.dvb.html</code>

6.10 Java 權限

本節說明「沙盒」中給未經簽署應用程式使用的權限，以及權限申請檔案 (permission request file) 可請求的權限，如何對應到 Java 平台中的各種 Permission 物件。

6.10.1 未經簽署應用程式的權限

MHP 安全政策規範中，有一組資源是所有應用程式執行時都可以存取的。所有未經簽署的應用程式，都只能存取到這些資源。

本節定義這些資源如何對應到各種 Java Permission 物件。這些權限應該賦予 DVB-J 應用程式使用。

6.10.1.1 `java.awt.AWTPermission`

這物件控制 AWT 中較為敏感的部分，是 MHP 應用程式所不需要存取的。因此，不管是經過簽署或是未經簽署的應用程式，都不應允許存取。

6.10.1.2 `java.net.SocketPermission`

由於在沙盒中是無法存取返回通道的，因此這個物件對於未經簽署的程式是不需要的。

6.10.1.3 `java.util.PropertyPermission`

除了那些標明只能給經過簽署的應用程式存取的屬性之外，未經簽署的應用程式對於本章所定義的所有的屬性，應該都有「讀取」的權限。不過，卻沒有「寫入」的權限。

至於其他的屬性，除非是特別標明只給經過簽署的應用程式，否則應該也可以提供讀取的權限。

6.10.1.4 `java.lang.RuntimePermission`

此權限不管是未經簽署或是經過簽署的應用程式都不應該擁有。

6.10.1.5 `java.io.SerializablePermission`

此權限不管是未經簽署或是經過簽署的應用程式都不應該擁有。

6.10.1.6 `java.io.FilePermission`

在平台實做掛載 `object carousel` 的節點以下的子樹狀階層，都應該可以提供讀取的權限。

6.10.1.7 `javax.tv.media.MediaSelectPermission`

由於 Media API（也就是 JMF）是在沙盒中執行的，因此，所有應用程式都應該有 `javax.tv.media.`

`MediaSelectPermission` 的權限，其定址器字串為「*」，代表可存取所有媒體串流。

6.10.1.8 `javax.tv.service.ReadPermission`

服務資訊的存取也是在沙盒內，因此，所有應用程式都應該有 `javax.tv.service.`

`ReadPermission` 的權限，其定址器字串為「*」。

6.10.1.9 `javax.tv.service.selection.ServiceContextPermission`

所有應用程式都應該有名稱字串為「`getServiceContentHandlers`」、動作字串為「`own`」的 `javax.tv.service.selection.ServiceContextPermission` 權限。

所有 MHP 應用程式都應該有 `ServiceContextPermission( "access", "own" )` 權限。

但是卻不應該有 `ServiceContextPermission( "access", "*" )` 權限。

6.10.1.10 `java.util.Locale.setDefault`

此方法將產生一個 `SecurityException`。

6.10.2 經過簽署的應用程式的額外權限

經過簽署的應用程式將可以獲得更多的權限。這些額外的權限，必須使用權限申請檔案 (permission request file) 來取得。本節定義權限要求檔案中的項目如何對應到 MHP 終端機可以授與的 Java 權限。

6.10.2.1 java.util.PropertyPermission

一個經過簽署的應用程式，對於未經簽署的應用程式能夠存取的所有屬性以及 `dvb.persistent.root` 應該都有讀取的權限。

6.10.2.2 java.io.FilePermission

在平台實做掛載 `object carousel` 的節點以下的子樹狀階層，都應該可以提供讀取的權限。

如果權限申請檔案中要求存取永久性儲存並獲得允許，就必須為此產生一個 `FilePermission`，來允許存取永久性儲存子樹狀階層。

如果權限申請檔案中包含了一個永久性儲存檔案的資格證明 (credential)，並且獲得允許，就必須依照下列方式來產生 `FilePermission`：

`file path = dvb.persistent.root 屬性值 + 資格證明中的檔案名稱`

`action = 「read」或「write, delete」(如果資格證明中標明的分別是「read」和「write」)`

註：只靠一個 `FilePermission` 實例，並不足以滿足檔案和目錄的存取限制。平台實做有責任貫徹這些要求，因此，不能只依賴一個 `FilePermission` 來達成這項目標。

6.10.2.3 org.dvb.net.ca.CAPermission

如果權限申請檔案中要求與 CA 系統溝通並獲得允許，則應該按下列方式產生一個 `CAPermission`：

在權限申請檔案中的 CA 系統識別碼 (ID) 字串，應該直接做為 `CApermission` 建構函式所需字串的第一部分，然後接著一個英文冒號 (:) 然後接著權限申請檔案中所列的一些屬性的字串 (屬性為 `true` 者)。

6.10.2.4 org.dvb.application.AppsControlPermission

如果權限申請檔案中要求一些額外的權限來控制應用程式的生命週期並獲得允許，則應該為此產生一個 `AppsControlPermission`。

6.10.2.5 org.dvb.net.rc.RCPermission

如果權限申請檔案中要求透過返回通道溝通的權限並獲得允許，則應該依下列方式產生一個 `RCPermission`：

對於預設 ISP 項目，在產生 `RCPermission` 時要指定「`target:default`」字串。

對於包含電話號碼的項目，則應該指定「`target: *`」+ 權限申請檔案中的電話號碼前置碼 +「`*`」。

在 `org.dvb.net.rc.ConnectionRCInterface` 上的 `getCurrentTarget` 方法應該永遠產生一個 `SecurityException`。當應用程式沒有權限使用其指定的 `target` 時，`setTarget` 方法，應該產生一個 `SecurityException`。當應用程式沒有「`target:default`」或「`target: *`」的權限時，`setTargetToDefault` 方法應該產生一個 `SecurityException`。

6.10.2.6 `org.dvb.net.tuning.TunerPermission`

如果權限申請檔案中要求存取選台 API 並獲得允許，則應產生一個 `TunerPermission`。

6.10.2.7 `javax.tv.service.selection.SelectPermission`

在預設情況下，`SelectPermission` 在產生時是使用「`*`」為定址器，並以「`own`」為動作 (action)，而預設的 `ServiceContextPermission` 則是以「`*`」為名稱，以「`own`」為動作，除非權限申請檔案中指定不要如此。

6.10.2.8 `org.dvb.user.UserPreferencePermission`

如果權限申請檔案要求讀取和/或寫入使用者偏好的權限並獲得允許，應該依下列方式產生一個 `UserPreferencePermission`：

如果權限申請檔案在「`read`」屬性上指定了「`true`」並獲得允許，則應使用「`read`」字串來產生 `UserPreferencePermission`。

如果權限申請檔案在「`write`」上指定了「`true`」並獲得允許，則應該使用「`write`」字串來產生 `UserPreferencePermission`。

6.10.2.9 `java.net.SocketPermission`

如果權限申請檔案中要求與遠端主機通訊並獲得允許，則應該依照權限申請檔案中指定的主機與動作來產生 `SocketPermission`。

當返回通道是以 `org.dvb.net.rc.ConnectionRCInterface` 的實例來代表時（也就是說，所有的返回通道都是以連線為導向時），MHP 終端機不應該授與這些權限，除非應用程式同時獲得 `org.dvb.net.rc.RCPermission`。

6.10.2.10 `org.dvb.media.DripFeedPermission`

如果權限申請檔案要求使用涓流傳送功能 (drip feed feature) 並獲得允許，就應該產生一個 `DripFeedPermission`。

6.11 內容參照

底下將說明「表 8-2：可定址個體、定址器、與其文字表示形式」和本章所定義的 DVB-J 方法的對應關係。此處將列出那些在宣告中接受或是回傳 `org.davic.net.Locator`、`javax.tv.locator.Locator`、`javax.media.MediaLocator` 等定址器或其子類別的方法。這些定址器的外部形式 (external form) 應該是「表 8-2：可定址個體、定址器、與其文字表示形式」中所列的文字表示形式。如果此處列出的一個方法能夠接受多種形式的定址器，那該方法就必須能接受此處所列的所有形式。

如果一個方法必須檢查其輸入，那麼，該方法就只能接受特定的定址器形式。該方法應該拒絕其他本處所列的定址器形式。如果一個方法沒有指定如何拒絕不適用的定址器，應該直接失敗，除非是不用檢查輸入參數的 `Exception` 和 `Event`，或平台有責任使用正確的定址器時。本規格並不禁止任何方法接受本規格之外的定址器形式。

下列各節對於是否包含 `network_id` 的定址器都適用。

6.11.1 傳輸串流

本規格中的下列方法，應該接受或傳回表示 MPEG 傳輸串流的 `Object` 實例。那些接受一個傳輸串流定址器當做輸入參數的方法，也應該接受其他包含傳輸串流識別資訊 DVB 定址器。如果定址器中有 `service_id`、`component_tag`、`event_id` 和 `path_segments`，這些資料應該保留不用，除非同時有一個方法可以接受該定址器做為輸入並且有另外一個查詢方法可以傳回該定址器。此時，應該以該查詢方法所指定的意義為準。

- `javax.tv.service.transport.TransportStream.getLocator()`
- `javax.tv.service.transport.TransportStreamCollection.retrieveTransportStream()`

註：本規則只有在 MHP 終端機支援 `TransportStreamCollection` 物件時才適用。

- `org.davic.net.tuning.StreamTable.getTransportStreams()`
- `org.davic.net.tuning.NetworkInterfaceController.tune()`
- `org.davic.net.tuning.NetworkInterface.getLocator()`
- `org.davic.net.tuning.NetworkInterfaceController.reserveFor()`
- `org.davic.net.tuning.StreamTable.listTransportStreams()`

- `org.dvb.si.SITransportStream.getDvbLocator()`

註：在定址器中的識別碼數字，就是對應 DVB-SI 表格中的數字，請參閱「8.1.5 dvb_entity = dvb_transport_stream」一節。

6.11.2 網路

本規格中用到的下列方法應該接受或傳回一個表示 DVB 網路的 Object 實例。

- `javax.tv.service.transport.NetworkCollection.retrieveNetwork()`
- `javax.tv.service.transport.Network.getLocator()`

6.11.3 Bouquet

本規格中用到的下列方法應該接受或傳回一個表示 DVB bouquet 的 Object 實例。

- `javax.tv.service.transport.BouquetCollection.retrieveBouquet()`
- `javax.tv.service.transport.Bouquet.getLocator()`

6.11.4 服務

6.11.4.1 MPEG/DVB 相關服務

本規格中用到的下列方法應該接受或傳回一個表示 DVB 服務的 Object 實例。那些接受一個 MPEG/DVB 服務定址器當做輸入參數的方法，也應該接受其他包含服務識別資訊 DVB 定址器。如果定址器中有 `component_tag`、`event_id` 和 `path_segments`，這些資料應該保留不用，除非同時有一個方法可以接受該定址器做為輸入並且有另外一個查詢方法可以傳回該定址器。此時，應該以該查詢方法所指定的意義為準。

- `org.davic.net.ca.CAModule.buyEntitlement()`
- `org.davic.net.ca.CAModule.queryEntitlement()`
- `org.dvb.si.SIDatabase.retrieveSIService()`
- `org.dvb.si.SIDatabase.retrievePMTService()`

`org.dvb.dsmcc.DSMCCStream.getStreamLocator` (若 `isMPEGProgram` 方法傳回 true)

- org.dvb.dsmcc.ServiceDomain.attach()
- org.dvb.dsmcc.ServiceDomain.getLocator()
- org.dvb.si.PMTService.getDvbLocator()
- org.dvb.si.SIBouquet.getSIServiceLocators()
- org.dvb.si.SIService.getDvbLocator()

org.davic.net.ca.TuneRequestEvent - 建構函式

- org.davic.net.ca.TuneRequestEvent.getLocator()

org.dvb.application.AppAttributes.getServiceLocator() - (若該服務是一個 MPEG / DVB 服務)

org.dvb.dsmcc.ServiceXFRReference - 建構函式 (若該服務是一個 MPEG / DVB 服務)

org.dvb.dsmcc.ServiceXFRReference.getLocator() - (若該服務是一個 MPEG / DVB 服務)

註：在定址器中的識別碼數字，就是對應 DVB-SI 表格中的數字，請參閱「8.1.1 dvb_entity = dvb_service」一節。

6.11.4.2 通用服務

本規格中用到的下列方法應該接受或傳回一個參照通用服務的 Object 實例。這些方法也必須接受或傳回前一節所述的定址器。

javax.tv.service.navigation.LocatorFilter - constructor

- javax.tv.service.navigation.LocatorFilter.getFilterValue()
- javax.tv.service.SIManager.getService()
- javax.tv.service.navigation.ServiceDetails.getLocator()
- javax.tv.service.Service.getLocator()
- javax.tv.service.SIManager.retrieveServiceDetails()
- javax.tv.service.navigation.ServiceList.findService()

`org.dvb.application.AppAttributes.getServiceLocator()` - (若該服務並非 MPEG / DVB 特定服務)

`org.dvb.dsmcc.ServiceXFRReference` - 建構函式 (若該服務並非 MPEG / DVB 特定服務)

`org.dvb.dsmcc.ServiceXFRReference.getLocator()` (若該服務並非 MPEG / DVB 特定服務)

`org.dvb.dsmcc.ServiceXFRException` - 建構函式

`org.davic.media.MediaLocator` - 建構函式

`javax.media.MediaLocator` - 建構函式

- `javax.media.Manager.createPlayer(MediaLocator)`
- `javax.media.Manager.createDataSource(MediaLocator)`

需要 JMF `MediaLocator` 做為參數的各種 JMF 事件的建構函式：

`org.dvb.media.CAStopEvent`, `org.dvb.media.PresentationChangedEvent`,
`org.dvb.media.ServiceRemovedEvent`,
`org.dvb.media.NoComponentSelectedEvent`

會傳回建構函式所得到的 JMF `MediaLocator` 的各種 JMF 事件：

`org.dvb.media.CAStopEvent`, `org.dvb.media.PresentationChangedEvent`,
`org.dvb.media.ServiceRemovedEvent`, `org.dvb.media.NoComponentSelectedEvent`

6.11.5 DVB 事件

本規格中用到的下列方法應該接受或傳回一個表示 DVB 事件的 `Object` 實例。

- `javax.tv.service.guide.ProgramEvent.getLocator()`
- `javax.tv.service.SIManager.retrieveProgramEvent()`
- `org.davic.net.ca.CAModule.buyEntitlement()`
- `org.davic.net.ca.CAModule.queryEntitlement()`

- `javax.tv.service.guide.ProgramSchedule.retrieveProgramEvent()`
- `org.dvb.si.SIEvent.getDvbLocator()`

6.11.6 MPEG 基礎串流

本規格中的下列方法，應該接受或傳回表示 MPEG 基礎串流的 Object 實例。下列接受一個定址器陣列做為輸入參數的方法，應該接受包含多個元件標記 (component tag) 的 DVB 定址器。

- `javax.tv.net.InterfaceMap.getLocalAddress()`

`javax.tv.service.selection.InvalidServiceComponentException` - 建構函式

`javax.tv.media.MediaSelectControl` - 所有接受或傳回 `javax.tv.locator.Locator` 實例的方法

`javax.tv.media.MediaSelectEvent` & subclasses - 建構函式

`javax.tv.media.MediaSelectPermission` - 建構函式

- `javax.tv.service.selection.ServiceContext.select()`
- `org.dvb.si.SIDatabase.retrievePMTElementaryStreams()`
- `org.dvb.si.PMTElementaryStream.getDvbLocator()`
- `org.dvb.dsmcc.ServiceDomain.attach()`
- `org.dvb.dsmcc.ServiceDomain.getLocator()`
- `javax.tv.media.MediaSelectControl.getCurrentSelection()`
- `javax.tv.service.navigation.ServiceComponent.getLocator()`
- `javax.tv.media.MediaSelectEvent.getSelection()` (以及子類別)
- `org.davic.net.ca.DescramblingStoppedEvent.getServiceLocator()`
- `org.davic.net.ca.DescramblingStartedEvent.getServiceLocator()`

`org.davic.media.MediaLocator` - 建構函式 - 同時應該接受包含多個元件標記的「dvb:」定址器

`javax.media.MediaLocator` - 建構函式 - 同時應該接受包含多個元件標記的「dvb:」定址器

`javax.media.manager.createPlayer( MediaLocator )` - 同時應該接受包含多個元件標記的「dvb:」定址器

- `javax.tv.service.selection.InvalidServiceComponentException.getInvalidServiceComponent()`

- `javax.tv.service.selection.ServiceContentHandler.getServiceContentLocators()`

註：在定址器中的識別碼數字，就是對應 DVB-SI 表格中的數字，請參閱「8.1.5 dvb_entity = dvb_service_component」一節。

6.11.7 檔案

本規格中用到的下列方法應該接受或傳回一個參照到檔案的定址器。

`org.davic.media.MediaLocator` - 建構函式 - 針對要從記憶體播放的音訊檔案

`javax.media.MediaLocator` - 建構函式 - 針對要從記憶體播放的音訊檔案

`javax.media.Manager.createPlayer(MediaLocator)` - 針對要從記憶體播放的音訊檔案

`javax.media.Manager.createDataSource(MediaLocator)` - 針對要從記憶體播放的音訊檔案

`org.dvb.dsmcc.ServiceDomain.getURL(Locator)` - 為「dvb:」定址器的實例，包括 `dvb_abs_path`

除上述之外，所有檔案參照都必須以 `java.net.URL` 實例包裝 (encapsulate)。

6.11.8 目錄

下列方法應該傳回參照到目錄的一個「dvb:」定址器。

- `org.dvb.application.AppIcon.getLocator()`

6.11.9 涓流解碼器

本規格中用到的下列方法應該接受或傳回一個參照到涓流解碼器的定址器。

`javax.media.MediaLocator` - 建構函式

- `javax.media.Manager.createDataSource(MediaLocator)`

6.11.10 不相干的方法

本規格中用到的下列方法，雖然在其宣告中用到 `Locator` 物件，但這並不是本規格的要求。

`javax.tv.service.ReadPermission`：所有應用程式此權限都應該是「*」（請參閱「6.10.1.8 `javax.tv.service.ReadPermission`」一節）。

`javax.tv.service.selection.SelectPermission`：應用程式此權限不是「*」就是根本沒有此權限（請參閱「6.10.2.7 `javax.tv.service.selection.SelectPermission`」一節）。

6.11.11 適用各種定址器類型的方法

本規格中用到的下列方法，適用於各種定址器類型。每一個方法都列出其適用的定址器類型。

`javax.tv.locator.LocatorFactory.transformLocator` - 將與傳輸無關的定址器轉成某一種傳輸方式的定址器

必須能接受 `org.davic.net.dvb.DvbLocator` 實例

必須能接受 `org.davic.net.dvb.DvbNetworkBoundLocator` 實例

`javax.tv.locator.LocatorFactory.createLocator` - 從一個字串產生一個定址器

必須能接受有效的「dvb」URL（請參閱「8.1 名稱空間對應（DVB Locator）」一節），並傳回對應的 `org.davic.net.dvb.DVBLocator` 實例

`javax.tv.service.SIManager.registerInterest` - 接受一個參照到多個 `SIElement` 的定址器做為輸入參數

`javax.tv.service.SIManager.retrieveSIElement` - 接受一個參照到多個 `SIElement` 的定址器做為輸入參數

這兩個方法都必須能接受參照到下列物件的定址器：Bouquet、Network、Event、

ElementaryStream、Service、TransportStream。

`javax.tv.service.SIElement.getLocator`

傳回「this SIElement」的定址器，如 JavaTV 指定的子介面 (sub-interface) 所述，沒有其他 SIElement 存在

6.11.12 DVB-J 對 HTTP 協定的支援

如果平台支援 HTTP 協定 (請參閱「2.3.7.1 HTTP 1.1」一節)，下列類別和方法就應該支援 HTTP URL，如下所述：

`javax.media.MediaLocator` 的建構函式 - 用於參照從記憶體播放的音訊檔案

`javax.media.Manager` 上接受 `javax.media.MediaLocator` 做為輸入參數的方法 - 用於建構從記憶體播放音訊檔案的 JMF 播放器

`java.net` 封裝中的類別和方法

本規格中接受 `java.net.URL` 做為輸入參數的方法，必須要接受包裝 (encapsulate) 「http」URL 的實例，並依照其規格運作。 - 譬如 `Toolkit.getImage(java.net.URL)`

當 HTTP URL 與 `DVBClassLoader` 實例搭配使用，在經過簽署的應用程式中透過互動式通道載入 DVB-J 類別時，仍應該遵守 MHP 安全模型的規範，才能從這樣的 URL 成功載入該類別。

7 繪圖參考模型

7.1 概述

MHP 提供了一些工具來進行各種定位控制：輸出裝置上的視訊、使用者介面元件（譬如按鈕和清單）、以及基礎繪圖物件。

每一個連接到 MHP 的螢幕都有三個顯示平面（plane），從後往前算依序為背景平面、視訊平面、繪圖平面。

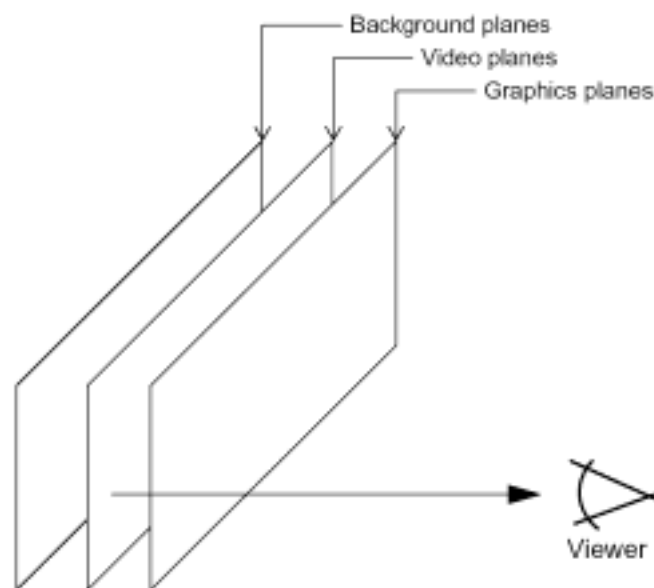


圖 7-1：三種顯示平面示意圖。

至於字幕平面（subtitle plane）則依實做而有所不同。不過，應用程式仍可以透過 API 讓這部分達到預期的效果。請參閱「7.5 字幕」一節。

應用程式所擁有的，是一個四方形繪圖區域，由各個繪圖平面重疊組成，應用程式可在這個四方形區域內繪圖（請參閱「7.3.3 HAVi 裝置和 AWT 元件」一節）。應用程式可以在四方形的繪圖平面上放置視訊影像、使用者介面元素、以及其他繪圖物件。

應用程式也能在 AWT 階層架構之外控制視訊平面，或是在背景平面上放置靜態影像、視訊涓流（video drip）、或是實色圖案。

在 MHP 規格中，終端機可以支援多個應用程式同時執行，每一個應用程式在螢幕上有自己的一塊繪圖區域。規格中也允許這些區域可以重疊。不過，這些功能的最低支援需求是隨不同應用組態（Profile）而定。

本參考模型繪圖平面與 MHP 終端機硬體繪圖平面之間的對應（mapping）方式，必須視實做而

定。舉例來說，我們當然可以將終端機硬體的「視訊」平面當作 MHP 模型中的「背景」平面，來放置 MPEG 靜態影像和視訊涓流。

7.1.1 應用程式間的互動

當螢幕上不同的應用程式畫面互相重疊時，被其他應用程式擋到的區域就會被裁切 (clipped)。如果應用程式畫面是半透明的，其畫面內容只會和該應用程式後面的視訊或背景影像混合 (blend)，而不會跟其他應用程式畫面混合。

7.2 一般問題

7.2.1 座標空間

MHP 包含許多座標系統，每一種都有其用途，座標系統之間也有轉換的方法：

- 輸入視訊空間

- 這代表向上取樣 (upsampling) 之後的 MPEG 像素 (pixel) 平面。

- 裝置空間

- 各種顯示裝置上的邏輯像素。不同類型的裝置 (例如：視訊和繪圖) 會有不同的裝置空間

- 正規化 (normalised) 螢幕空間

- 相對於輸出裝置 (HScreen) 的正規化座標。

繪圖與視訊系統中的不同元件，必須利用不同的介面和這些座標空間來存取。

7.2.1.1 正規化螢幕空間

使用正規化的螢幕座標系統，就能指定 MHP 裝置視訊輸出中的位置跟大小，不需指定像素型態。

在這個座標系統中，螢幕的左上角座標為 {0, 0}，螢幕的右下角座標為 {1, 1}。在

org.havi.ui 封裝 (package) 的一些類別 (class) 中，就是用這個座標系統來指定繪圖與視訊的位置，同時此座標系統也用於 JMF 控制元件

org.dvb.media.VideoPresentationControl 當中。

正規化的座標系統是透過 HScreen 來存取。

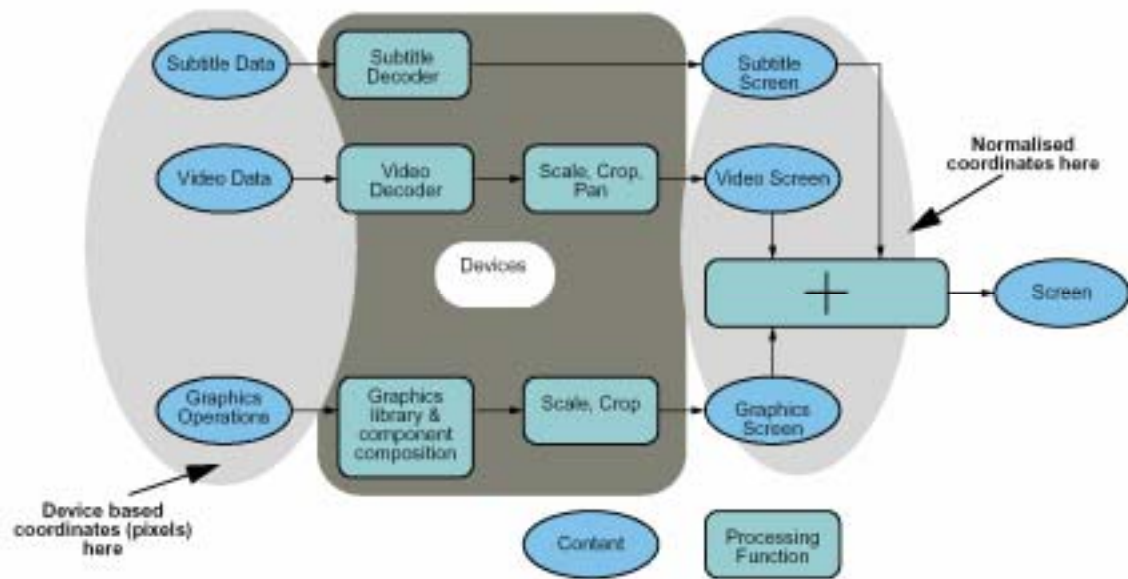


圖 7-2

7.2.1.2 使用者空間 (User Space)

在 java.awt 中所使用的繪圖座標系統，是由應用程式在建立 HGraphicsDevice 時所定義的。其根容器 (root container)，也就是 org.havi.ui.HScene 類別的實例 (instance)，可以放在正規化座標系統中。HSceneTemplate 類別可讓應用程式指定根容器的需求。HSceneFactory 會用這個類別的實例來傳回一個 HScene。

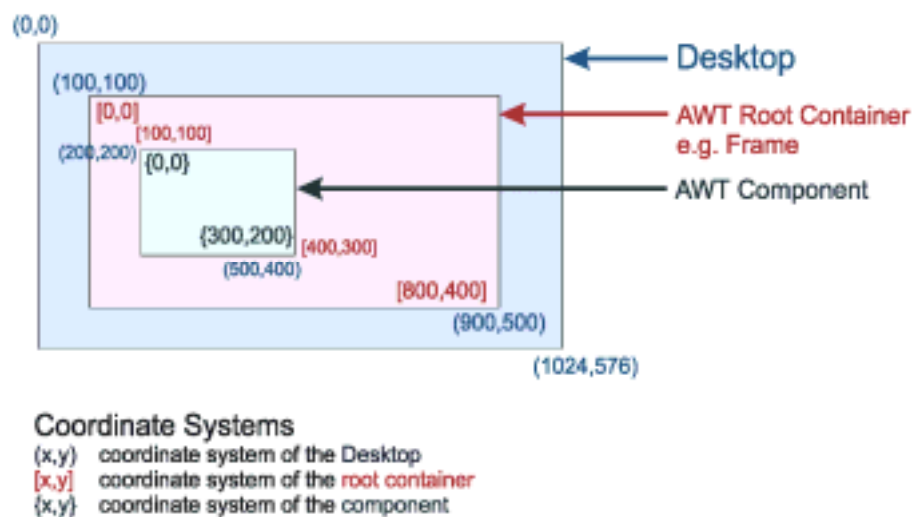


圖 7-3：AWT 座標系統在電腦環境的應用

圖 7-3 顯示在一般電腦環境中的 AWT 座標系統。在一個 MHP 裝置上，HGraphicsDevice 可以看做是桌面 (Desktop)，而 HScene 則是 AWT RootContainer。因此 HScene 是在

HGraphicsDevice 的座標系統內。HScene 本身定義了一個新的座標系統，其中的像素是與 HGraphicsDevice 對齊的，但是原點卻已經過轉換（如圖 7-3）。

要從使用者空間轉到正規化座標系統時，必須指定 HGraphicsDevice 在正規化座標系統中的大小和位置，以及 HGraphicsDevice 的像素解析度。

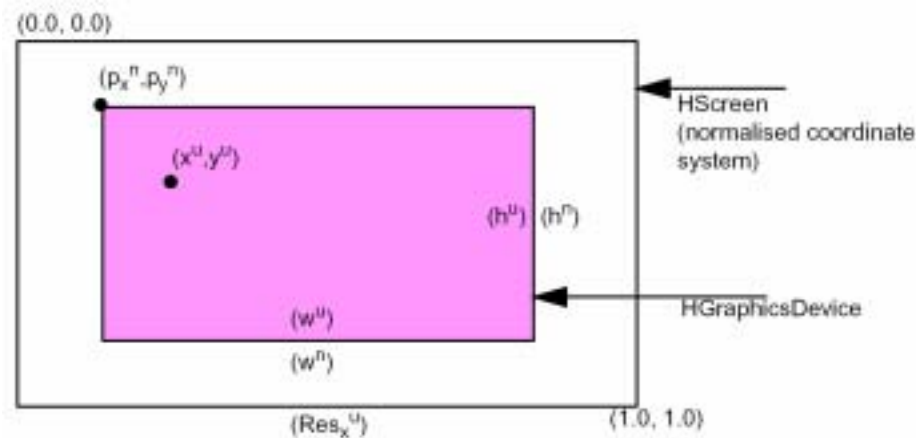


圖 7-4：正規化座標系統與使用者座標系統轉換

(x_n, y_n) = HScreen 的正規化座標系統

(x^u, y^u) = HGraphicsDevice 的座標系統（以像素為單位）

x^{vu} = 相對於 HScreen 的「虛擬 (virtual)」使用者像素座標系統（以像素為單位）

假設 HGraphicsDevice 的座標原點在 (p_x^n, p_y^n) ，寬高為 (w^n, h^n) ，像素解析度為 $w^u \times h^u$ ，那麼，一個點在正規化座標系統上的位置 (x^u, y^u) 計算方法為：

$$x^n = \frac{x^{vu}}{Res_x^{vu}} \quad y^n = \frac{y^{vu}}{Res_y^{vu}}$$

其中：

$$Res_x^{vu} = \frac{w^u}{w^n} \quad Res_y^{vu} = \frac{h^u}{h^n}$$

也就是 HScreen 的虛擬解析度，以 (sub)pixel 表示，而：

$$x^{vu} = p_x^{vu} + x^u \quad y^{vu} = p_y^{vu} + y^u$$

是點 (x, y) 在虛擬座標系統的位置，且：

$$p_x^{vu} = p_x^n \cdot Res_x^{vu} \quad p_y^{vu} = p_y^n \cdot Res_y^{vu}$$

是 HGraphicsDevice 在虛擬座標系統中的位置。

當給定一個點 (x^n, y^n) ，下列公式可求得點 (x^u, y^u) ：

$$x^u = floor[(x^n - p_x^n) \cdot Res_x^{vu} + 0.5]$$

且

$$y^u = floor[(y^n - p_y^n) \cdot Res_y^{vu} + 0.5]$$

當給定一個 HGraphicsDevice，若其為全螢幕大小，公式則可以簡化為：

$Res_x^{vu}=w^u$ ， $Res_y^{vu}=h^u$ ， $x^{vu}=x^u$ ， $y^{vu}=y^u$ ，因此：

$$x^n = \frac{x^u}{w^u} \quad y^n = \frac{y^u}{h^u}$$

且

$$x^u = x^n \cdot w^u \quad y^u = y^n \cdot h^u$$

在上述的計算過程中，應該使用浮點數 (float) 來計算才夠精確。若要轉成整數，應該在以像素為單位的座標系統上才做，也就是使用者空間。

要指定一個 HGraphicsDevice 的解析度，可透過 setPreference(int, Object, int) 方法，以常數 HScreenConfigTemplate.PIXEL_RESOLUTION 和一個 Dimension 物件來指定。

另外，HSceneTemplate.SCENE_PIXEL_LOCATION 和 HSceneTemplate.SCENE_PIXEL_DIMENSION 兩個常數，可以讓應用程式指定 HScene 在 HGraphicsDevice 座標系統內的位置跟大小 (以像素為單位)。而 HSceneTemplate.SCENE_SCREEN_LOCATION 和 HSceneTemplate.SCENE_SCREEN_DIMENSION 兩個常數可以讓應用程式以正規化座標設定 HScene 在螢幕上的大小和位置。

如此，就能在繪圖裝置像素與 MHP 視訊輸出裝置之間做轉換。不過，只有某些轉換是本規格中要求必須支援的。

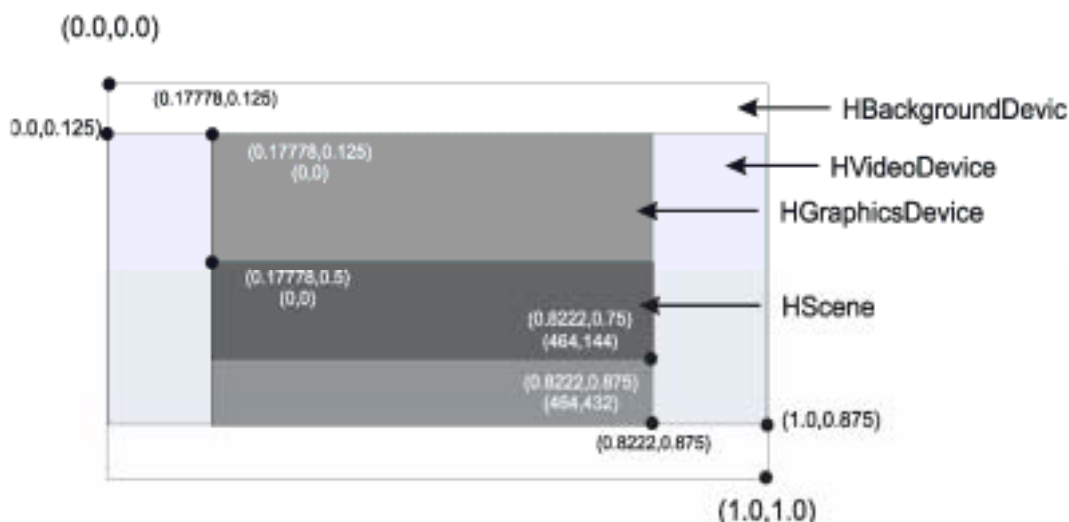


圖 7-5 : HAVI 裝置可能的組態

圖 7-5 顯示 HAVI 裝置可能的組態。其中 HBackgroundDevice 代表整個螢幕，HVideoDevice 則涵蓋 (0.0, 0.125) 到 (1.0, 0.875) 的區域。

當設定視訊輸出位置時，可以將視訊位置貼到鄰近的一條掃描線（上下貼齊），譬如要維持視訊與顯示域（display field）順序，或是貼到鄰近的一點（左右貼齊），譬如要維持顯示色彩結構（display chroma structure）。至於要往哪個方向貼齊，則應選擇與指定座標誤差最小的方向。

在圖 7-5 中 HGraphicsDevice 被設定在 (0.17, 0.125) 到 (0.82, 0.875) 的區域，像素解析度為 464x432。

HScene 的原點位置與尺寸是可以指定的。要設定 HScene 的原點位置，可以用一個 Point(0,216) 物件來設定 HSceneTemplate.SCENE_PIXEL_LOCATION 偏好設定，或以一個 HScreenPoint(0.17,0.5) 物件設定 HSceneTemplate.SCENE_SCREEN_LOCATION 偏好設定。要設定尺寸，則可以用一個 Dimension(464,144) 物件指定 HSceneTemplate.SCENE_PIXEL_DIMENSION 偏好設定，或是用一個 HScreenDimension(0.64,0.25) 指定 HSceneTemplate.SCENE_SCREEN_DIMENSION 偏好設定。

在有些裝置，HGraphicsDevice 上的像素並未對應到顯示裝置上的點。例如當 HGraphicsDevice 是一個模擬的裝置時便是如此。

7.2.1.3 像素寬高比 (Aspect Ratio)

一個以像素為單位的座標系統，並沒有指出其像素寬高比。像素寬高比 ($AR_x^{pixel} / AR_y^{pixel}$) 的定義，就是顯示螢幕的寬高比 ($AR_x^{display} / AR_y^{display}$ ，4:3 或 16:9)，也就是 HGraphicsDevice (w^n, h^n) 涵蓋的區域，其像素解析度為 HGraphicsDevice (w^u, h^u)。請看圖 7-6。

$$AR^{pixel} = \frac{AR_x^{pixel}}{AR_y^{pixel}} = \frac{AR_x^{display} \cdot w^n \cdot h^u}{AR_y^{display} \cdot h^n \cdot w^u}$$

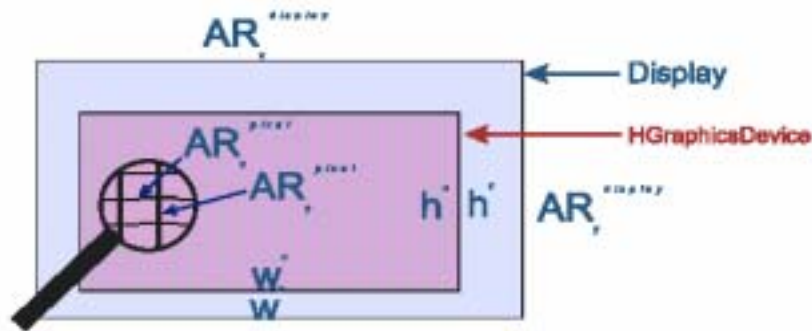


圖 7-6：如何計算像素寬高比

表 7-1 顯示一個典型全螢幕 HGraphicsDevice 的解析度，以及各種顯示器寬高比之下的像素寬高比。

表 7-1：典型常用的解析度與其像素寬高比（參考值）

	4:3 顯示器	16:9 顯示器
全螢幕 HGraphicsDevice 解析度	像素寬高比	像素寬高比
720x576 (註 1)	1150:1053	4600:3159
768x576	1:1	48:36
1024x576	36:48	1:1
註 1：以 ITUR BT.470 [28] 和 ITU-R BT.601 [29] 所得出來的 702 個可顯示像素為基礎。由於還得容忍一些誤差，所以只是近似值。		

7.2.1.4 視訊空間

視訊用的座標空間是視訊輸入訊號所定義的空間，再經過本平台的要求（ETR154 要求）縮放之後的空間。視訊的縮放與裁切可以透過 JMF 控制元件

org.dvb.media.VideoPresentationControl 來達成。而 JMF 控制元件

VideoFormatControl 可以讓應用程式查詢各種視訊訊號解碼與顯示時所做的一些轉換。

7.3 繪圖

7.3.1 MHP 顯示堆疊組成模型

下列各節將說明 MHP 顯示堆疊 (display stack) 的理論模型。不過，在現實世界中，會有許多限制條件，請參閱「7.6 逼近法」。圖 7-7 顯示繪圖、視訊、字幕的處理流程。

圖 7-7 顯示的是從應用程式觀點所看到的一個「圖像內容」處理流程的概念模型。圖像內容的來源有四個：字幕、背景、視訊以及應用程式繪圖物件，這四個來源經過處理之後都投射在單一螢幕上。該圖顯示整個過程牽涉到的一些處理功能、各種內容、以及應用程式在處理過程中的幾個控制點。

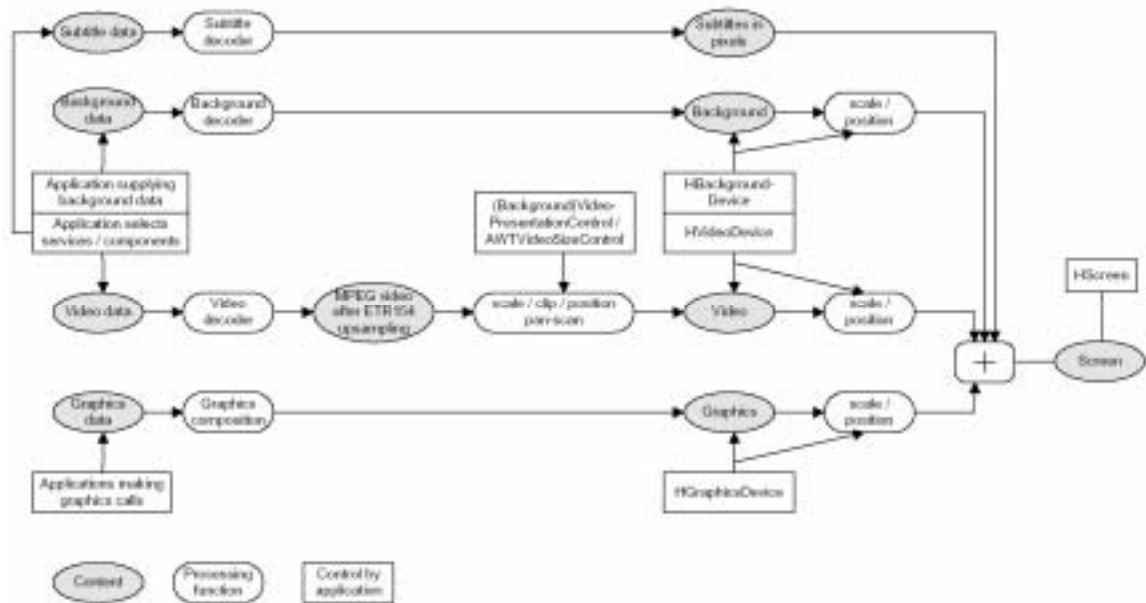


圖 7-7：繪圖、視訊與字幕處理流程

如該圖所示，應用程式可以完全控制內容的來源。此外，應用程式也可以在不同的階段控制內容的裁切、縮放、位置等等。當然，應用程式只能執行底層系統所支援的功能。

字幕的裁切和位置設定則是在視訊處理之後。至於當視訊被縮放之後，字幕如何變化，則隨平台而定。從應用程式作者的角度來看，圖像組合流程有三大要素：

繪圖元素是根據傳統繪圖模型，以 Porter-Duff 規則（參閱 Porter- Duff [D]）來進行組合，預設的規則是 SRC_OVER。

背景與視訊平面則使用 Porter-Duff 中的 SRC_OVER 規則來組合（請注意，視訊平面只用到 alpha 0 和 1）。

這些組合結果進一步再以 SRC_OVER 規則來組合（以繪圖的結果為來源，背景和視訊的結果為目的地）。

圖 7-8 說明這整個流程。

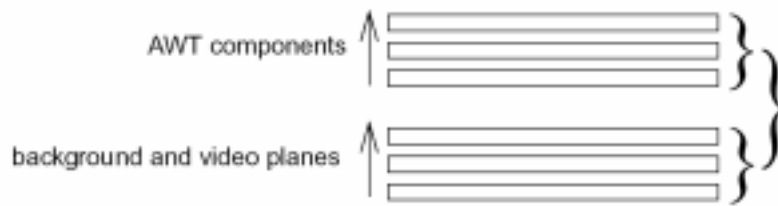


圖 7-8 : AWT / HAVi 平面的組合流程

HAVi 堆疊在背景只有一個全螢幕 HBackgroundDevice。在這前方，還依序排列零個或多個 HVideoDevices。每一個視訊裝置都能夠獨佔整個螢幕，或是只佔其中一部分。任何視訊裝置沒有佔用到的位置則為透明。被佔用的位置（視訊像素）則視為不透明，因此，只要重疊就會互相遮蓋。非作用中（Nonactive）的視訊裝置則會隱藏起來。

只能支援一個全螢幕視訊裝置（以及部分螢幕視訊裝置）的系統，必須將能支援全螢幕的裝置當做第一個（最後方的）視訊裝置。

每一組的組合規則，應該遵循傳統的「畫家（painter）」演算法，也就是說，從最後方開始一層一層依序往前組合。

一個 Xlet AWT Root Component 是透明的（如圖 7-9 所示），因此，在預設情況下，HAVi 視訊與背景裝置組合出來的結果，會變成應用程式圖形的背景。

在應用程式啟動時已經在 HVideoDevice 播放的視訊會繼續播放。

圖 7-9 顯示整個堆疊。

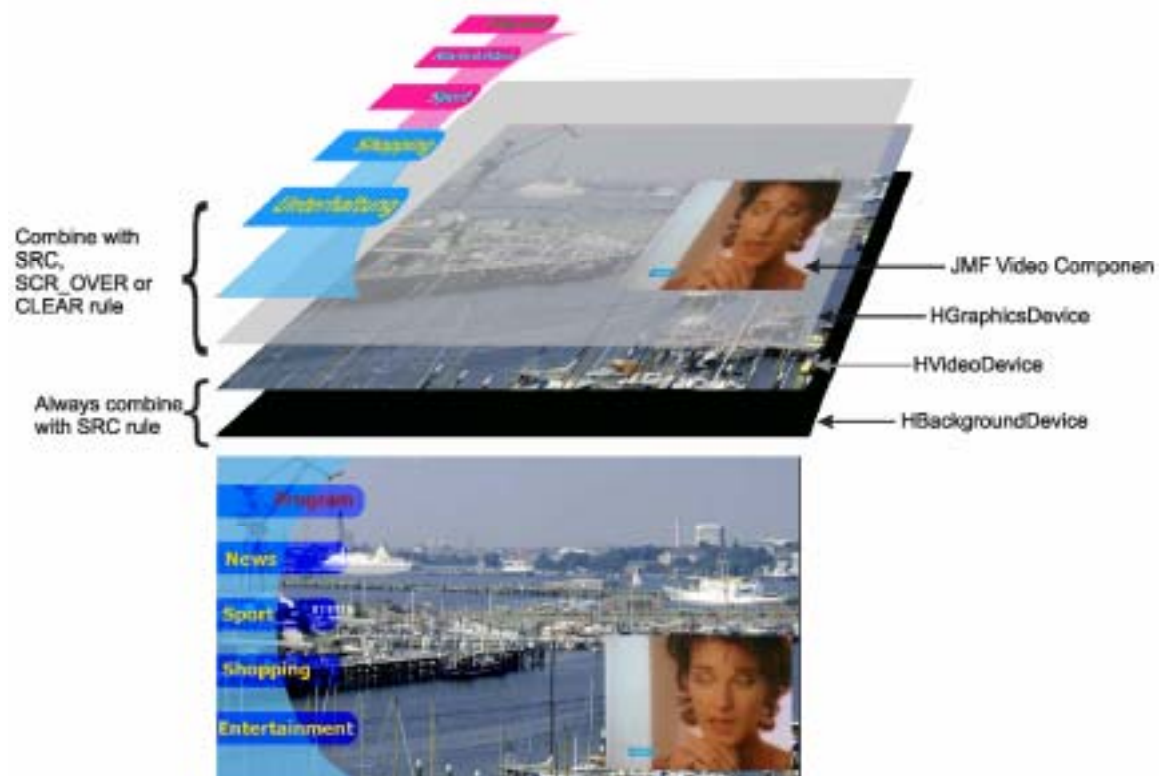


圖 7-9：MHP 顯示堆疊示意圖

7.3.2 AWT 在 MHP 中的參考模型

AWT 所有的輕量級 (lightweight) 元件的繪圖都是透過 `java.awt.Graphics` 類別來達成。當一個元件必須重繪 (repaint) 時，它會發出一個重繪指令，指令會從上到下在元件之間傳遞，直到指令到達重量級 (heavyweight) 元件為止。雖然在 MHP 中並沒有重量級元件，HScene 可以當做類似重量級元件來看。因此，在 MHP 裝置中，重繪指令會一直傳到根容器為止，也就是 HScene。而 HScene 執行重繪的方法，會儘快呼叫該元件的 `update` 方法來執行重繪。在呼叫子物件的繪圖方法之前，原點座標會先轉換到子物件自己的座標系統，而且要繪置的圖形會根據子物件的大小來裁切 (看圖 7-10)。

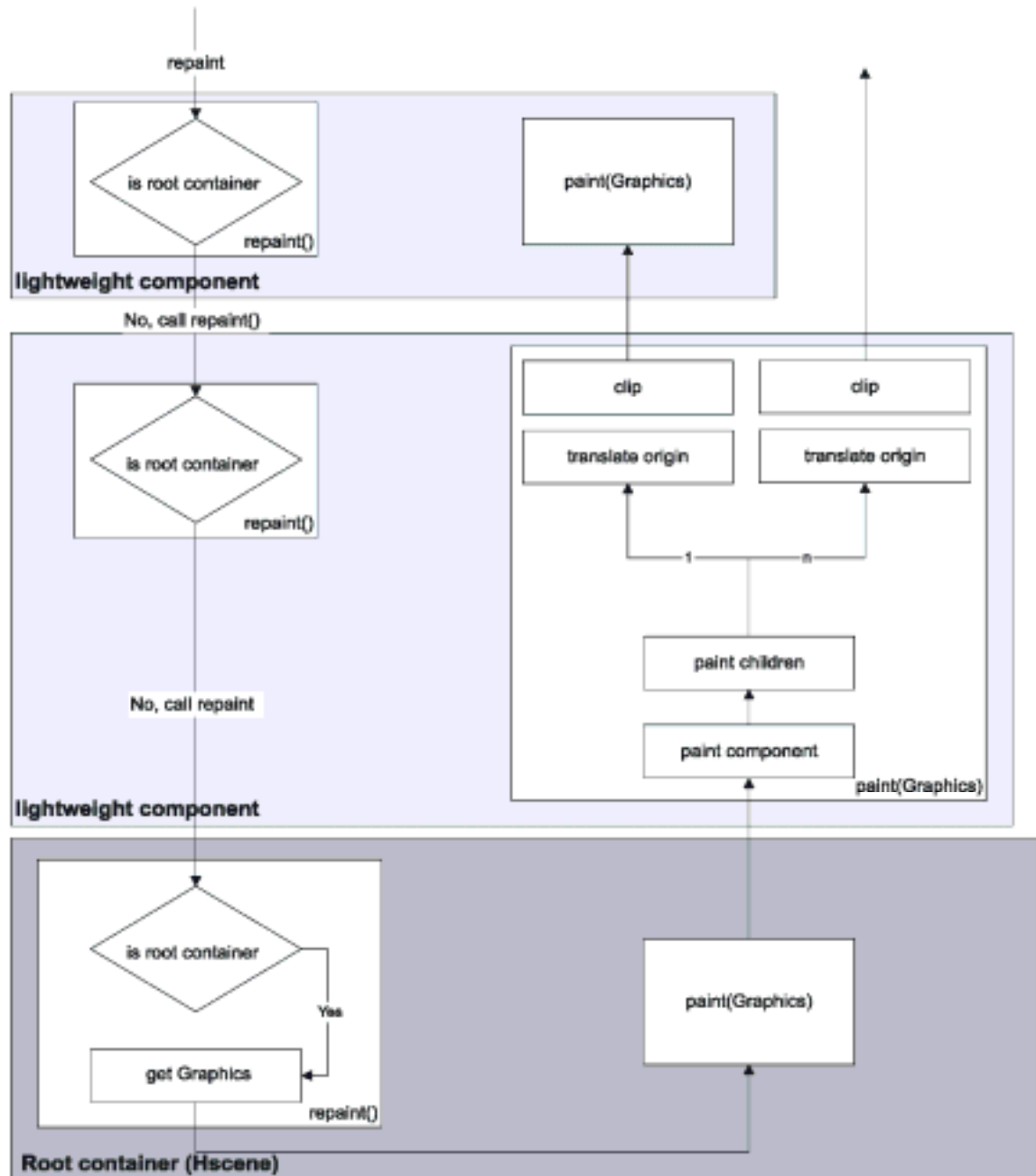


圖 7-10：MHP 的重繪模型

7.3.3 HAVi 裝置和 AWT 元件

一個 DVB-J 應用程式的使用者介面最上層的容器是以 `org.havi.ui.HScene` 類別表示。在 DVB-J 應用程式中，要獲得該類別的實例，可以透過 `org.havi.ui.HSceneFactory` 類別。而 `HSceneFactory` 類別也提供一些方法，讓 DVB-J 應用程式依照個別需求來獲得所需的 `HScene` 物件。`HScene` 在概念上，與 `java.awt.Window` 或 `java.awt.Frame` 是相同的，

只不過，HScene 所模擬的是各種數位電視接收器的繪圖環境。在這個繪圖環境中，對於根容器的大小與位置通常都有許多限制條件。

從 HSceneFactory 取得 HScene 物件的一個方法，就是在 HSceneTemplate 中填入一些條件，然後再利用 `getBestScene()` 方法來取得一個符合這些條件的 HScene。若應用程式需要全螢幕大小的 HScene，則可以使用 `getFullScreenScene()` 方法，並指定用來顯示該全螢幕畫面的繪圖裝置。

一個 MHP 終端機可允許只支援「非」重疊式 HScene。不過此時 MHP 終端機應該遵循下列原則：

當 HSceneFactory 被要求產生重疊式 HScene 或是改變 HScene 大小時：如果對方指定了「REQUIRED」參數，則應傳回無效參照 (null reference)；如果對方指定了「PREFERRED」參數，則應儘可能傳回有效值。

當 awt 改變大小而發生重疊時：大小不變，呼叫直接失敗，無需額外回應。

MHP 規格採用 HAVi 規格中所使用的一個視訊輸出裝置通用模型。最終的視訊輸出訊號是以 `org.havi.ui.HScreen` 類別來表示，此一訊號是合併各種視訊元素的結果。

繪圖解碼器的輸出

視訊解碼器的輸出

特殊解碼器的輸出，譬如背景

每一個解碼器分別用一個從 HScreenDevice 繼承的子類別來表示：HGraphicsDevice、HVideoDevice、HBackgroundDevice。每一個裝置又可以有各式各樣的組態，分別用一個從 HScreenConfiguration 繼承的子類別來代表：HGraphicsConfiguration、HVideoConfiguration、HBackgroundConfiguration。當裝置支援多重組態時，應用程式可以利用範本 (template) 來選擇不同的組態。這些範本是從 HScreenConfigTemplate 繼承的子類別。

這些子類別，除了提供一般通用功能之外，還提供一些裝置專屬的功能。

HGraphicsConfiguration 支援載入影像以及列出該組態所支援的字型。同時也支援各種座標系統轉換。HVideoDevice 可提供目前所解碼的視訊來源 (一個定址器 Locator)，也提供用來解碼的解碼器物件 (`javax.media.Player`)。而 HBackgroundClass 則透過 HStillImageBackgroundConfiguration 類別提供 MPEG I-frame 支援。

而 `HScreen.getCoherentScreenConfigurations (HScreenConfigTemplate[] configs)` 方法則可以讓應用程式透過一組共用的視訊、繪圖和背景條件，來得到一致 (coherent) 的組態。在 HGraphicsConfigTemplate 中，有一個 VIDEO_MIXING 常數定義可讓應用程式取得一個繪圖物件重疊在視訊畫面上方的組態，不過像素未必對齊。在 HScreenConfigTemplate 中有些常數定義可讓應用程式取得下列幾種組態：

VIDEO_GRAPHICS_PIXEL_ALIGNED - 視訊與繪圖的像素大小一致並對齊

ZERO_VIDEO_IMPACT - 繪圖組態不得改變現有視訊組態

ZERO_GRAPHICS_IMPACT - 視訊組態不得改變現有繪圖組態

7.3.3.1 視訊和繪圖像素對齊

圖 7-11 顯示 HGraphicsDevice 和 HVideoDevice 在像素對齊

(VIDEO_GRAPHICS_PIXEL_ALIGNED) 的組態。請注意，此處 HVideoDevice 是經過「解碼器格式轉換」(Decoder Format Conversion) 處理之後的像素，與 HGraphicsDevice 的像素對齊。因此，HGraphicsDevice 和 HVideoDevice 兩個裝置的像素要對齊有下面的條件限制：

兩個裝置的像素寬高比必須相同。

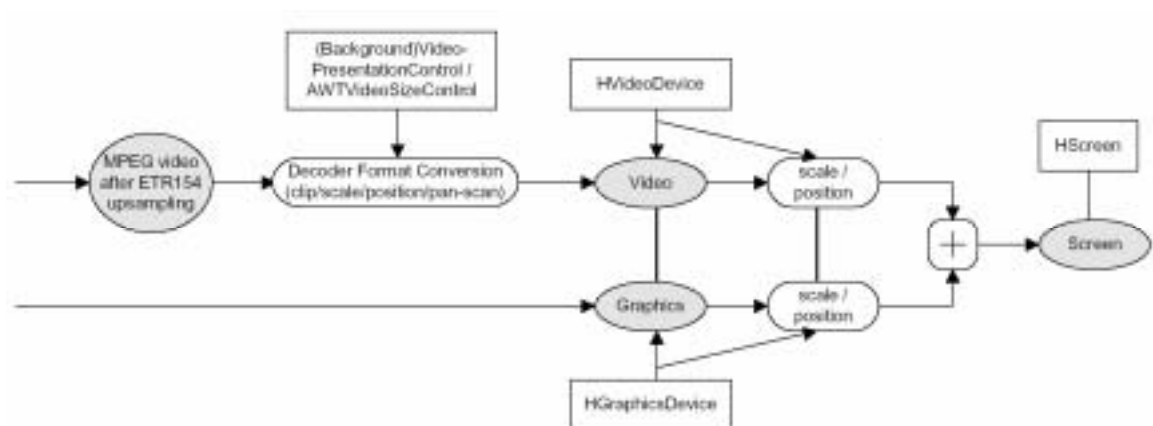


圖 7-11：視訊像素與繪圖像素間的交互影響

7.3.3.2 繪圖像素不受影響

當應用程式不希望 HGraphicsDevice 或 HVideoDevice 的像素受到影響時，可以分別用 ZERO_GRAPHICS_IMPACT 和 ZERO_VIDEO_IMPACT 兩個常數來避免。一般來說，當改變 HGraphicsDevice 的組態時，是有可能會自動改變 HVideoDevice 的組態（當然，應用程式必須有這項變更的權限），這是因為有些系統可能無法同時處理兩種不同的組態。因此，應用程式若不希望影響到另一個裝置的組態，就必須指定 ZERO_GRAPHICS_IMPACT 或 ZERO_VIDEO_IMPACT 組態範本。

7.3.4 組合

7.3.4.1 AWT 繪圖規則

一般來說，繪圖時必須遵循 AWT 的繪圖規則。也就是說，當最上層的容器（此處為 HScene）重繪時，就會促使其中的元件以遞歸的方式（recursively）逐一重繪。

正常的行為表現應該是每個元件都直接繪到根元件 (root component)。即使過程中使用了暫時儲存 (temporary storage) 或是雙重緩衝 (double buffering) 技巧，也不能改變 AWT 最終的組合結果，或是改變後續與視訊組合之後的結果。

圖 7-12 顯示此一流程。圖中箭頭上的數字代表繪圖時的時間順序 (請留意根容器是第一個重繪的)。當一個容器內有許多元件時，這些元件將會以 $N, N-1, \dots, 0$ 的順序重繪，其中 N 代表該元件被加入此容器的先後順序。(請參閱 JAE 1.1.8 API [31] 中有關 `java.awt.Container` 的說明)。

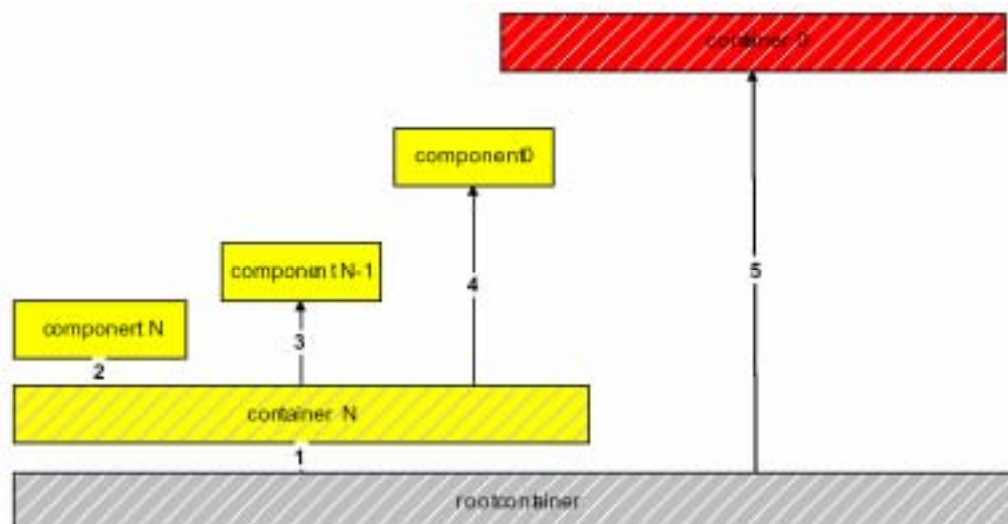


圖 7-12：繪圖的時間順序

7.3.5 組合規則

7.3.5.1 元件通則

要產生混色的效果，只要對 `DVBGraphics` 設定好適當的 `DVBAlphaComposite` 規則，再配合半透明色就行。而複雜的畫面，則可以透過許多的元件來組合而成。圖 7-13 的背景圖片 (港口圖片) 是視訊平面。半透明的灰色四方形，則是利用 `SRC` 規則畫在一個透明的螢幕下 (off-screen) 緩衝區內。然後再利用八種最常用的 Porter-Duff 運算規則，將紅色的圓形畫到這個螢幕下緩衝區內。當 AWT 在這個螢幕下緩衝區內繪圖完成之後，整個緩衝區的內容再以 `SRC_OVER` 的規則與視訊訊號組合成螢幕上最終的畫面。



圖 7-13 : Porter-Duff 的八種組合規則示範

7.3.6 延伸 AWT 繪圖能力

在繪圖方面，MHP 規格主要採用 AWT 的繪圖能力（參考 JAE 1.1.8 API [31] 規格）。不過，還是有一些延伸功能。

為了讓 MHP 擁有繪圖組合（compositing）能力，MHP 提供了 `org.dvb.ui` 封裝來提升 JAE 1.1.8 API [31] 的繪圖能力。

此封裝包含了 `DVBGraphics`、`DVBAlphaComposite`、`DVBBufferedImage`、`DVBColor` 等類別，以及一個 `Exception` 類別。

7.3.6.1 MHP 中的繪圖物件

`org.dvb.ui.DVBGraphics` 類別延伸了 `java.awt.Graphics` 類別的功能，增加 alpha 組合能力。

在 MHP 中，每個由平台所產生的繪圖物件，都必須是 `org.dvb.ui.DVBGraphics` 類別的實例。

`DVBGraphics` 和 `DVBAlphaComposite` 原則上支援八種不同的 Porter-Duff 組合規則，但是並非所有的 `DVBGraphics` 物件都必須支援全部的組合規則。

不同的 `DVBGraphics` 物件可支援不同的組合規則。譬如，利用 `DVBBufferedImage` 產生的 `DVBGraphics` 物件，若指定 `DVBBufferedImage.TYPE_ADVANCED` 為影像類型，則所有八種 Porter-Duff 規則都支援，但若指定 `TYPE_BASIC` 影像類型，則只支援 `SRC`、`CLEAR`、`SRC_OVER` 規則。應用程式可以使用 `DVBGraphics.getAvailableCompositingRules()` 來查詢可用的組合規則。若指定了一個不支援的組合規則，則會產生一個

`org.dvb.ui.UnsupportedDrawingOperationException` 例外狀況。只要是一個影像的 `getGraphics` 方法所傳回的物件有效（non-null），該影像就必須支援 alpha 組合能力。

所有繪圖物件預設的組合規則則是 `SRC_OVER`。

7.3.6.2 經過緩衝的影像

在 `org.dvb.ui` 封裝中的 `DVBBufferedImage` 類別，新增了一個可存取的透明影像，可做為螢幕下的緩衝區。`DVBBufferedImage` 支援兩種隨平台而定的樣本模型 (sample model)。當使用 `TYPE_BASE` 樣本模型時，有可能會使用逼近法來替代 (使用 `SRC` 而不是 `SRC_OVER` 規則，如圖 7-21，或是使用 `alpha` 的逼近值，但若是使用 `TYPE_ADVANCED` 樣本模型，則會使用程式所指定的組合規則。

`TYPE_ADVANCED` 永遠都是使用直接色彩模型 (direct colour model)，但是 `TYPE_BASE` 則可以使用以 `CLUT` 為基礎的色彩模型。

7.3.6.3 DVBColor

註：此類別的設計，是要模仿 `JDK 1.2` 在色彩處理上的功能。

除非另有說明，否則平台內部實做 (譬如 `java.awt`) 應該使用 `org.dvb.ui.DVBColor` 類別，而不是使用 `java.awt.Color`，除非技術上不可行，譬如，並非 `java.awt.Color` 的建構函式 (constructor)。至於類別宣告 (class signature) 則不應該改變。比如說，當一個方法必須傳回 `java.awt.Color` 時，就必須傳回一個 `org.dvb.ui.DVBColor` 實例。

7.3.6.3.1 修改後的色彩資料格式

一個用來表示 RGB 色彩的整數，其最高位元組 (most significant byte) 用來儲存 `alpha` 值，如圖 7-14 所示。這個資料類型在本 API 中的各種建構函式或方法都會用到，以 `TYPE_INT_ARGB` 來表示。

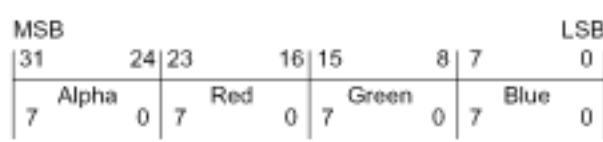


圖 7-14 MHP 色彩資料格式 (`TYPE_INT_ARGB`)

7.3.7 14:9 寬高比支援

除了 4:3 和 16:9 等一般顯示器寬高比之外，MHP 還支援一種 14:9 的寬高比。當在這種比例模式時，所有的文字的寬高比例都會稍微有點扭曲。不過，在這樣的比例下，文字的折行位置比較能夠預期，廣播業者就不必為每一種類型的顯示器而重新製作內容。

所有 `HGraphicsDevices` 都必須能夠模擬 `HEmulatedGraphicsDevices`，並在其裝置上模擬出 14:9 的寬高比。

註：應用程式還是可以將 `HEmulatedGraphicsDevice` 的組態指定到 `HGraphicsConfiguration` 當中，並且不需透過模擬就能直接支援此組態。

若一個 HEmulatedGraphicsDevice 處於 14:9 的組態，且該裝置又未保留給任何 MHP 應用程式使用時，當 MHP 終端機的真實顯示寬高比在 4:3 和 16:9 之間切換時，應自動維持 14:9 的比例。

註：應用程式可能會發現 HGraphicsConfiguration 在一開始執行時，可能會不太穩定，這是因為可能已經有一個執行中的 MHP 應用程式已經佔用了該 HGraphicsDevice 並設好了 HGraphicsConfiguration 組態。

7.4 視訊

7.4.1 元件式播放器與背景式播放器

MHP 所接收的視訊是一連串 MPEG 壓縮的畫面 (frame)，畫面中包含大量以縱橫排列的影片元素 (picture elements，簡稱 pel)。MHP 將 MPEG 串流解壓縮成顯示用的串流，然後顯示在視訊平面，或者是繪圖平面上的一個元件內。

由於有這兩種播放視訊的方式，因此就產生兩種 JMF 播放器：一是背景式播放器，一是元件式播放器。

元件式播放器是在一個 AWT 元件中播放視訊，元件中播放的視訊會隨著元件位置與大小的改變而改變，視訊畫面也會佔滿整個元件的大小。並非所有的應用組態 (profile) 都必須支援元件式播放器。

不過，對於廣播串流格式來說，背景式播放器卻是必須支援的（請參閱「3.2 廣播串流格式」一節）。

背景式 JMF 播放器會在視訊平面上播放視訊，因此是完全獨立於 AWT 元件階層架構之外的。

當一個元件式播放器停止播放時，將顯示最後一個播放的畫面，直到播放器底層的解碼器資源再度被重新使用為止。至於當底層的視訊解碼器被重新使用時的行為表現，則須視實做而定。

當背景式播放器處於實現狀態 (realized state)，其所使用的視訊平面將變成透明的，以便顯現其背後的内容，譬如背景平面。請參閱「圖 7-1：三種顯示平面示意圖」。

7.4.2 MPEG 解碼與顯示流程模型

圖 7-15 顯示一個 ETSI TR 101 154 [9] 相容的 SD 數位接收器底層的格式轉換控制流程。此一模型中，視訊解碼器從 MPEG 資料產生出「全螢幕」視訊，也就是說，解碼器會處理所有 MPEG 廣播中的次級取樣 (sub-sampling)。後續的「解碼器格式轉換」將會根據顯示裝置來調整這些視訊訊號，並處理下列資訊：

廣播結構資料 (meta data) (譬如寬高比、pan/scan、作用中格式描述)

顯示相關資訊 (譬如 4:3 或 16:9、解析度)

使用者偏好（譬如顯示有上下黑邊的寬螢幕畫面）

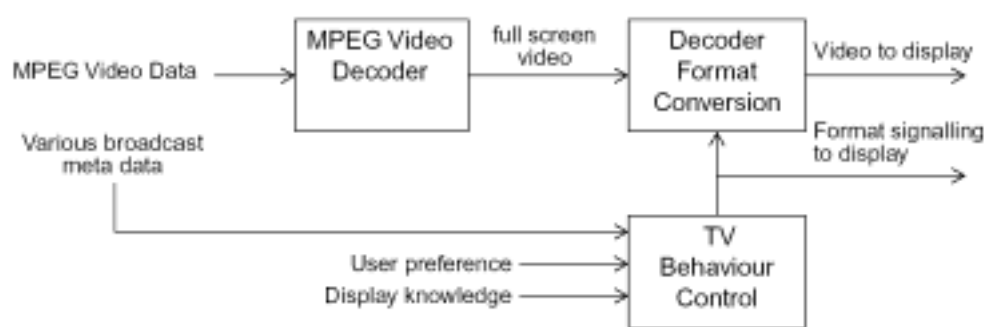


圖 7-15：電視模式中的格式控制

在適當的情況下，傳送到顯示器的視訊通常會伴隨著正確的 WSS 或 SCART 訊號，來表示視訊的格式。

請注意，在 MHP 做過解碼器格式轉換之後，顯示裝置有可能還會自己進行解碼器格式轉換。但這已在 MHP 裝置的控制之外。

本 MHP 規格版本的 JMF 播放器是屬於「DVB ETR 154 Standard Definition」播放器，因此播放器會假設它的邏輯視訊來源是 MPEG 視訊解碼器所產生的全螢幕輸出，如圖 7-16 所示。此外，解碼器格式轉換流程有兩種控制方式：

傳統電視格式控制行為（如圖 7-15）

應用程式格式控制行為

這兩種方式可由應用程式自己選擇。應用程式存在之前與之後的控制方式，則是傳統電視方式。而在應用程式執行期間，JMF 播放器預設的解碼器格式轉換行為還是傳統電視方式。

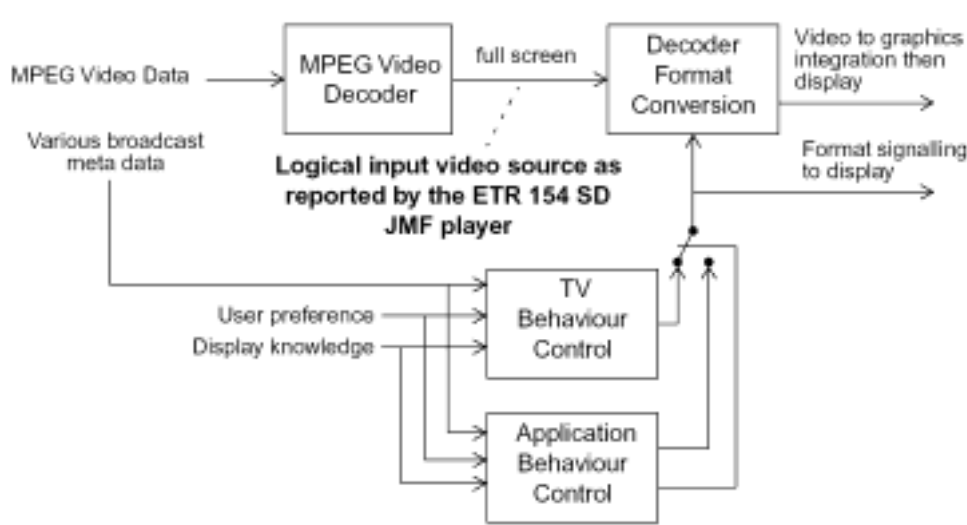


圖 7-16：JMF 播放器存在時的格式控制

解碼器格式轉換流程包含三個步驟，其輸入為全螢幕視訊（也可能是經過 MPEG 視訊解碼器向上取樣才變成全螢幕）。如圖 7-17 所示，這三個步驟是裁切 (clipping)、縮放 (scaling) 與定位 (positioning)。

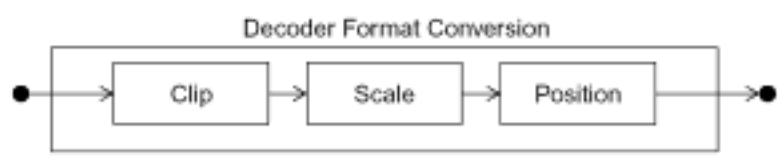


圖 7-17：解碼器格式轉換參考模型

應用程式可以透過 `org.dvb.media.VideoPresentationControl` 來針對解碼器格式轉換中的每個步驟查詢實做所支援的能力。譬如，應用程式可以查詢實做所支援的縮放比例。

應用程式也可以透過 `org.dvb.media.VideoTransformation` 物件個別設定解碼器格式轉換的每個步驟，此物件同時包含了裁切、縮放、定位等參數。應用程式可以獲得一些標準解碼器格式轉換的視訊轉換設定，譬如在一個 4:3 的顯示器上呈現 16:9 畫面（有上下黑邊的 letterbox）。應用程式也可以利用這些轉換物件直接設定解碼器格式轉換，或是在設定之前先變更一些參數值。本 API 也支援查詢目前的視訊轉換設定。

7.4.3 座標空間

解碼器視訊轉換功能區塊的輸入訊號永遠都是全螢幕畫面。如果必要，MPEG 視訊解碼器功能區塊會依照 ETSI TR 101 154 [9] 的要求進行向上取樣，以獲得全螢幕視訊。

應用程式是以全螢幕的像素座標空間來指定裁切區域。對於 50Hz SD 視訊來說，這就是 720x576 的空間。

對於背景式 JMF 播放器來說，視訊位置的指定，是以正規化的座標表示。而對於元件式的播放器來說，視訊元件的位置，是以裝置的像素座標空間來表示，（也就是說，一個視訊元件與其他的 AWT 元件是一樣的）。

7.4.4 視訊元件

一個 AWT 階層架構中的元件，可以用來播放一個局部放大的視訊畫面。這樣的呈現方式會同時受到 AWT 和 JMF 的影響。

視訊元件與其他元件是相同的。不過請注意，視訊中的像素都是不透明的。與來源對應的功能是由一個視訊呈現控制元件所提供，該控制元件讓局部視訊畫面顯示在 AWT 階層架構中。

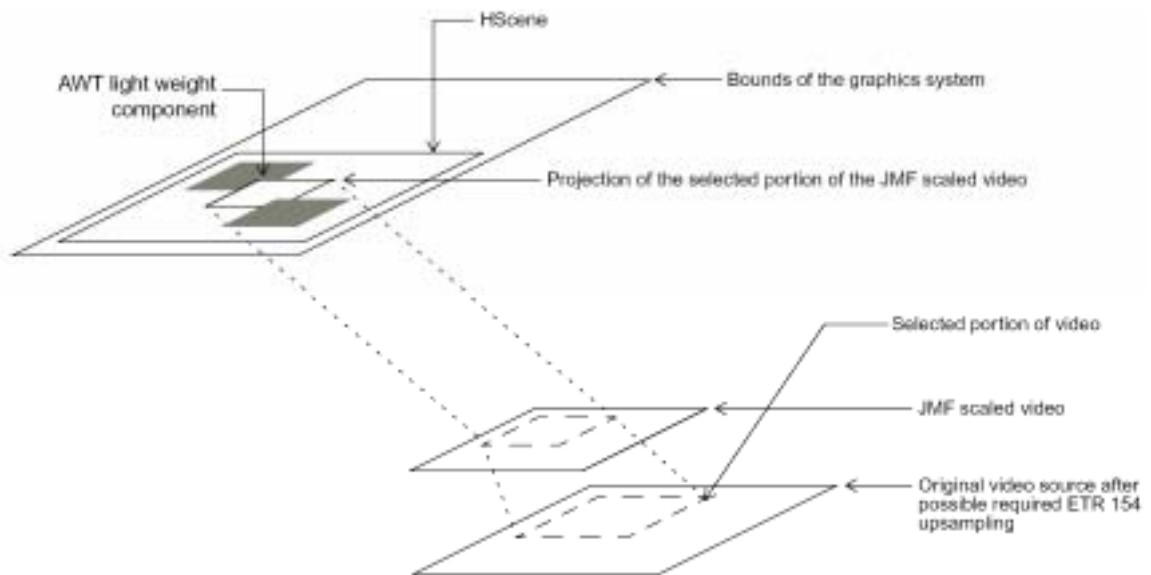


圖 7-18：在 AWT 元件架構中導入視訊

7.5 字幕

7.5.1 語言和顯示設定

底下的參考模型顯示字幕的顯示方式。字幕語言的選擇需視使用者偏好和音訊語言而定，同時也可以由應用程式強制指定。終端使用者可以開啟或關閉字幕，預設值則決定於使用者偏好。不過，這些還是可以由應用程式強制設定，就算使用者已經開啟字幕，應用程式還是可以將它關閉。

下圖顯示整個決定的流程：

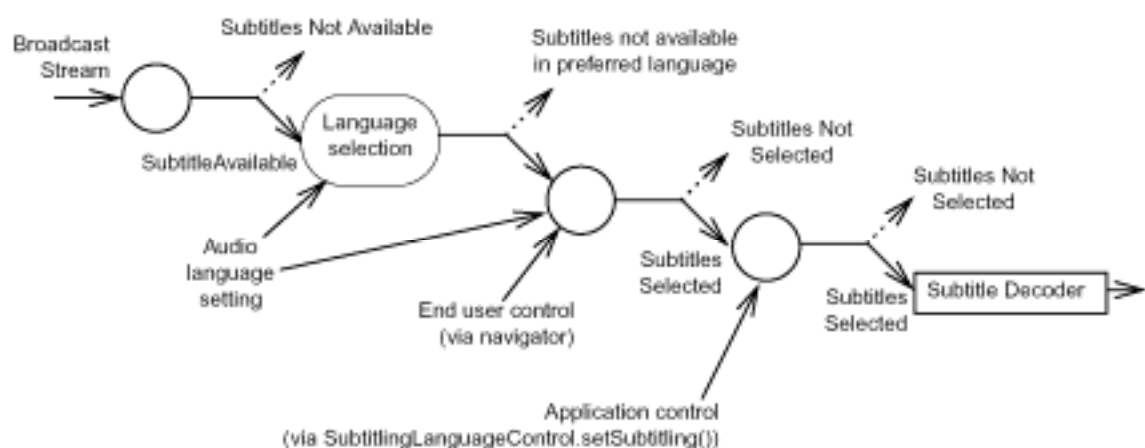


圖 7-19：如何決定字幕語言以及顯示設定

DVB-J API 當中有一個控制元件（請參閱 SubtitlingEventControl 一節）可以在字幕狀態（有

無字幕和顯示設定) 改變時收到通知。這對應圖中左側輸入狀態和右側輸出狀態。語言的選擇以及最後決定的偏好語言, 都是由實做的演算法來決定。譬如, 實做平台可以讓音訊與字幕的語言偏好設定是分開的。

應用程式可以利用 `org.davic.media.SubtitlingLanguageControl` 來查詢目前設定的字幕語言、強迫蓋過預設語言、關閉字幕、或是讓終端使用者自行控制。這對應圖中語言選擇階段以及應用程式控制的切換部分。

7.5.2 與繪圖平面的關係

在理想狀態下, 字幕應該顯示在視訊平面上方以及繪圖平面下方。不過, 符合本規格的 MHP 終端機只需在應用程式繪圖物件 (也就是 DVB-J 應用程式的 HScene 物件) 沒有覆蓋的區域支援字幕功能。當字幕與應用程式繪圖物件重疊時的處理方式則沒有規定。因此, 當應用程式在螢幕上顯示繪圖物件時, 應用程式應該先關閉字幕, 要不然, 廣播業者應該協調字幕與應用程式繪圖物件所使用的區域, 讓它們不會重疊。

字幕平面是全螢幕的, 因此, 字幕可以顯示在螢幕的任何角落。字幕文字的顯示位置, 是字幕內容串流的一部份, 因此完全控制在廣播業者。

7.5.3 座標空間

字幕資料會解碼到一個與 HVideoDevice 座標系統、位置皆相同的平面上。

7.6 逼近法

7.6.1 組合時的逼近法

MHP 規格在組合時所使用的是 Porter-Duff 規則。實做時的組合過程可以採用下面所述的逼近法。

7.6.1.1 各種模式的做法

典型的做法, 就是透過硬體使用 SRC_OVER 規則來混合繪圖平面和視訊平面 (請參閱圖 7-20)。這樣對於顯示模型的一致性會有一些實際上的限制。

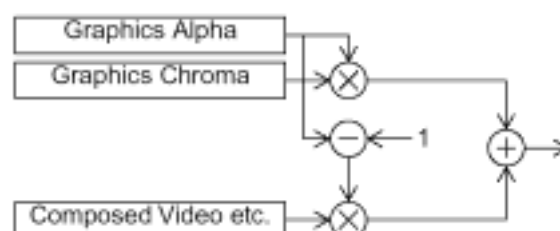


圖 7-20：當前技術的典型做法

SRC 規則只要求新的像素色彩與 alpha 值必須取代原有的像素特性。

HVideoComponents 的 alpha 值為 1，因此，當在螢幕上放置一個視訊元件時，不管是用 SRC 或 SRC_OVER 規則，所呈現出來的效果都是視訊內容完全取代原先畫面上的內容。

在此，我們假設 MPEG I-frame 和視訊涓流是當成視訊內容而不是繪圖物件（即便是實做上可能會部分利用到繪圖系統）。

7.6.1.1.1 繪圖物件直接覆蓋視訊

當繪圖物件直接覆蓋在視訊上方時：

會呈現 SRC_OVER 模式的效果，因為，繪圖物件的 alpha 值會控制繪圖物件與視訊混合的效果。

7.6.1.1.2 繪圖物件覆蓋繪圖物件

7.6.1.1.2.1 SRC

如果繪圖物件是以 SRC 模式畫在其他繪圖物件上方，得到的結果將正如預期。

7.6.1.1.2.2 SRC_OVER

如果繪圖物件是以 SRC_OVER 模式畫在其他繪圖物件上方，當 $0 < \alpha < 1$ 時，結果需視實做而定。當 $\alpha = 0$ 時（也就是完全透明），彩色繪圖並不會對既有的像素有任何影響。圖 7-21 顯示以 SRC_OVER 模式將繪圖物件畫在其他繪圖物件上方的一些做法。當樣本模型是 TYPE_BASE 時，有一種 JAE 1.1.8 API [31] 中允許的做法，就是 `java.awt.Graphics` 中那些不需要「bgcolor」參數的 `drawImage` 方法所採用的做法。確切來說，也就是只有完全不透明的像素才會畫上去。要畫上去的像素只要有一點點透明度，就不會對原來的像素有任何影響。

[a] 在綠色和紅色區域混合的地方，以邏輯運算的結果顯示中間色。

此一方法必須用於繪圖物件對繪圖物件的混合流程，同時也需要色彩與 alpha 兩個通道上都有很大的色域（gamut）。在早期的做法中，這可能不適用。

底下示範幾種簡化的混合方法。但這些是比較沒那麼好的逼近法：

[b] 只有在繪圖物件直接畫在視訊元件上方時，才去處理繪圖物件的 alpha 值。當繪圖物件畫在其他繪圖物件上方時，alpha 值就當成 1 來處理。

[c] 當來源繪圖物件畫在視訊元件上方時，就要處理其 alpha 值，即使中間有一個半透明的繪圖物件也是如此。當來源物件是畫在不透明的繪圖物件上時，就使用繪圖物件到繪圖物件的混合作法。

- [d] 不管 MPEG 視訊影像上方已經存在的繪圖內容為何，都將來源繪圖物件的 alpha 值設為硬體混合 (hardware mixing) 時的 alpha 值。
- [e] 在 alpha 值已經是小於 1 的區域，就使用來源繪圖物件的 alpha 值，做為硬體混合時的 alpha 值。當 alpha 值等於 1 (也就是正畫在不透明的繪圖物件上)，就繼續使用 1 做為 alpha 值。

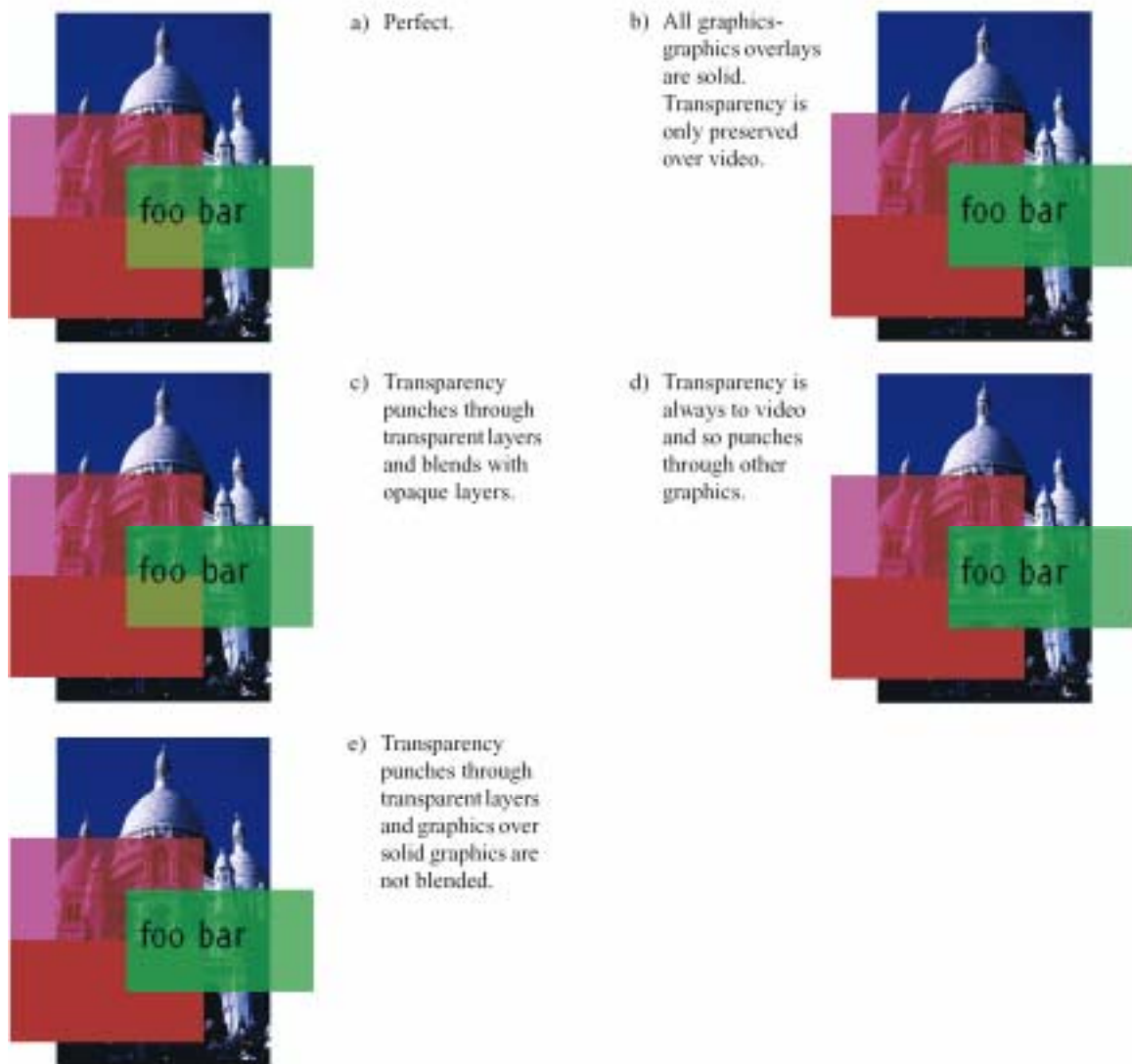


圖 7-21：混合的做法

7.6.1.1.2.3 CLEAR

如果繪圖物件是使用 CLEAR 模式畫在其他繪圖物件上，結果將正如預期。此運算模式的結果，與當來源繪圖物件的 $\alpha=0$ 時採用 SRC 規則一樣。

7.6.1.2 Alpha 的逼近法

在實做上，Alpha 呈現的精準度需視不同的 DVBGraphics 物件而定。想知道某個顏色最後真正呈現的顏色，可以透過 `org.dvb.ui.DVBGraphics.getBestColorMatch()` 方法來查詢。

7.6.1.3 色彩的逼近法

邏輯上來說，應該是使用「真實色彩」(true color) 模型。但是，也可使用逼近色彩。

8 系統整合層面

8.1 名稱空間對應 (DVB 定址器)

DVB-SI 和 object carousel 中的檔案，需使用延伸格式的 DAVIC DVB URL (DAVIC 1.4.1p9 [3]) 來定址。此延伸格式的 DAVIC 定址器是與原始版本的 DAVIC 定址器以及 UK DTG 延伸版本 (UK MHEG Profile [B]) 是向下相容的。主要延伸部分，是支援使用多個元件標記來指定一個服務中的某些元件，並提供一種方式來參照 object carousel 中的服務。

此延伸格式定址器非正式表示方法如下：

dvb://<original_network_id>[.<transport_stream_id>][.<service_id>[.<component_tag>{<component_tag>}];<event_id>]]{<path_segments>}

或

dvb://'<textual_service_identifier>'[.<component_tag>{<component_tag>}];<event_id>]]{<path_segments>}

而以 BNF 表示的正式規格，則如下所示：

表 8-1：DVB URL 語法

```
dvb_url           = dvb_scheme ":" dvb_hier_part
dvb_scheme        = "dvb"
dvb_hier_part     = dvb_net_path | dvb_abs_path
dvb_net_path     = "://" dvb_entity [ dvb_abs_path ]
dvb_entity        = dvb_transport_stream | dvb_service | dvb_service_component
dvb_transport_stream = original_network_id "." transport_stream_id
dvb_service       = dvb_service_without_event [ dvb_event_constraint ]
dvb_service_component = dvb_service_without_event "." component_tag_set [ dvb_event_constraint ]
dvb_service_without_event = original_network_id "." [ transport_stream_id ] "." service_id |
    "" textual_service_identifier ""
component_tag_set  = component_tag *( " " component_tag )
dvb_event_constraint = ";" event_id
original_network_id = hex_string
transport_stream_id = hex_string
service_id         = hex_string
component_tag      = hex_string
event_id           = hex_string
hex_string         = 1*hex
hex                = digit | "A" | "B" | "C" | "D" | "E" | "F" | "a" | "b" | "c" |
    "d" | "e" | "f"
digit              = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
dvb_abs_path      = "/" path_segments

(path_segments as defined in IETF RFC 2396 [41])
```

請注意，此處的語法完全符合 IETF RFC 2396 [41] 當中所訂定的 URI 通用語法，並使用其 registry-based naming authority 版本。此外，所有 IETF RFC 2396 [41] 中的通用定義對於 DVB URL 也適用（譬如，檔案名稱內特殊字元的表示方式）。

IETF RFC 2396 [41] 定義了一些在路徑 (path_segments) 上包含參數的方式（以英文分號「;」開頭）。本規格目前沒有用到這類參數。因此，要符合本規格，實做時最好忽略這類參數，以確保能與本規格未來版本相容。

8.1.1 dvb_entity = dvb_service

當一個 URL 中包含一個路徑，而路徑的 dvb_entity 部分指向一個 DVB 服務時，該路徑便指向服務中的 object carousel 內的物件。如果 URL 中沒有 dvb_service_component，則該服務就應該只有一個 Object Carousel。

而 original_network_id、transport_stream_id (如果有的話) 和 service_id 則應該對應到 SDT 中的相關欄位。

8.1.2 dvb_entity = dvb_service_component

當一個 URL 中包含一個路徑，而路徑的 dvb_entity 部分指向一個 DVB 服務中的一個元件，而且該元件包含了一個 object carousel 串流，該路徑便指向 object carousel 中的一個物件，同時這個 object carousel 的「root」（也就是 DSI 訊息）也經由所指到的元件傳送。此時，應該只有一個元件標記。

前述 URL 中的路徑，如果其 dvb_entity 部分不是上述兩種情形，則其意義不在本規格的定義中。

而 original_network_id、transport_stream_id (如果有的話) 和 service_id 則應該對應到 SDT 中的相關欄位。

8.1.3 dvb_hier_part = dvb_abs_path

如果 URL 中沒有 dvb_net_path 部分，而且只有 dvb_abs_path，該 URL 便指向當前服務的預設 object carousel 當中的檔案。所謂「當前」服務，要看使用的環境而定。

8.1.4 dvb_abs_path

dvb_abs_path 部分，有下列限制：

整個路徑名稱的長度，包括路徑分隔字元與檔案名稱，必須小於等於 254 個位元組 (byte)。

檔案和路徑名稱當中不得使用 null 字元 (0xC080) 和位元組「零」(byte zero)。

檔案名稱的編碼方式為 UTF-8 (請參閱「3.1.5」一節)。

目錄的分隔字元 (也就是 Java 的路徑分隔字元屬性) 為斜線 (slash) 字元 (0x2F)。

絕對路徑必須以一個斜線字元開頭 (如前面的 BNF 所示)。

8.1.5 dvb_entity = dvb_transport_stream

original_network_id 和 transport_stream_id 兩個識別碼數字應該對應到 NIT 中的欄位。

8.2 保留名稱

以「dvb.」字串開頭的檔案名稱，必須保留給目前與將來版規格「眾所皆知」(well known) 的檔案使用。

開發人員不應該使用這樣的名稱，以免與標準所定義的檔案名稱衝突。

8.3 XML 標記法

在 MHP 中以 XML 來進行檔案的編碼與處理時，應該遵守下列規則。

XML 格式檔案規則：

這些檔案必須是格式完整 (well-formed) 的 XML 文件 (不過，對此規格版本中所指定的 DTD 來說並不一定是有效的 (valid))。此處所謂「格式完整」(well-formed) 和「有效」(valid)，是根據 XML 1.0 規格的定義 (請參閱 XML 1.0 [65])。

註：由於 XML 有可能只對某一個 DTD 才有效，因此才有此說明。對本規格將來版本所定的 DTD 有效的 XML 文件，就會對目前版本所定的 DTD 無效，不過，這裡所做的規則，就是希望讓符合當前標準的 MHP 終端機也能處理將來的文件。

XML 檔案在開頭的 prologue 部分，可以包含 XMLDecl 項目 (「<?xml ?>」標記)。

所有 XML 檔案都應該以 XML 預設的 UTF-8 編碼。

就算檔案開頭有 XMLDecl 項目，也不能以「encoding」屬性來指定 UTF-8 以外的編碼方式。

如果檔案開頭有 XMLDecl 項目，則應該在其中標明「XML version 1.0」。

XML 檔案應該包含一個文件類型宣告 (「<!DOCTYPE >」標記)，其中，Name 必須與 root element 的名稱相同。

文件類型宣告應該包含一個 ExternalID 項目，包含「PUBLIC」識別記號，並且包含一個 PublicLiteral 和一個 SystemLiteral。本規格中用到的文件類型都有指定的 PublicLiteral。本規格也指定一個 SystemLiteral，可以用來指出 DTD 所在位置。在文件類型宣告中的 SystemLiteral，應該指向 Internet 上一個可以透過本規格指定的 HTTP 協定存取的位置。

PublicLiteral 是用來辨識檔案的類型。對於本規格訂定的檔案類型，其 PublicLiteral 應該遵從下列語法：

```
"-//DVB//DTD " <document type> " " <version_number> "//EN"
```

其中：

<document type> 語法如下：

```
<document_type> = letter letters
```

```
letters = "" | letter letters
```

```
letter = uppercase_letter | lowercase_letter | space
```

```
uppercase_letter = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H"
```

```
                | "I" | "J" | "K" | "L" | "M" | "N" | "O"
```

```
                | "P" | "Q" | "R" | "S" | "T" | "U" | "V"
```

```
                | "W" | "X" | "Y" | "Z"
```

```
lowercase_letter = "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h"
```

```
                | "i" | "j" | "k" | "l" | "m" | "n" | "o"
```

```
                | "p" | "q" | "r" | "s" | "t" | "u" | "v"
```

```
                | "w" | "x" | "y" | "z"
```

```
space = " "
```

<version_number> 語法如下：

```
<version_number> = major_version "." minor_version
```

```
major_version = digit digits
```

```
minor_version = digit digits
```

```
digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
```

```
digits = "" | digit digits
```

PublicLiteral 中的 <document_type> 部分是用來辨識文件類型。此文件類型，若本規格將來的版本設計出更新、但向下相容的版本時，<document_type> 部分將不會改變，而 <version_number> 部份則會使用一個新的、該類型文件從未用過的版本編號。

XML 檔案對於 ExternalID 文件類型宣告內指定的 DTD 來說必須有效 (valid)。此處所謂「有效」(valid)，是根據 XML 1.0 規格的定義。

文件類型宣告不應該包含方括號 ([]) 括起來的宣告。

當 XML 元素中使用到 PCDATA 類型，其中並包含一個字串（譬如檔案名稱）時，這些字串不得包含「<」或「>」字元，因為這些會被誤認為是 XML 標記 (tag)。

註：如果字串中必須用到這些字元，則應該指定使用 CDATA 編碼，這樣雖然語法上比較複雜，但是卻能夠使用這些字元。

該檔案應該只包含本規格（或本規格未來版本）所定義的標記 (tag) 和屬性 (attribute)。

該檔案不能包含 XML entity 宣告 (<!ENTITY .> 標記)。

該檔案不得包含 XML 字元或 entity 參照（以「&」開頭的參照）。

該檔案不得包含 XML 處理指令，除了 XMLDecl 項目「<?xml ?>」以外。

該檔案可以包含 XML 註解（「<!-- -->」字串），但是不能出現在那些標示為 PCDATA（並包含必須依本規格編碼的字串）的元素當中。

MHP 終端機在處理 XML 格式檔案時的規則：

解析器 (parser) 應該使用 XML 文件類型宣告（「<!DOCTYPE >」）中的 PublicLiteral 來辨識檔案類型。

在文件類型宣告中的 PublicLiteral 會指出該檔案所使用的 DTD 版本。解析器沒有必要自行至 SystemLiteral 所指定的 URL 去取得 DTD。因此，由平台實做自行決定是否要到那個 URL 去取得 DTD。如果那個 URL 所指定的位置無法取得 DTD，該如何處理則由平台決定。

對於某一種檔案類型，解析器應該能接受 PublicLiteral 指定的版本與本規格指定的版本有所出入。這樣的檔案，應該是依照本規格不同版本所產生出來的。對於這種檔案，解析器應該辨認並處理所有本規格 DTD 中所定義的元素和屬性。

解析器應該略過所有本規格版本 DTD 中沒有定義的 XML 元素（開始標記、結束標記、以及兩者之間出現的字串）。

註：這樣的設計，讓將來在延伸 DTD 時能保持相容性，由於現在的 MHP 終端機會忽略以後新增的標記，因此不受影響。

解析器應該略過所有本規格版本 DTD 中沒有定義的 XML 標記屬性。

解析器應該略過所有的 XML 1.0 規格中所定義的 XML 註解。

對於本規格所定義的 XML 元素，解析器必須接受空的 (empty) 元素，包括有開始和結束標

記的形式，以及只有開始標記的形式（譬如「<tuning value="true"></tuning>」也可以寫成「<tuning value="true"/>」）。

只要遵循這些規則，就算本規格再怎麼變，也能維持相容性（也就是說，雖然後續版本可能新增一些 XML 標記和屬性，但是既有元素的意義將不會更改）。

如果檔案編碼方式違反上述的編碼規則，解析器如何處理將由實做決定，包括可能完全丟棄整個檔案，或是當作沒有這個檔案。

8.4 網路訊號

MHP 終端機在遇到所收到的資料格式不正確時，不管是經由何種方式傳送或取得，除了本規格或是本規格參照到的規格有明確定義錯誤處理方式之外，都是由平台實做自行決定如何處理。MHP 終端機在這種情況下可以採用任何方式。甚至可以將網路上收到的資料直接傳給應用程式而不做任何檢查。因此，如果從網路傳來的資料就已經不正確，本規格 API 所傳回的資料也可能不正確。

MHP 終端機必須遵守 ETSI TR 101 154 [9] 中第 4.1 節的規定。

註：強烈建議 MHP 終端機在設計時，應該考慮到將來 DVB SI\ DSMCC 或其他 MHP 終端機需要讀取的資料格式可能會延伸，因此要提供未來的相容性。當 MHP 終端機遇到自己沒有用到的「reserved」欄位 (ISO)、「reserved_future_use」欄位 (ETSI) 以及「user defined」欄位 (ETSI EN 300 468 [4])，都應該自行忽略。其中，「reserved」和「reserved_future_use」欄位將來在標準中可能會有其他用途，而「user defined」欄位則永遠不會標準化。當 MHP API 在提供這些資料時，MHP 終端機會直接將資料回傳給 MHP 應用程式而不做任何檢查或修正。

8.5 應用程式識別碼的文字編碼方式

當 organisation_id 或 application_id 以文字方式編碼時，應該遵照下列規定：

以十六進位的方式表示該值

使用小寫字母

前面不額外加零 (如同 Integer.toHexString 所產生的數值)

當 organisation_id 和 application_id 合併成一個應用程式識別碼時，將使用前述方法轉成單一的十六進位數字，其中 organisation_id 將靠在最高位元組，而 application_id 將靠在最低位元組。

8.6 檔案名稱要求

8.6.1 永久性儲存

接收器應該支援下列 BNF 所定義的 `persistentpath` 和 `persistentfilename`：

```
lowalpha = "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" |  
           "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" |  
           "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z"  
  
upalpha = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" |  
           "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" |  
           "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"  
  
alpha = lowalpha | upalpha  
  
digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |  
        "8" | "9"  
  
punct = "_"  
  
persistentfilechar := alpha | digit | punct  
  
persistentfilessubstring := persistentfilechar | persistentfilessubstring persistentfilechar  
  
persistentfilename = persistentfilessubstring  
                    | persistentfilessubstring "."  
                    | persistentfilessubstring "." persistentfilesuffix  
  
persistentpath = persistentfilessubstring | persistentpath "/" persistentfilessubstring  
  
persistentfilesuffix = persistentfilechar | persistentfilesuffix persistentfilechar
```

接收器必須支援：

小於等於 8 個字元的 `persistentfilessubstring`

小於等於 3 個字元的 `persistentfilesuffix`

接收器不一定要接受超過這些規定的檔案名稱，不過應用程式使用這樣的名稱就是不合規定。

接收器永久性儲存的檔案系統必須要「區分大小寫」(case sensitive) 或是「維持大小寫」(case

preserving)。應用程式撰寫時，則兩種情況下都要能運作。「維持大小寫」的檔案系統，在處理已經存在的檔案時並不會區分大小寫，但是會維持檔案建立時指定的檔案名稱大小寫。

8.6.2 DSMCC object carousel

接收器應該支援符合下列最低要求的 object carousel：

檔案路徑（也就是目錄名稱、分隔字元、檔案名稱等加起來）必須 ≤ 1024 位元組 (byte)。

在檔案路徑長度的要求內，檔案和目錄名稱，皆必須 ≤ 255 位元組 (byte)。

OC 檔案系統必須區分大小寫

註 1：目錄的層數並沒有限制，只要檔案路徑不要超過長度限制即可。

註 2：內容製作人員不應該絕對相信 OC 會區分大小寫，因為，有些內容製作平台不會。

註 3：為了讓使用傳統檔案系統的電腦在開發應用程式時更方便，本規格建議檔案的名稱應限制使用字元碼 0x21 到 0x7E 之間的字元，並排除 0x22、0x27、0x3A、0x3B、0x5C 等字（也就是排除英文的雙引號、單引號、冒號、分號以及反斜線）。

8.7 檔案和檔案名稱

MHP 規格有規定應用程式應該如何使用檔案名稱來存取檔案內容。不過，卻也刻意不談到 MHP 終端機的檔案系統和檔案系統名稱空間，不過，還是有下列幾點說明：

當一個 MHP 應用程式啟動時，應用程式所在的檔案系統，就會被掛載 (mount) 到 MHP 終端機的檔案系統名稱空間中。對於一個 DVB-J 應用程式來說，「6.5.1 廣播傳輸協定存取 API」一節有提到，可以用 `java.io.File(".")` 來產生一個應用程式基底 (base) 目錄的參照。此基底目錄可能是該檔案系統中的一個子目錄。

只要是獲准存取永久性儲存的 MHP 應用程式，就能存取永久性儲存檔案名稱空間。對於 DVB-J 應用程式來說，此一名稱空間的最上層目錄，可從「`dvb.persistent.root`」系統屬性獲得。

MHP 應用程式或許可以在其 MHP 終端機的檔案名稱空間上掛載額外的檔案系統。DVB-J 應用程式可以使用 `org.dvb.dsmcc.ServiceDomain` 類別的 `attach()` 方法來取得一個 object carousel 做為額外的檔案系統。若是在其他方法中使用了一個 DVB 定址器，而這個定址器包含 `dvb_abs_path` 部分，則不應該掛載指定的 object carousel 檔案系統。

符合規格的 MHP 應用程式，不應該嘗試存取本規格允許之外的檔案和檔案系統。違反這項規定會導致什麼結果，需視實做而定。實做平台可以使用平台的安全機制，來限制 DVB-J 應用

程式的存取權限，譬如使用 `java.io.FilePermission`。

在參照存放於檔案中的內容時，應該使用以文字編碼的檔案名稱，或是「file:」URL (如 IETF RFC 1738 [67] 所定義)。「dvb:」URL 中的檔案名稱編碼，在使用之前應該轉換成「file:」URL。對於 DVB-J 應用程式來說，檔案名稱應該使用 Java String 物件來編碼，而「file:」URL 應該使用 `java.net.URL` 的實例。請參閱 `ServiceDomain.getURL(Locator)`。

當建立一個使用「file:」協定的 `java.net.URL` 時，主機 (host) 部分就應該是空的，而路徑 (path) 部分則應該和傳入 `java.io.File` 建構函式中的一樣，並將平台本身使用的路徑分隔字元通通換成「/」而且，開頭第一個分隔字元應該捨棄。

8.8 定址器與內容參照

下表列出本規格定址器所能定址的個體。同時也定義出每一種個體的文字表示形式。如果沒有標準化的文字表示形式，則使用平台自己的表示形式。本規格並不要求 MHP 系統可以透過一個定址器或 URL 定址到其他類型的個體。

表 8-2：可定址個體、定址器、與其文字表示形式

個體	文字表示形式
傳輸串流	包含「dvb_transport_stream」的 DVB 定址器。
網路	沒有標準化的文字表示形式。
Bouquet	沒有標準化的文字表示形式。
DVB 服務	包含「dvb_service」的 DVB 定址器 (註 3)。
通用服務	除非此服務也是 DVB 服務，否則沒有標準化的文字表示形式。
DVB 事件	包含「dvb_service」和「dvb_event_constraint」的 DVB 定址器。
MPEG 基礎串流	如果基礎串流中含有元件標記 (component tag)，則應該是包含「dvb_service_component」的 DVB 定址器，否則就沒有標準化的文字表示形式。
檔案	如 IETF RFC 1738 [67] 所定義的「File:」URL (註 1)。 包含「dvb_abs_path」的 DVB 定址器 (註 2)。
目錄	如 IETF RFC 1738 [67] 所定義的「File:」URL (註 1)。 包含「dvb_abs_path」的 DVB 定址器 (註 2)。
涓流解碼器	「dripfeed://」
註 1：一個「file:」URL 中的主機部分應該永遠是空的字串。這些 URL 永遠都只能指到接收器的名稱空間內。如果透過網路傳出應用程式之外，將變得毫無意義。	

註 2：MHP API 可能會傳回包含「dvb_abs_path」的 DVB 定址器，用來參照到目前未掛載 carousel 中的檔案。這些定址器只能用來掛載新的 carousel，或是在新的 carousel 掛載之後用來轉成「file:」URL。

註 3：本規格不建議在 DVB 服務的參照中包含 transport_stream_id，因為 original_network_id 加 service_id 已經足以辨識該服務。

DVB 規格中有兩處定義，可以在一個 MPEG 基礎串流中包含多個邏輯服務元件：DVB 字幕和 MPEG-2 多軌音訊。本規格並未提供定址器來區分單一 MPEG 基礎串流中所包含的各國語言字幕，或是各軌的音訊。因此，那些傳回服務定址器或基礎串流定址器的方法，並無法區分這些邏輯服務元件，因此在這類邏輯元件之間切換時，所傳回的定址器是相同的。而那些接受服務元件定址器或基礎串流定址器的方法，將根據「6.4.2.2 釐清」一節所定的基礎串流選擇規則，來切換這些邏輯服務元件。

當標準化文字表示形式不存在時，可以用實做自己的表示形式來建構定址器，以便可以定址到原來可定址的個體，不過，這也只能用在該應用程式內。

註：這表示本規格並沒有要求這類應用程式專屬的表示方式，必須在從永久性儲存讀回來之後還能使用，或是可以傳到另外一個應用程式使用。

8.9 服務識別碼

在 MHP 中有兩種機制可以用來識別不同服務：

三個碼組成的 SI 識別碼：

original_network_id、transport_stream_id 和 service_id (對應 ETSI EN 300 468 [4] 中定義的廣播服務資訊所包含的三個相同名稱的識別碼)。

註：本規格不建議使用 transport_stream_id，因為 original_network_id 加 service_id 已經足以辨識該服務。

以文字表示的服務辨識碼：

就是 SDT 中選擇性的 Service identifier descriptor 內的 textual_service_identifier_bytes (請參閱「5.12.1」一節)。

這兩種方式，即使在全球都能夠提供唯一的服務識別資料。

文字形式的識別碼還有其他的特性：

可以將多個數字識別碼不同服務看成是相同的服務。

服務供應商可以自行決定兩個數字識別碼不同的服務，是否為相同的服務。

可以讓單一服務擁有多種識別方式。

8.9.1 文字形式識別碼的語法

一個文字形式的識別碼，其語法為：

```
<service_name> "." <service_provider_domain_name>
```

其中：

<service_name> 是一個服務名稱，此名稱在服務供應商網域內必須是唯一而不能重複的。

<service_provider_domain_name> 是一個服務供應商擁有控制權的 Internet DNS 網域名稱。本規格利用 Internet DNS 網域名稱來做為服務供應商的識別機制，這樣，服務識別碼才能夠全球通用。

<service_name> 欄位必須遵循 Internet DNS 名稱的命名規則，這樣，整個文字形式的服務識別碼才可以變成一個有效的主機名稱 (host name)，並用在 Internet DNS 當中，如 IETF RFC 1035 [75] 所定義。

底下是一個文字形式的服務識別碼範例：

```
movie-channel-1.broadcaster-b.com
```

其中「broadcaster_b.com」是該廣播業者所擁有的一個 Internet DNS 網域名稱，「movie-channel-1」則是該服務供應商所提供的特定服務名稱。

註：文字形式的服務識別碼與 Internet 主機名稱的命名語法相同，因此，必須定義於服務供應商擁有控制權的網域內。不過，該服務名稱並不需要可以透過 DNS 服務對應到某個 IP 位址，不過，如果真的可以對應到一個 IP 位址，本規格並沒有規範如果有人用 IP 通訊協定與這個位址進行通訊時，該主機應該提供什麼服務。

8.9.2 MHP 終端機如何處理文字形式的服務識別碼

MHP 終端機取得某一服務文字識別碼的方式，就像當初發覺該服務一樣，是從 SDT 表格而來。MHP 終端機取得可用服務的文字識別碼，與取得服務的數字識別碼 (original_network_id、transport_stream_id 和 service_id) 方法是相同的。

當應用程式在使用一個指向 DVB 服務的 URI 時，不論該 URI 所包含的是文字識別碼或是數

字識別碼，其解析到服務的過程是相同的。

應用程式所提供的 URI 字串，如果與 SDT 中的某個字串相同時，就算是相符。

8.10 CA 系統

此處所謂的「CA 系統」泛指各種實做方式的 CA 系統，因此涵蓋嵌入式 CA 系統，以及透過 DVB Common Interface 所實做的系統。

如果 MHP 應用程式可以看到某個 CA 系統，則應用程式也可以透過所有與 CA 相關的 API 存取到該 CA 系統，譬如：

CA API

`org.dvb.media.PresentationChangedEvent` 中與 CA 相關的原因代碼

`javax.tv.service.selection.PresentationChangedEvent` 與子類別

8.10.1 服務選擇

當 CA 系統造成當前服務改變時，其效果應該與服務選擇相同。

如果拿 DVB Common Interface 來說，就是當某個模組發出 Host Control tune 請求時。此時應該發出一個 `javax.tv.service.selection.AlternateContentEvent` 事件。

8.10.2 媒體元件選擇

如果 CA 系統要求改變媒體組成元件（音訊、視訊或字幕），這樣的改變應該自動執行（其效果等同於使用 `javax.tv.media.MediaSelectControl`），同時應該透過

`org.davic.net.ca.PIDChangeEvent` 和

`org.dvb.media.PresentationChangedEvent` 通知應用程式，其中原因代碼應指定為 `CA_FAILURE` 或 `CA_RETURNED`，視其所用的事件而定。通知時應該傳送

`MediaSelectEvent` 給所有目前註冊的 `MediaSelectListener`，來改變其當前的選擇。

如果拿 DVB Common Interface 來說，這就是當某個模組發出 Host Control replace / clear_replace 請求時。此時應該分別發出

`javax.tv.service.selection.AlternativeContentEvent` 和 `NormalContentEvent` 事件。

8.10.3 非媒體元件選擇

如果 CA 系統要求改變「非媒體」組成元件 (譬如 DSM-CC Object Carousel 和私有資料), 這樣的改變「不應」自動執行, 而是應該透過 `org.davic.net.ca.PIDChangeEvent` 通知應用程式來處理。

如果拿 DVB Common Interface 來說, 這就是當某個模組發出 Host Control replace / clear_replace 請求時。

附件（二） 出國報告

- | | |
|---|--------|
| 1. 2004 年新加坡亞洲廣播通訊展 , Broadcast Asia 2004 | 附 2-1 |
| 2. DVB 出國報告 | 附 2-19 |
| 3.出國報告-中國 | 附 2-21 |

2004 年新加坡亞洲廣播通訊展

Broadcast Asia 2004

一．前言

2004 年新加坡亞洲廣播通訊展，於 93 年 6 月 15 日至 90 年 6 月 18 日在新加坡展覽場(Singapore expo)所舉辦，去年因 SARS 的關係而停辦一年，今年參展國達五十國，共三百八十八家廠商，一千一百五十四個攤位，此外，大陸視此次參展為推動國際化發展的重要參展，參展廠商多達七十七家，是這次參展規模最大國家。新加坡亞洲廣播通訊展是結合了廣播器材及有關通訊方面的廠商結合起來一起展出，其中廣播器材是在展覽場所之 HALL 1 及 HALL 2 大部分空間所舉行，通訊器材則是在 HALL2 部分空間及 HALL 3~HALL6 舉行。廣播方面主要有數位語音廣播設備及接收機、節目製作設備廠商、頭端設備廠商、機上盒廠商、儀器廠商及相關應用軟體及服務廠商等。通訊器材規模則大多了，展覽空間是廣播器材面積 3 倍左右，其涵蓋通訊器材、網路設備、網路服務提供者、光纖通訊等..。此行的目的主要是針對廣播部份尤其是目前數位電視相關產品更為主要著墨之處。



(圖一：新加坡展覽館平面圖)

此行原本是跟隨著由國內產、官、學所組成的考察團隨行參訪，並安排參訪 Media Corp 及 SBA 兩家公司，並實際體驗 Mobil TV 移動電視，很可惜此考察團

因故無法成行，改為自行參訪，因此沒法體驗移動電視的魅力了。同時，因為我們自行參訪的時間較短，所以只能重點參訪，較無相關的產品及廠商也就沒有時間多加探討了。

在展覽期間，也舉辦了多場國際研討會，這些研討會在科技發展上提供了重要的新知。

另外，這次參展也看到了有關 DVB-H 的相關設備級產品，但是只有少數幾家，而且不夠完善。

接下來我們將這次的參訪就儀器設備 接收端 DVB-MHP (Multimedia Home Platform)、IPTV (Internet Protocol TV) 及其他等分別詳加討論之，並在最後做總結。

二. 儀器設備

此次亞洲廣播展中在數位電視相關的領域方面，儀器設備類的廠商數量最多、規模也較大，主要在分佈在兩種類型：1. 測試及量測設備 (Test & Measurement Equipment)。2. 頭端設備 (HeadEnd) – 包含了調變器 (Modulator)、視訊壓縮編碼器 (Encoder) 以及發射器 (Transmitter)

數位電視開播後，數位電視產業逐漸興起，不管是正要投入或已經投入這個產業的廠商對於各項設備的需求大增，最直接受益的就是這些設備業者。尤其在亞洲，數位電視屬於正起步階段，因此 2004 年亞洲廣播展當然是這些設備業者不會放過的戰場。

1. Test & Measurement Equipment :



Ro.ve.r：一家義大利公司。主要針對數位電視調變訊號不管是 DVB-T、DVB-C、DVB-S 提供一系列的量測儀器，值得一提的是 Ro.ve.r 提供一完整系列小型化與輕便化的簡易量測儀器，適合攜帶外出，使工程人員不必被侷限在實驗室。此系列產品主要針對 QPSK、QAM、COFDM 單一調變訊號提供簡易量測與監控數值。除此之外也提供一般混合型量測平台，主要是 COFDM 與 QAM 現場並有天線直接接收隔壁攤位發送的訊號作為展示。據說台灣業者中嘉也已經採用他們的產品。



(圖二：Ro.ve.r所展示的量測儀器)



R&S：我想 Rohde & Schwarz 這家公司深為我們熟知，在亞洲廣播展中是不會缺席的在，這裡我們不多加介紹。他們的 SFQ 系列也為我們及國內多家業者所採用。

2. HeadEnd：此次參展頭端設備業者主要可區分成兩種類型：A.調變器與視訊編碼器 - 此類型以 Harmonic 在展場中佔地較大，為自有的視訊編碼器搭配其他廠牌的調變器構成一頭端系統。 B.發射器 - 展示整套發射端系統，搭配多套調變器與視訊編碼器為大型發射台系統，這方面設備與我們較無相關，簡單列出廠商資料，僅供參考。



Harmonic：在北美市場市佔率第一的頭端廠商，積極發展 MPEG4 Part10 壓縮技術。現場主要展示多套視訊壓縮編碼系統，分別支援 MPEG2、MPEG4 Part10、VC9 (VC9 為微軟主導推出之視訊壓縮編碼規格)。DVB 標準採用 MPEG2 技術為壓縮規格，為現階段數位電視業者所採用。MPEG4 Part10 規格能將訊號壓縮的更小，因此可解決網路視訊訊號流佔用頻寬的問題，對於 IP Box 的營運業者相當具有吸引力，據亞太區市場總監喻彤表示約在今年 8 月份可推出 MPEG4 Part10 壓縮產品機型，Harmonic 所有壓縮編碼器均可以使用軟體更新方式從 MPEG2 規格更新 MPEG4 規格，甚至 VC9 壓縮編碼規格，對期待採用 MPEG4 或 VC9 規格的業者是一大福音。主要透過乙太網路接口(Ethernet port)傳送壓縮型態資料。在展場亦安排有演講與展示，不過因為時間的關係我們無緣見到。



(圖三：Harmonic 所展示的編碼器)



(圖四：Harmonic 亞太區市場總監喻彤/照片來源：電子時報)



Radyne ComStream：主要展示產品為各系列 MPEG2 壓縮編碼器與通訊調變器含 QPSK、QAM、COFDM 等調變系統。我們也有採用此公司的 QAM 256 Modulator 產品。

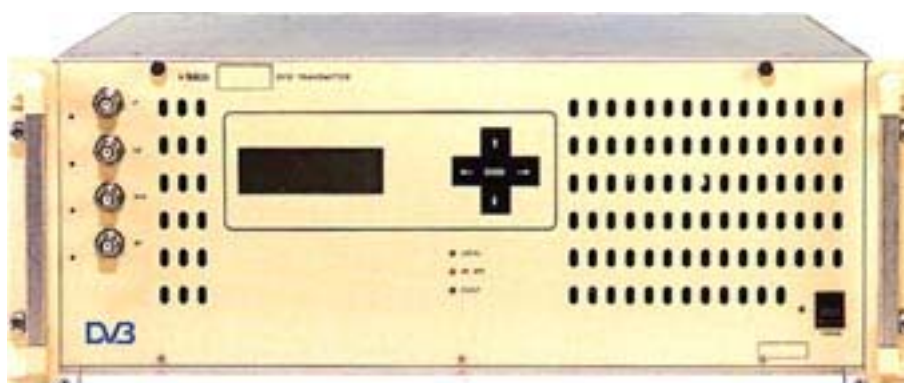


QAM256 Digital Video Modulator and Cable Upconverter

(圖五：Radyne ComStream 所展示的 QAM 256 調變器)



Electrosys：提供一系列數位電視 (DVB) 及數位廣播 (DAB) 發射端設備。在數位電視方面，現場展示以整套地面廣播 (DVB-T) 設備為主，包含 MPEG2 壓縮編碼器、COFDM 調變器與功率發射器。



(圖六：Electrosys 所展示的 DVB-T 調變器)



(圖七：Electrosys 所展示產品系統圖)

Harris：Harris Corporation 國際級通訊設備的廠商，也是屬於相當知名的數位電視設備廠商。主要合作對象為政府組織及通訊廠商，在現場展示了一套數位電視 (DVB-T) 發射端設備，現場不喜歡他人拍照，不過我們還是偷拍到了一張珍貴的照片。



(圖八：Harris Corporation 所展示數位電視 (DVB-T) 發射端系統設備)

三. 接收端

這一次的展覽公司中，仍有許多家廠商展示其機上盒，大部分是韓國廠商，中國及台灣其次，接下來我們簡略說明這些廠商的產品及規格：

1. Arion Technology Inc

韓國公司，主要以 DVB 規格的產品為主，衛星廣播部分，包含 Free to Air 衛星接收機上盒及條件存取 CA(Conditional Access)衛星接收機上盒，另外，接收地面廣播的機上盒也有產品。

2. Aston Korea Corporation

韓國公司，產品包含消費性及專業級的條件存取模組，機上盒及天線定位用 Positioner。

3. Eastern Electronics Co, Ltd(泰山電子)

台灣公司，主要產品有 DVB-S, DVB-C 及 DVB-T 機上盒，並擁有 Free to Air, CA 及個人影像錄製 PVR(Personal Video Recoder)等功能。

4. Gospell Digital Technology Co, Ltd

中國公司，有 DVB-S 及 DVB-C 機上盒產品。

5. Hyundai Digital Technology Co,Ltd

韓國公司，有完整的機上盒系列產品，衛星機上盒，有 Free to Air, CI, CA, MHP(Multimedia Home Platform), PVR 等功能。有線機上盒，有 CA 功能。地面廣播機上盒，Free to Air 及 PVR 等功能。

6. Kaon Media Co,Ltd

韓國公司，有完整的機上盒系列產品，含 Free to Air, CI, CA, MHP 等功能外，並擁有兩系列 Combo 的機上盒產品，分別為可同時接收衛星地面及可同時接收衛星有線。另外，也有 ATSC 規格的機上盒。



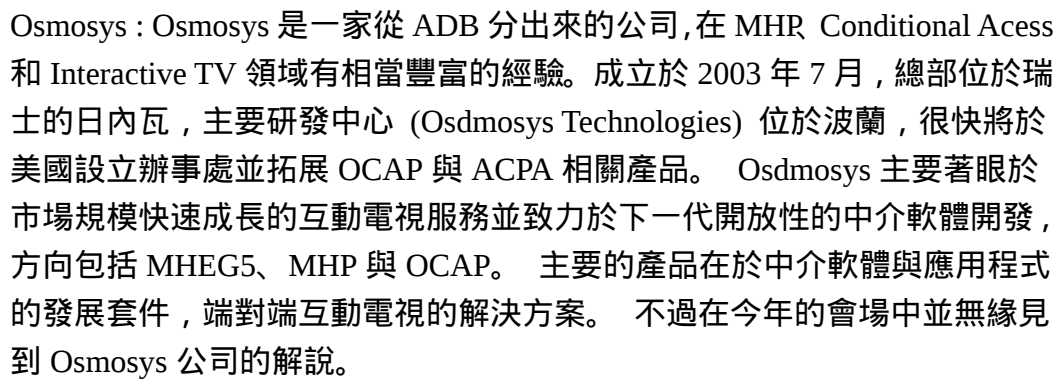
(圖九、圖十：Kaon Media Co,Ltd 所展示數位電視機上盒)

四. DVB-MHP

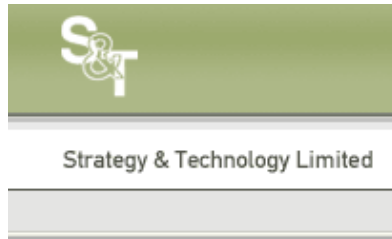
在今年的亞洲廣播展, DVB 組織本身也有參與展出，攤位主要推廣 DVB-MHP 和兩個新的標準 DVB-S2 及 DVB-H:

1. DVB-S2:the most advanced satellite distribution technology available today

- 並有 DVB 的專家在現場為大家解說，並結合多項由 DVB 組織成員所提供的展示: ADB、Irdeto Access、Osmosys、Rohde & Schwarz 和 Strategy & Technology with SysMedia。



(圖十一：Osmosys 所展示 MHP/OCAP/ACAP 架構圖)



Strategy & Technology with SysMedia : Strategy & Technology Ltd.此次與 SysMedia 一起合作，展示了一套可同時支援 MHP 與 MHEG 的電視字幕系統。由 SysMedia 提供 Plasma Magenta 搭配 Strategy & Technology 的 TSBroadcaster 與 TSPlayer，組合成一套可生成符合 MHP 與 MHEG 串流的系統。



(圖十二、圖十三: Strategy & Technology with SysMedia 共同展示的電視字幕系統)



ADB：遠昇科技（Advanced Digital Broadcast；ADB）也就是大家所熟知的波蘭公司 ADB 在今年的展場中，展出多款支援多媒體家用平台（MHP）的數位機上盒，不僅僅是有地面廣播（DVB-T）衛星（DVB-S）或有線（DVB-C）型態的數位機上盒，也推出擁有隨選視訊，線上遊戲，電子郵件功能的網路數位機上盒（IP-STB）。遠昇科技在多媒體家用平台上可說是世界級的領導廠商，不但本身展出多款機上盒，許多與多媒體家用平台相關的應用廠商也多使用 ADB 的機上何來作為展試驗證平台(如 Zentek)。



(圖十四: ADB 展示的 DVB-MHP STB)



Zentek: 這家公司是一家日本公司，成立於 1997 年。集團的機構分布於日本美國與新加坡，它是 DVB、CableLabs、Open mobil Alliance 和 ARIB 的組織成員。雖然它是 DVB 的組織成員並展示和 MHP 相關的發展軟體，但它的展場並不與 DVB 展場一起，主要的展示項目有兩個: MediaStack 與 MediaMaster。

MediaMaster 提供圖形設計者和內容創作者適用於 DVB-MHP 與 OpenTV 平台的應用發展工具。 MediaStack 則為開放標準的互動電視應用程式與內容開發提供強有力的執行環境。 Zentek 並可幫助客戶將 MediaStack 作為中介軟體移植到多種硬體平台。 2003 年 8 月通過 DVB-MHP v1.02 認證。 不僅如此，Zentek 更在新加坡當地成立互動電視內容製作中心，提供端對端 DVB-MHP 互動電視內容與應用程式解決方案， 成為新加坡地區第一個 MHP 內容製作中心，為互動電視的內容製作、測試與驗證提供了完善的環境，另外不可免俗的 Zentek 藉由與 Tcom&dtvro 的合作，在現場也展示了網路視訊數位電視機上盒系統的解決方案，提供網路視訊的服務。



(圖十五：Zentek 展示 MediaMaster 生成的畫面於電視接收端效果)



(圖十六：Zentek 展示利用 MediaMaster 編輯圖形)

DVB-MHP 標準已經走了一段時間，在今年的亞洲廣播展中，雖然現場參與展示的廠家只佔一小部分，但是 MHP 的影響力漸增確實是明顯的趨勢，很多多媒體相關產品都打出 MHP 支援、相容的功能，例如 SIDA 所推出 PC-Base 的 DTV received Module 與台灣廠商 KatDC 展出的多媒體應用軟體都號稱可支援或提供 MHP 功能。但是受限於現有數位電視內容不夠豐富與多樣化，多媒體家用平台還無法在今年的亞洲廣電展中成為數位電視的主角。另一個趨勢是幾乎所有 MHP 的廠商都強烈的想要跨出歐規的侷限，不但適用於 MHP、OCAPE 甚至支援 MHEG。以往業界部份人士認為數位機上盒僅是一項過渡性產品，事實上隨著中介軟體的標準日趨確定之後，數位電視廠商不需因為各地系統差異開發不同產品，而數位電視功能也能以模組化形式出現，甚至轉而內建至單一電視機當中。使得內建型式的數位電視機亦成為另一趨勢。

相關報導” ADB 9 月份將在義大利推出 17 吋的 MHP LCD TV，接下來則會推出 20 吋以及 30 吋的機種，而未來也會推出內建 802.11g 的電視產品，消費者透過電視畫面控制家中的資訊網路。另外，由於 Philips 以及 PACE 也將推出類似產品，STB 的各項功能，包括 HomeGateway 等功能，也將陸續內建至電視當中。”(資料來源：電子時報)

五. IPTV

這一次參展當中，IPTV 的廠商雖然不多，但是仍然有讓人驚喜的產品，其中有一家以色列公司叫 Infogate Online 推出一套產品叫 OnDema，被 Swisscom、Maltacom 等多家公司所採用，這次參展有展示這項產品如圖所示，Infogate Online 研發及銷售 OnDema 這項產品，是一套能夠讓電信業者、DSL 業者、有線業者、寬頻 ISP 業者及衛星業者提供給他們的客戶有高品質的 On Demand 內容及服務的軟體平台，例如提供給客戶高品質的 VOD (Video On Demand)、電視頻道、Pay per View 及 GOD (Game On Demand) 等服務，OnDema 是唯一一套能夠管理及傳送 IPTV 及 PC 家庭娛樂服務在相同平台的整合性軟體。接下來我們舉這項產品為例：



(圖十七：Infogate 展示會場)



(圖十八：OnDema 主畫面)



(圖十九：OnDema 引導畫面)



(圖二十：OnDema 中文化介面)

1.使用 OnDema 的 IPTV 架構圖

如圖所示是使用 OnDema 的 IPTV 的架構圖, OnDema 的分配 Scalable 及 Teco-grade 的設計, 可依照市場的需求做內容的調整, 可使用於小區域、整個國家或跨國, 且可以使用 Unix 及 Windows 平台, 整個架構主要有四部分如下：

(1) 內容提供者入口 (Content Provider Gateway ; CP Gateway)

內容提供者入口使的內容提供者能夠定義、上載及分配他們的內容, 包含內容的包裝、定價及保護。

(2) 服務提供者主要的單元 (Service Provider Master Unit)

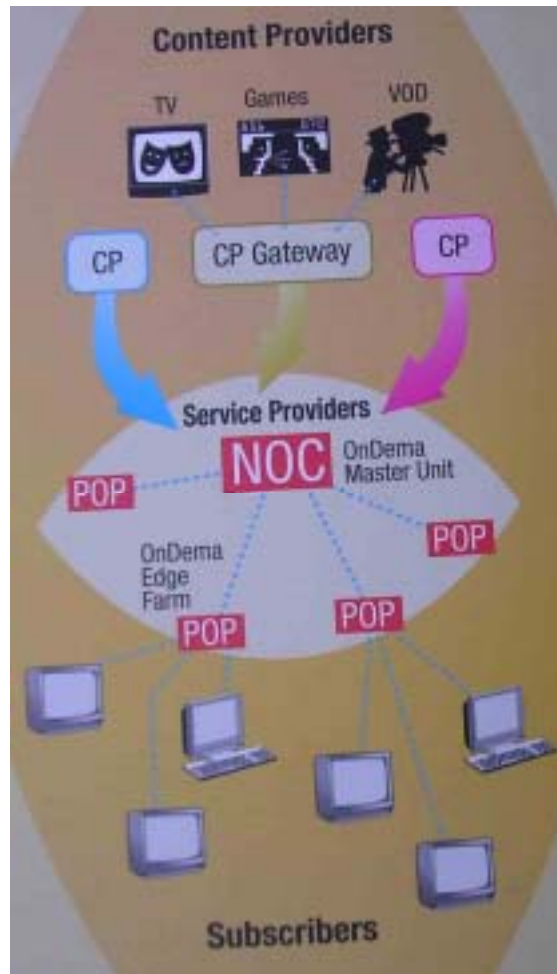
服務提供者主要的單元提供網路階層的管理, 確保內容被正確地分配, 它同時管理內容提供者、消費者的授權及保密。

(3) 邊際伺服器 (Edge Farm Servers)

邊際伺服器主要是傳送電視節目、遊戲及其他服務, 主要的格式是 MPEG 及 Windows Media formats。

(4) 消費者介面 (Subscriber Interface)

此介面提供適當的存取介面給消費者, OnDema 的 SDK 容易設計出符合客戶所希望的 IPTV 及 PC 介面。



(圖二十一：使用 OnDema 的 IPTV 架構)

2. OnDema 架構

OnDema 提供了從內容的擁有者到消費者終端到終端的管理及內容傳送，下圖是 OnDema 的架構，我們依序說明之：

(1) 內容管理 (Content Management)

內容提供者入口 (CP Gateway) 是一套強大的內容管理軟體工具，包含：

- 內容等級
- 產生 Packages 及 Campaigns
- 定義價格模式及使用團體
- 定義分配模式及區域

(2) 內容加密 (Content Security)

內容加密能夠確保內容的所有權及拷貝權被保護，包含

- Fraud prevention
- 數位所有權管理 (Digital Right Management ; DRM)
- 條件存取 (Conditional Access)

(3)內容分配 (Content Distribution)

OnDema 管理從內容提供到邊界的內容分配，使得服務提供者能夠管理內容提供者的多種內容形式，如同提供電視頻道、VOD (Video on Demand、NVOD (Near Video on Demand) 及遊戲等。

(4)計費及定價 (Metering and Pricing)

OnDema 基於消費者所所使用的服務內容來計價收費，計價的方式有多種，可採用月租的方式、會費的方式或服務包套的方式，一個 API (Application Interface)將會帳單傳送給相關收帳單位。

(5) 消費者管理 (Subscriber Management)

OnDema 使用登錄、確認(authentication)及授權等功能來對消費者做管理，服務提供者能夠定義出不同的消費者提供不同的服務選擇，消費者也能夠從帳單上去管理內容如下：

- 父母親能夠控制內容的等級、使用時間及花費限制等
- 家庭帳單及子項目服務帳單
- 線上存取及過去的帳單

(6)傳送引擎 (Delivery Engines)

邊際 (Edge Farms) 傳送消費者所要求的電視頻道、影像、遊戲及其他內容給消費者，服務提供者能夠選擇較節省成本的伺服器來傳送 MPEG、Windows Media 或者其他格式給消費者，OnDema 的獨特遊戲伺服器藉由減低延遲時間來提高遊戲的速度。

(7)描述層及消費者介面 (Presentation Layer & Subscriber Interface)

這一層是針對服務消費者的介面，OnDema 強大的 SDK 能夠讓服務者、ISP 業者及內容提供者設計出屬於自己的介面及商標。



(圖二十二：OnDema 架構)

3.OnDema 的特性：

(1)服務：

- VOD (Video On Demand)
- Nvod (Near Video On Demand)
- 電視頻道
- Pay per view
- GOD (Game On Demand)
- 教育內容
- EPG (Electronic Program Guide)/IPG (Internet Pointer Guide)
- 網路 PVR
- 個人喜好內容的歸列
- 家庭帳單及子項目服務帳單
- 功能限制設定，例如收看的時間、節目內容、花費等等
- 線上存取到到客戶的帳單

(2)管理：

- 用 Wizard 定義新的內容
- 內容的評價
- 內容的包裝
- 價格及銷售的競選
- 促銷及試驗
- 控制內容分配
- 使用團體
- 數位所有權管理 (Digital Right Management ; DRM)
- 條件存取 (Conditional Access ; CA)
- Third party 內容的合約管理
- 內容使用報告

(3)格式

- Windows Media 9 系列
- MPEG-1、MPEG-2 及 MPEG-4
- RealPlayer
- QuickTime

(4)平台的特色

- Unix 及 Windows 版本
- 可調整外觀尺寸及健全分散式架構
- Telco-grade 效能
- 立即的及可安排的内容分配
- 多重内容提供者的支援
- 線上或不在線上的客戶登錄
- 計費系統介面

- 消費者保護系統介面
- 使用者確認
- 支援內含 Linux、Windows 及 embedded XP 的機上盒
- PC 使用者

六. 總結

2004 年亞洲數位廣播展於 6 月 18 日落幕，那時我們已經離開新加坡這個城市。綜合此次 3 個展覽來說，數位電視並不是這次展覽中最具人氣的焦點，不過相關產業參展的廠家其實不在少數，更可觀的是視訊多媒體隱然成為未來市場的主流，在數位電視如此、網路如此、手機產業更是如此，而且使得各產業間的分歧正在縮小，DVB-H、IP-Box 的出現正是這趨勢的證明。這次展覽我們看了幾個大焦點：數位電視系統、可撥放視訊影片的手機、網路隨選視訊系統。撇開手機不談，網路隨選視訊已是多家廠商競爭的重點項目，MPEG4 與 VC9 的出現更是帶給這些網路業者更大的能力與希望，此次展中從頭端設備業者所展示的各項設備就是最好的證明。

此外，中國大陸此次參展家數較上屆成長一倍，共有七十七家廠商參展，也是這次參展規模最大的國家，可見中國大陸已經成為亞洲各國強不可忽視的競爭對手，至少在量方面已經超過亞洲各國，是不容忽視的警告。



(圖二十三：日本松下展示可撥放視訊影片的手機)

工業技術研究院
電腦與通訊工業研究所
視訊與光通訊技術組
900221 廖益正
900655 周燕郎



DVB 出國報告

(2004 June 20-27)

一、出國摘要：

- (一)時間：2004 年 6 月 20-27 日。
- (二)地點：DVB/EBU (Digital Video Broadcasting Organization/European Broadcasting Union) Geneva, Swiss (瑞士，日內瓦)。
- (三)人員：王鴻智。

二、目標：

- (一)參加 DVB 第 56 屆 Technical Module (TM) 會議。
- (二)協助建立台灣政府部門(交通部電信總局)與 DVB 的官方聯絡管道。
- (三)向 DVB 取得台灣數位廣播內容與服務產業專用的 Original Network ID 與 Network IDs。
- (四)邀請 DVB Executive Director, Peter MacAvock 於 2004 年底來台灣參觀訪問。

三、內容：

- (一)DVB 第 56 屆 Technical Module (TM) 會議是 DVB 的技術規格訂定大會，約每季再 DVB 總部日內瓦舉辦一次，由所有的會員參加，並 Approve TM 所屬的各個技術規格制定小組所提報上來的結論。本次會議的主要會議結論如下：
 - 1. 同意由 HEAD Ad Hoc Group 所提出的新的頭端介面規格，SIMCOMP。
 - (1) SIMCOMP 是一個介於 Simulcrypt 與 Multiplex Configuration (MUXCONFIG)元件間的介面格式以支援頭端的 Multiplexer Redundancy 服務。
 - (2) SIMCOMP 支援 N:M 組合的 Hot 或 Cold Redundancy 方式。
 - (3) SIMCOMP 定義了 MUXCONFIG 至 SIMCOMP 之間的 Connective Information 與 Signal Redundancy Switch Over Event。
 - (4) TM 將本規格草案送交給 ETSI 審查。
 - 2. 提出 DVB-H IP Datacast 的技術草案要求：
 - (1) 同意由 CBMS (Convergence of Broadcast and Mobile Services) Ad Hoc Group 所提出的 IP Datacast 技術草案要求 (Call for Proposal)。
 - (2) 預定在 2004 年九月十六日前彙整所收到的提案，並於第 57 屆 TM 會議中討論。
 - 3. DVB 應與 OpenCables 共同討論 MHP PVR Extensions 規格草案與功能測試
 - (1) 目前 TAM (Technical Application for MHP) Ad Hoc Group 共有兩個 MHP PVR 的 Proposal。分別由 NDS 與 OpenCables 所提出。
 - (2) TM 就未來技術整合的觀點，偏重於採用 OpenCables 所提的方案。
 - 4. 同意 TM 與 Open Mobile Alliance (OMA)和 DSL Forum 技術交流與合作。
 - 5. 下次(第 57 屆)TM 會議時間定為：9 月 22-23 日。
- (二)已成功協助建立台灣政府部門(交通部電信總局)與 DVB 的官方聯絡管道，並轉交交通部電信總局廣電技術處洪仁桂處長送給 DVB Executive Director, Peter MacAvock 的親筆信，與取得 Peter 給洪處長的回函。
- (三)取得台灣數位廣播內容與服務產業專用的 Original Network ID 與 Network IDs。



1. 已取得 Original Network ID 值為：0x209E
2. 已預留台灣專用的 Network IDs 值為：0x3301-0x3400
3. 上述值將於 6 月 28 日後生效，依<http://www.dvb.org/index.php?id=16>網站公佈為準。

(四) Peter 同意於 2004 年 11 月 27-30 日訪台，並參加國際論壇與演講。細節由電通所公關部門洽談中。



出國報告-中國

(2004 July 31 ~ August 8)

一、出國摘要：

- (一)時間：2004 年 7 月 31 日至 8 月 8 日。
- (二)地點：中國上海、北京。
- (三)人員：王鴻智、龐德俊、黃淑敏。

二、目標：

- (一)參加第五屆中國國際大屏幕顯示設備與數字電視產業發展戰略研討會。
- (二)拜訪 DVB-MHP 相關廠商“上海高清藍信軟件技術有限公司”、“上海全景數字技術有限公司”及“中國數字電視產業聯盟”。
- (三)邀請上海高清藍信軟件技術有限公司，廖洪濤博士於 2004 年底來台灣參觀訪問。
- (四)蒐集國內外數位電視多媒體平台相關標準現況，以供制定國內數位電視多媒體平台之共訂規格參考，並期完成“數位廣播電視系統整合之研究”報告。

三、內容：

(一)第五屆中國國際大屏幕顯示設備與數字電視產業發展戰略研討會：

1. 研討會內容以大屏幕顯示設備為主，討論範圍涉略超大屏幕 CRT、陰罩式 PDP、LCD、LED、OLED 顯示技術等；而“HDTV in China and ST solution”、“未來數字電視發展的主流是液晶電視”、“關於數字電視與數字影院的現狀與發展”則與數位電視相關。
2. 中國電子視像行業協會林元芳常務副會長以電子信息產業經濟運行與發展概況提出(1)電子信息產業成為國民經濟支柱產業，是新世紀的戰略性產業、(2)電子信息企業結構逐步向大公司為主，中小企業互相配套的格局轉變、(3)產品門類基本齊全，軟件和集成電路成為新的亮點、(4)投資類產品成為市場的主導產品和拉動經濟增長得主要動力、(5)多種經濟成分並存發展，為產業注入新的活力、(6)形成國內市場國際化，國際競爭國內化的新格局、(7)IT 產業聚集使產業鏈日趨完善，具有特色的區域經濟正在形成、(8)技術和市場相互促進互為制約聯繫更加明顯、(9)市場規模取決於宏觀經濟需求的增長和技術發展創新。
3. 意法(北京)半導體公司邱金德經理報告主題為“HDTV in China and ST solution”，主要針對高解析度數位電視 HDTV 提出 ST 的策略方案，並針對“satellite- DVB-S2 finalized summer 2003”、“DVB for Handhold market in trial in Europe”、“Video Codec-MPEG4 AVC finalized in 2003”、“Audio MPEG2 AAC”、“Connection interface”提出目前新技術發展；另外，邱經理亦認為目前中國在 HDTV 的策略並不明確，HDTV 的標準未完成，例如：HDTV 節目的頻寬要求一定要是 SDTV 的 4-5 倍嗎？傳送的 Video 是 50Hz 還是 60Hz？以及 China codec 等。
4. 在“關於數字電視與數字影院的現狀與發展”中，提及 8 英寸以下的液晶顯示已成為個人移動信息顯示的主體，依據功能需求的不同，可以選不同的 TN、STN、TFT 等不同的 LCD

產品，屏幕的尺寸和顏色亦可以任意選取。12-20 英寸的液晶顯示可以根據需要做成應用顯示器和多媒體顯示器，是個人大信息量顯示應用的主體。而 20-37 英寸的液晶顯示，將成為未來高清晰度，數字電視顯示發展的主流。

(二) 拜訪 DVB-MHP 相關廠商，訪談內容成果：

1. 這次出訪中國，順利地訪談中國數字電視中間件軟體標準制定小組的組長廖洪濤博士，並經由訪談過程中得知製定中間件標準的情形與目前的狀態；另外亦針對提供前端數字廣播與機頂盒中間件部份，討論可能合作方向，以提供中國與台灣廠商的商業技術橋梁。
2. 目前已知在提供前端數字廣播方面，上海高清已與廣西、湖北內蒙簽約，並於南京、濰坊、濟南試用，另外上海高清亦與廠商技術合作，並收取移植(技術支持,source code)費用及權利金之部份費用。
3. 另外，中國數字電視中間件標準制定小組在制定標準時，與 DVB-MHP 最大不同在於拿掉 JavaTV 部份僅留 TV Guide SI selection package，拿掉不符合市場需求部份，並加上需要的 component，與 DVB-MHP 相比，約有 30%~40%的改變。
4. 至於，在數字電視宣導方面，中國亦開始積極向民眾宣導關於數字電視的諸多觀念，例如，什麼是數字電視？為什麼要改用數字電視？數字電視的好處為何？如果由模擬電視改用數字電視，民眾需要多付費嗎？.....等等問題，並請專家學者在中央電視台直接解答民眾的疑惑，並與現場觀眾互動，探討數字電視，個人覺得中國在宣導數字電視部份的方式，有些部份是值得我們參考與學習的。
5. 中國數字電視產業聯盟主席，郭柯主席提出大陸數字電視發展策略為：
 - (1) 吸收國外先進技術成果，自主制定數字電視標準。
 - (2) 以標準清晰度和有線數字電視為切入點，啟動數字電視市場和產業。
 - (3) 面向兩個市場，構造數字電視產業鏈。
 - (4) 採用“機卡分離”的數字電視終端產品技術政策，建立水平的市場模型。

郭主席並提及現今所面臨的主要問題有：

- (1) 節目制約，包括節目不足、數字電視節目製作能力不足以及節目管制政策等。
- (2) 廣電運營體制與市場化程度不適應發展，包括廣播電視行業的體制改革尚處於起步階段及廣播電視運營體制正在向市場化的方向轉變。
- (3) 產業技術儲備不足和結構缺陷，包括產業在數字電視前端設備、專用集成電路、核心軟件、新型顯示器件和數字電視生產檢測專用器設備等領域的研發投入強度不夠，產業化準備不足，以及配套政策不完善使得數字電視產品的產業化處於被動。
- (4) 缺乏可盈利的商業模式。

(三) 參觀第五屆中國國際大屏幕顯示設備及光電產品展覽會：

會場展示絕大部分為顯示設備與光電產品，除此之外尚有一些 DTV 設備廠商參展；例如北京牡丹電子集團有限責任公司所展示的 DTV 設備有“DTV 專用信號發生器”、“HDTV 流發生器”、“HDTV 誤碼測試儀”、“HDTV 播放機”等，而這些設備對頭端廠商而言都是不可或缺的工具。

(四) 中國數字電視中間件軟體標準制定小組組長廖洪濤博士同意於 2004 年 11 月底訪台，並參加國際數位電視高峰論壇暨研討會。

附件（三） 會議記錄

- | | |
|--|--------|
| 1. 台灣數位電視產業聯盟技術發展工作小組第一次座談會－探索數位電視技術的需求與發展方向 | 附 3-1 |
| 2. Network ID 註冊機制的討論－日本 Data Component ID 的做法 | 附 3-4 |
| 3. 台灣數位電視產業聯盟標準制定工作小組第一次座談會－台灣數位電視產業技術規範研討 | 附 3-6 |
| 4. 無線數位電視多媒體共通平台技術規範研討會 | 附 3-11 |
| 5. 「數位廣播電視系統整合之研究」座談會 | 附 3-12 |



會議名稱	台灣數位電視產業聯盟技術發展工作小組第一次座談會 －探索數位電視技術的需求與發展方向		
時 間	自 93 年 5 月 19 日 13 時 30 分起 至 93 年 5 月 19 日 16 時 30 分止	地 點	科技大樓 605 室
主 持 人	交通大學蔡淳仁教授	記 錄	電通所焦信達
出 席 人 員	35 家廠商 60 人次		
報告、討論事項及結論			
一、主席致詞(蔡淳仁教授-略)			
二、引言： 1、學術界對數位電視產業發展的看法(陳永昌教授-略) 2、MPEG Video Coding 標準的最新發展(杭學鳴教授-略) 3、DVB/MHP 的最新發展(郭天穎教授-略) 4、工研院在數位視訊領域技術發展的佈局(王鴻智執祕-略) 5、由歐盟 TIRAMISU 計畫談數位電視的內容保護與管理(高億憲博士-略)			
三、Coffee Break			
四、台灣數位電視產業聯盟及技術發展工作小組簡介(略)			
五、座談討論			
決議事項： 1. 民視目前已經完成了宜蘭竹子坑以及南投等地的 SFN 測試環境。涵蓋範圍超過 100 公里，已經超過德國所作的類似實驗的規模，希望能在模擬驗證與實際檢測方面與學界合作，以作為台灣發展 SFN 實際的 deployment 前的參考。台大陳宏銘教授表示，台大呂學士教授是這方面問題的專家，聯盟技術發展工作小組將邀請呂教授就 SFN 做專題演講。 2. 民視侯銘罡組長提到在 DVB 網站上註冊 Network ID 不方便的問題，技術發展工作小組將請高億憲博士代為向 DVB 的執行秘書長 Peter 反應這個問題。			

會議記錄(續頁)

第 2/3 頁

3. 民視希望能知道有關 Tiramitsu 計劃在無線電視台數位磁帶加密技術上的規劃。

4. 技術發展工作小組依現場業界的要求，未來將舉辦一個關於 H.264/MPEG4 AVC 的 workshop，並邀請國內外相關業者與專家與會。

5. 中央蔡宗漢教授希望技術發展工作小組能舉辦學界 DTV 相關技術發表的 workshop。

Q & A:

1. 溫普敦資訊林宗源總監:

Q: 如何讓台灣的 solution 能散播至大中華市場？

A (王鴻智經理): DTV 國際交流小組會負責與大陸進行 DTV 相關標準的交流

Q: 台灣與大陸只少有六種不同的中文碼
如何解決兩岸中文碼互通的問題

A (王鴻智經理): 工研院電通所目前已經與交通部進行相關的研究計畫，以處理兩岸中文編碼及字型問題

2. 民視侯銘罡組長:

Q. HDTV 是不是能用 H.264 的 decoder？

A:

蔡淳仁教授: 使用 H.264 可能會有 license fee 的問題，雖然大多數的歐美業者希望能採用 baseline 免費或是較單純的收費法則，但也有些業者如 SONY 不介意要收費（原因之一是他們在 MPEG LA 裡掌握不少的股份），所以 H.264 在日本市場可能會比較早被採用。而目前 H.264 收費模式尤其對歐洲的數位廣播業者不利。

會議記錄(續頁)

第 3/3 頁

杭學鳴教授: Nokia 在手機上已經有 software-based solution

蔡淳仁教授: 已經有不少相關的學術性論文指出 H.264 decoder 的複雜度大約是 MPEG-4 SP 的三倍。而在 100 MHz 的 ARM 9 上 QCIF 解析度大約可以有十幾個 frames per second 的效能。

3. 台灣寬頻技術副總 Heniz:

Q. 工研院電通所高博士所提出的 TIRAMISU 計畫基本上是歐盟採用的 DRM 技術，但 content 主要的提供者是美國，為什麼不採用美國標準的 DRM？

A. 高億憲博士: 工研院目前 focus 在歐盟的 DRM 標準，未來會研究美國標準的 DRM 技術，而且美國的 content 會輸出到歐洲，屆時一定也會用到歐盟的 DRM 標準，所以台灣採用 TIRAMISU，並不會有缺乏美國方面 content 的問題。

蔡淳仁教授: Content 和 DRM 是分開的，所以不需要太擔心這個問題。

議題：

- 技術小組未來工作方向討論
- 技術小組未來會議主題
- 業界及學界對技術小組的期待和需求
- 技術小組如何促進業界、學界合作進行 DTV 相關技術開發



會議名稱	Network ID 註冊機制的討論－日本 Data Component ID 的做法		
時 間	自 93 年 6 月 1 日 13 時 00 分起 至 93 年 6 月 1 日 14 時 30 分止	地 點	11-251
主 持 人	王鴻智	記 錄	林育仁
出 席 人	駿昇科技：劉錦銘總裁 吳秀珊副總裁 藍偉明副理 CCL：王鴻智經理 梁修淵 鄭翰聰 黃淑敏 林育仁 邱志偉 龐德俊		
報告、討論事項及結論			
1.Gemstar 介紹 Data Component ID 之定義與應用 (a).Data_Component_ID 由可交通部來 assign (b). 每家公司可為自家的應用服務 (如：股票資訊) 註冊登記一個 Data_Component_ID			
2.Gemstar 介紹 Japan ARIB - Data Component ID 之規範定義 (a).SI Table->Descriptors->Data_Component_ID (b).可與 Country_Code 搭配使用,以區分各國具有相同 descriptor_tag 之 descriptor 的定義			
3.ARIB Spec. - (Data Component ID) (a).EIT table, Data_Contents_Descriptor, Tag value 0xC7 (此 Descriptor 為 Optional) (b).PMT table, Data_Component_Descriptor, Tag value 0xFD (此 Descriptor 為 mandatory)			

會議記錄(續頁)

第 2/2 頁

4.Example:

PMT(PID=0x40) Gemstar PMT

Type	PID	stream type	component tag	component ID
Data stream 1	0x200	0x0D	0x52	0xA
Data stream 2	0x201	0x0D	0x53	0xA

5.Gemstar 建議台灣採用

(a).EIT table, Data_Contents_Descriptor, Tag value 0xC7

(b).PMT table, Data_Component_Descriptor, Tag value 0xFD

6. 若台灣採用 Gemstar 的建議時 ,Gemstar 希望能先為自己註冊一個 Data_Component_ID=0x0A

『台灣數位電視產業聯盟』---標準制定工作小組

- 台灣數位電視產業技術規範研討 -

會 議 記 錄

指導單位：經濟部工業局、技術處、交通部電信總局數位電視共通平台技術諮詢委員會

主辦單位：經濟部數位視訊工業發展推動小組、台灣數位電視產業聯盟、工研院電通所

協辦單位：台灣數位電視委員會、民間全民電視公司、台灣區電機電子工業同業公會

會議名稱		台灣數位電視產業技術規範研討	
時間：九十三年六月十一日（星期五）上午 9:30 ~ 11:30		地點：台灣區電機電子工業同業公會七樓第二會議室	
主席：石佳相博士（中環公司）、陳宏銘教授（台灣大學）		記錄：徐國書 執秘	
出席人員		數位電視產業相關廠商 15 家暨廠商 30 人次	
會議大綱暨議程：			
引言：數位電視產業標準制定			
主持人：DTV Taiwan 標準制定工作小組產業召集人 石佳相博士			
產業規範說明：台灣地面數位電視接收機技術需求(草案)			
主持人：工研院電通所 高億憲 博士			
產業規範說明：數位電視系統服務之網路識別規範(Network ID)			
主持人：民間全民電視公司 侯銘罡 組長			
產業規範說明：數位電視 MHP 接收機系統技術規範(DVB-Java 部份)			
主持人：工研院電通所 蕭毓欣先生			
休息			
產業標準規範研議座談			
主持人：DTV Taiwan 標準制定工作小組產業召集人 石佳相博士			
DTV Taiwan 標準制定工作小組學研召集人 陳宏銘教授			
會議報告、討論事項、內容：			
一、主席致詞：引言 - 數位電視產業標準制定工作會議 (石佳相博士-略)			
二、產業規範說明:			
1、台灣地面數位電視接收機技術需求(高億憲博士-略)			
2、數位電視系統服務之網路識別規範- Network ID (侯銘罡組長-略)			
三、Coffee Break			
3、數位電視 MHP 接收機系統技術規範-DVB-Java 部份(蕭毓欣先生-略)			
四、產業標準規範研議座談討論			
● 提案討論：台灣區數位電視產業標準制定程序			
1、主席石佳相博士說明【台灣區數位電視產業標準制定程序】研議目的，此項議題以協助及制定數位電視相關標準之規範為主。未來本項程序規範書通過後，將成為台灣數位電視產業聯盟標準制定之程序規範依據。			

2、此項標準制定程序主要的規範內容，例舉如下：

- ◆ 標準制定組織架構及參加辦法
- ◆ 標準規範提案方式及管理辦法
- ◆ 標準制定及審查委員資格認定
- ◆ 標準規範發佈程序及主管單位
 - 例如 交通部、經濟部、新聞局、公學會
- ◆ 標準規範內容引用範圍及參考來源
 - 例如 DVB、DigiTAG、ATSC
- ◆ 國內相關組織互動模式及標準引用辦法
 - 例如 電視學會、台灣數位電視委員會、經濟部標準局

3、本項工作之進行以成立一 Task Force 之「程序規範書工作小組」負責；小組成員以公開方式由廠商專家自願性參加為原則。

4、成員的代表性希望可以 Cover 各個領域。

五、會議記錄綜合摘要(詳細討論內容，請參閱備註 Q&A 部份)

針對今日之標準規範說明及議題討論，整理成如下幾點；

就業者及廠商所關心的議題部份：

- 1、Network ID 或 SI 的 PID 規範，標準應該考慮整個業界的範圍，如有線、衛星與 Content 業者，而不是只有無線台的部份而已。
- 2、MHP 的技術規範標準制訂應儘量與世界接軌，不應該自行關起門來訂一套只有台灣才會用的標準，如此才可使廠商可以開發國外的市場及機會，廠商也比較願意 Follow。
- 3、中文編碼的問題應明確的訂定規則，是 Unicode UTF-8 還是 BIG5、CNS？如何使用在 Content 的製作及 MHP Service Application 的應用等。這部份不但要考慮國內的情形，更應該考慮兩岸 Content 的互通。
- 4、MHP 技術規範若是只以 Linux based 為 OS 基礎，不足以滿足業界的需求，也不符合業界實務的情形，業者針對不同的作業系統版本將不知如何開發對應的軟體技術，驅動程式的撰寫也增加許多版本上的困擾，建議應將其他的 RTOS 的 Solutions 考慮進來。
- 5、MHP 的 STB 平台採用 Java based 的 JVM 基礎，在執行效率、程式撰寫及 Content 應用會有一定程度的困難，電通所就這部份應該有完整的輔助工具或技術提供業界使用。
- 6、MHP 技術規範在 Content 的格式部份應該不用限制太多，並且要有彈性擴充的空間以符合未來的新的格式，例如 MPEG-4 AVC、H.264。
- 7、MHP 的標準制訂應該從消費者的角度來考量，不要太偏技術面的規範，要從市場面及應用面來制訂，可能會較符合市場的預期。
- 8、聯盟標準小組的任務，以及對業界參加產業標準制訂的期望，希望能夠明確；而且在小組在制訂標的態度上應該廣納業者的意見，並做為與政府相關決策單位的連絡管道，反應業者的心聲，業者也才會有較正面的貢獻產生。
- 9、數位電視產業標準的制訂意義在那裡？未來這些產生的標準與全球市場及世界其他標準的關連性為何？與兩岸的標準是否可以共通？種種的問題都是台灣數位電視產業聯盟必須積極思考的，否則一旦數位電視產業快速成長時，業者若無所適從，將喪失許多市場機會，更可能無法與世界同步接軌。

就聯盟標準小組與工研院電通所可能解決的方案部份：

- 1、MHP 技術規範的標準制訂，將以 MHP 共訂規格所形成的共識來擬訂，原則上是以符合 DVB MHP 規範之最基礎的部份為主，也就是 Minimum Requirements，而非 Mandatory Requirements 為出發點。
- 2、工研院電通所將配合標準的制訂，提出相關的技術開發 Reference Design 與輔助工具，協助業界盡快進入數位電視的產業。相關的技術開發議題，原則上在聯盟的技術發展小組的部份開會討論。若業者有急迫的需求，可以提案在技術發展小組專題討論解決。
- 3、現階段標準小組的工作以完成「標準制訂程序規範書」、「MHP 技術規範書」以及中文編碼相關問題為優先，其他有關 MHP Service 及 Application 的應用形態(Browser、Email)的部份，會留在制訂「MHP 應用與服務規範書」時討論。
- 4、聯盟標準小組運作的方式，將以蒐集業者廠商的意見想法出發，形成業界共識來推動數位電視產業。未來也許會有許多地方須要廠商提供協助或解決方案的部份，希望業者也可以共同參與制訂標準，做出實質的貢獻。這部份的機制會以聯盟的「標準制訂程序規範書」為依據。

決議事項：

- 1、主席石佳相博士就【研議台灣區數位電視產業標準制定程序】公開徵求廠商與專家擔任「程序規範書工作小組」成員
 - 「程序規範書工作小組」成員原則上以 5 至 6 位成員即可，完成程序書的時約需一個月
 - 標準制定小組產業、學研召集人及執秘為工作小組之當然成員
 - 現場調查其他自願擔任工作小組成員之廠商暨專家
 - i. 遠昇科技市場策略專案經理 官振鵬博士 C.Kuan@adbgloba.com (02) 89131500 ext 564
 - ii. 東森媒體科技技術應用處 高正一協理 kao.riker@etmc.com.tw (02) 55528320 ext 1368
 - iii. 溫普敦資訊股份有限公司 林宗源技術總監 JimmyLin@WinPerTurn.com (02) 26550317
- 2、「程序書工作小組」希望在近期內成立，並在一個月內完成【台灣區數位電視產業標準制定程序規範書】，相關工作小組成員有責任及義務協助完成該項任務。

臨時動議：

無

備註：

Q & A:

1. 有關 PID 及 Network ID 標準規範說明部份，目前五家無線電視台之情況分為 Content 製作及播放平台兩方面；PID 的部份在 Content 製作時可以自行設定。另外在 Network ID 部份，台灣的產業不是只有五家無線台，若以 DVB-SI for Taiwan 為標準規範題目，似乎沒有兼顧 Satellite 及 Cable 部份的情況，以及其他 Content 製作商的需求及考量。所以這個部份應該以 Content 製作及播放平台兩個層次來看，並兼顧產業目前的現況才較合理！
2. (侯組長) 很同意這位先生的說法；這部份的說明是以無線台為主，以 SI for Taiwan 稱呼的確不適合，將趕快修正。另外在 Network ID 部份，除了 Original Network ID 不可重複，其他的 Network ID 是可以重複，Satellite 及 Cable 業者要用是沒有問題的。目前我們請電信總局協助申請的 Original Network ID 也是只有地面廣播部份，Satellite 及 Cable 業者也是可以自行申請的。至於 Content Provider 交給無線台的 Content，目前試播之五家無線台協議會統一 ReMUX 轉成上述說明的 PID。
3. 有關 MHP Application，一般來說有 Xlet 及 HTML-Based 兩種，而且 Xlet 這類程式是 Java Application，國內會撰寫這種程式的人可能不多，請問電通所在這方面有什麼輔助的方法與規劃？
4. (蕭毓欣)針對 HTML-Based 的 Application，DVB-MHP Internet Profile 已經詳細定義 WWW 及 Email Client，電通所打算開發設計 Web Browser 來提供這類的應用。另外在 Xlet 的撰寫部份，電通所也在開發 Authoring Tool，以 Visual Editor 的方式協助 Content 的製作者在不須要懂 Java 程式的情況下開發 Xlet Application 並自動 Generate 這類程式。
5. 有關 MHP 部份的 Windowing System 採用 Linux Based 方式，是否以後的 STB 都將採用 Linux 系統，其他的

- 系統如 Windows Based 系統是否也會考慮？而且貴所也在開發 Web Browser 及 Email Client，這類程式有很多是 Java 的 Applet，若你們的 STB Environment 是以 Java based 為主可能會有許多問題，還是你們會有 Dual OS 系統的規劃？
6. (蕭毓欣)其實 STB 的環境與 PC 是類似的，理論上在 PC 可以執行的程式及可以達成的功能，理論上也應該在 STB 上也可以執行與完成，沒有太大的差異，唯一的差異是 STB 的硬體資源有限，許多功能可能無法完成。
 7. 若貴所小組制定的標準是以 Linux based 為主，現在業界考慮的重點是 Content 可否在此 STB 上執行，以及關於兩岸中文處理的 Solution，是不是你們已經了解大陸方面的做法？而且台灣的市場太小，你們在定義這個部份，應該考慮的是市場面而非政治面！
 8. (電通所王經理)這部份並沒有政治的考量，現在的了解是中國在這部份是以 Linux based 在進行，而且 STB 這部份也是 OS Independent。我們的了解市場也傾向以 Linux based，MHP Spec 也沒有這樣的要求，這不是一項 Requirement，最後還是須以市場的需求為主。
 9. 另外一個比較有興趣的部份是 WWW Browser 及 Email Client，不知這部份標準的 Rough Spec 在何時會制訂出來？
 10. (蕭毓欣)時間可能目前無法決定，不過電通所就這部份已經著手在研究及進行。
 11. (徐國書執秘)針對 WWW、Email Client 及 Internet Service，這部份屬於較上層的 Application 範圍，與 MHP 技術規範內容上有點差距，目前可能沒有答案。不過針對這部份，聯盟標準小組會在 MHP 技術規範制訂告一段落後或今年後半年，安排在 MHP 應用服務規範制訂的部份來討論。
 12. 關於中文碼部份，大家都知道兩岸的中文編碼及使用情形，有 GB、Unicode UTF-8、CNS、BIG5... 零零總總也有 6 種之多。中文碼的部份在 Leading code 採用 0x14 等，或是還有其他的中文標準規範應該趕快釐清，這個標準必須趕快完成才會有商機。
 13. (徐國書執秘)就中文標準部份補充說明，當初在提案中文 Leading code 時是 0x13，但是這個碼已經由 NDS 及 Nokia 寫了兩份 Proposal 給 DVB 幫大陸提案，因此電通所才改提案 0x14。0x14 Leading code 的編碼實際上是表示後面的碼是 Unicode BIG5 Subset 的部份，編碼實際上還是 Unicode。Content Provider 在製作時還是可以自行決定是否要用 0x14，並沒有強制要使用 0x14；而且當初提案這個編碼是考量 STB 製造商的成本需求，希望在 STB 可以使用較便宜的 BIG5 Font Chip，而不是一定要用 Full Set 的 Unicode Font Chip。
 14. 請問在你們的 MHP 架構，為何在 JVM 下面有一個 Window System，它的功能到底是什麼？這部份是你們制訂的，還是 MHP 規範就是這樣？如果 MHP 標準已經定義好了，為何電通所還要再做整理？另外，若是需要一個 Web Browser，這是一個 Java Based 的 Application，那它的 Performance 如何？
 15. (蕭毓欣)這部份 Window System 主要是一個泛稱；我們電通所整理出來覺得是應該長的像這個樣子，因為 MHP 需要一個 Window System，它不是專指 Microsoft Windows System，而是一個獨立的 Window System。而且這個部份只有 Basic 部份，MHP 應該還有其他的部份我們還沒放上來。另外 Java based Browser 部份我們還沒做任何的測試。
 16. 這樣一個 MHP 標準只有牽涉基本的部份，是否與 e 化 Taiwan 有沒有關連性？而且台灣的 MHP 的市場可能不大，這樣一個不完全的標準，到底要把台灣的數位電視產業帶到那裡去呢？有沒有一明確的方向呢？
 17. (電通所王經理)電通所提出的 MHP 技術補充說明，目前電通所要制訂的規範是台灣的 MHP 標準，是考慮台灣的產業所需要的功能及需求為主。若是廠商或業者有國際市場的需求，他必 Follow 國際的 MHP 標準。目前這個 MHP 技術標準，是依據去年國內 MHP 共定規格聯盟的需求，以及解決中文問題來訂定的，可能暫時沒有考慮到與國際市場接軌的問題。另外，現階段國內正在進行試播，馬上需要一共通性的 MHP 規範及測試環境來發送及接收 MHP Stream，然而 STB 廠商到底能接收多基本的 MHP Stream 及功能，這個標準就是在定義這些最基本的功能。至於電通所在開發 MHP 的技術與制定 MHP 標準是兩回事，制定的 MHP 標準可能電通所還沒有開發完成，但是後續還是會配合開發特定的技術給業界當作 Reference Design。
 18. 有幾個意見，1、有關 MHP Audio/Video 規格，若是在 2006 制訂成為國家標準，必須有一更進步的觀念以接受更新的 A/V 規格才行！。2、目前制訂的這個 MHP 標準，好像不是國際的標準，又不是國家的無線廣播標準，其中有關數位內容的標準部份，國內的有線或中華電信 MOD 都是 Transport Stream，不知道這部份以後是不是可以播放的出來？3、目前這個標準是 Java based，播放 Stream 時需要 JVM 的處理，若切換頻道需要 9~10 秒，消費者可能沒有辦法接受。所以這個 Java based 的標準，速度會是一個很大的問題！4、這個 MHP 的架構是 Linux based，若是要符合無線廣播需要一個 Middleware 的架構，目前這個架構是沒有，而且正常的情形也應該考慮其他的 RTOS 需求，如 WinCE。再者這裡又需要一個 JMF，JMF 這麼大，誰負責來做？何況業界 Java 的技術有兩個版本，一是 Sun 的 Java SunONE 及 Microsoft 的 .NET，到底該採用那種 Solution 呢？5、另外 STB 又有一個 CA 的版本，而 CA 又分成 Firmware 層次、Middleware 層次及上層 Xlet 的版本。這麼多的版本，若是屆時 2006 成為國家標準或業界標準，到底我們硬體廠商或 Middleware 廠商該怎麼 Follow 或是標準該怎麼訂，又是一個大問題！
 19. (電通所王經理)關於 Java 速度較慢的問題，電通所可以配合硬體廠商來開發相關的技術，目前國內也有晶片廠商開發晶片來解決及改善這方面的問題。另外 JMF 在 RTOS 的方案選擇，廠商還是可以自由選擇希望使用的

OS Solution，工研院的作法是至少提供一個 Solution 給廠商做參考。至於 License 的問題應該不會有太大的困難，因為 MHP 與 Sun 之間有達成一個協議，可以將整個 License 費用降得很低，國內廠商應該與國外的廠商 License 費用沒有太大的差異。

20. 另外，你們的標準在 Return Channel 的定義太模糊，到底你的 Return Channel 用來做網路系統的連接還是做系統回應？它的連線方式是走 GSM GPRS、PSTN 還是 PPPoE？而且系統是 Linux 的話，那它的 Driver 如何寫？若改成 WinCE 的 Java，版本又不同了。
21. (電通所王經理)有關 Return Channel 的部份，這裡並不會規定要採用那一種 Return Channel，電通所的目標還是只 Focus 在 Middleware 的部份，並提供一份 Reference Design。若是對 Implement 有需求，聯盟的技術開發小組不定期也會討論相關的技術，提供業者相關的 Solution 情況，或者業界廠商也可以組成研發聯盟的方式，共同開發相關的技術。另外才剛才提到的 A/V Format 的問題，是不是可以支援較新的格式，例如 MPEG-4 AVC、H.264？我想這部份可以 Option 的方式在標準中規定，主要還是以 DVB 的 Main Profile 為準，若是廠商有其他的需求，電通所在這部份還是站在中立的立場，配合業界的需求來制訂 MHP 的標準。
22. 外籍人士 Dennis 發問(原文-略)，主席石博士轉述。我們在這裡的標準討論不需要太偏技術面，而應該從消費者的角度來看。消費者可能對畫質的要求比介面方面的標準更有興趣，譬如說這部份可能就要將 HD 列入考慮。
23. 就聯盟標準小組所制訂出來的標準，將來所有的 Venders 要銷售產品到台灣來，是不是要受這個標準的規範或者通過這份標準的測試及檢驗？若是答案很明確的話，業者是要等標準完成，以及要等多久。若是要等很久，業者最後可能連商機都沒有了！而且最重要的是最後總是要有一個單位出來負責，說明標準及規範給業者的遊戲規則為何，使業者可明確的 Follow，而不是以論壇的方式討論討論而已，最後還是沒有答案。
24. (主席石博士)這個標準的問題可能不是一時之間可以解決的，畢竟無線、有線業者可能採取不同的作法。我們希望在聯盟標準小組這裡大家來討論及共同制訂這樣需求的標準。並且我們也嘗試說服政府對標準採取一種較 Open 的態度，讓業界的標準先走，實際的在市場上運作。等到市場較為成熟之後，政府再出面主導整個相關規範成為國家標準。
25. 針對聯盟標準工作小組的 Mission 是什麼，是不是有一個很明確的策略。是不是做為業者與政府溝通的管道，讓業者的聲音可以透過這個管道讓政府了解。最重要的是每次的會議議題以及業者的 Concerns，應該要明確的記錄下來做為以後執行的依據，否則下次就越來越少人會參加了。另外，標準小組的態度要能 Open，得接納各種不同的意見。最終還是需要有一機制，讓業者表達他的看法給政府，標準小組又是如何希望業者來參加這樣會議的目的與期待在那裡，希望我們提出何種的需求想法或建議，是不是可就這方面了解說明一下。
26. (主席石博士)聯盟標準小組並不是代表政府出面來說明標準的制訂問題，基本上我們的態度是很 Open 的。數位電視產業聯盟是工業局所主導成立的一個業界聯盟，其中設立有八個小組，主要就是希望透過這些小組的運作，蒐集業界的意見形成共識，使得數位電視的產業能快速成長。實際上，標準小組的立場是與業者站在一起的。最後今天的提案就是希望成立一個這樣的標準制訂程序規範，來滿足大家的需求，使業者的意見想法可以記錄起來，相關 Document 也會記錄起來。等到比較完整的時候，再向相關的主管單位提出申請，如標準局、電信總局，形成國家標準。
27. 若是聯盟標準小組的精神是要扶持國內的業者，這個標準的制定要儘量能夠與世界的標準同步或接軌。並且，這個標準不要訂定太多的細節把自己限制了，比較有彈性的方式，我想會有比較多的業者願意 Follow。
28. (主席石博士)這部份我想業界相當 Comprised 的，就算 DVB 所制訂的標準涵蓋面也是相當廣泛。台灣有台灣產業特殊的地方，不管在工程及技術的部份，有線、無線及衛星的作法皆必須兼顧，台灣也沒有足夠的大的市場支持一個完整的新規格，最終還是須採取國際的標準為基礎。



會議名稱	無線數位電視多媒體共通平台技術規範研討會		
時 間	自 93 年 8 月 20 日 10 時 00 分起 至 93 年 8 月 20 日 12 時 00 分止	地 點	電信總局 8F 會議室（臺北市 延平南路 143 號）
主 持 人	王鴻智	記 錄	梁修淵
出 席 人	電信總局：陳子聖、謝慧靜；華視：徐俊方；台視：林瑞榮、林訓 民視：侯銘罡；中環：石佳相；工研院電通所：王鴻智、焦信達、 梁修淵 （敬稱略）		
報告、討論事項及結論			
1. 無線數位電視多媒體共通平台技術規範（草案）報告：焦信達（詳附件）			
2. 建議將組態層次 0，位階 0 之中有關字幕（中文）及資料廣播服務列為層次 0， 位階 1 之功能；原則上，並將層次 0，位階 0 之採用時程提前自 94 年 7 月 1 日起實施，確定日期將由電視學會工研會於 8/27（五）討論訂定後，於 8/31 （二）「數位電視多媒體共通平台技術標準諮詢委員會」會議中提出。			
3. 有關公用頻道（Engineering Channel）之規劃，請電視學會工研會擬定初步 Proposal 提交通部討論。			
（以下空白）			



會議名稱		「數位廣播電視系統整合之研究」座談會	
時間：九十三年十一月十六日（星期二）下午 14:00～16:30		地點：臺北市延平南路 143 號 10 樓會議室 （電信總局廣電技術處）	
主席：王鴻智博士		記錄：黃淑敏	
出席人員	台灣數位電視產業及有線電視相關廠商 9 家暨廠商 20 人次		
會議大綱暨議程：			
引言：數位廣播電視系統整合之研究 主持人：王鴻智博士 技術規範說明：數位電視多媒體共通平台技術規範 主持人：工研院電通所 焦信達博士			
會議報告、討論事項、內容：			
一、主席致詞：引言 - 交通部電信總局委託工研院電通所研究（王鴻智博士-略）			
二、產業規範說明： 1、數位電視多媒體共通平台技術規範(焦信達博士-略)			
三、會議記錄綜合摘要			
針對今日之技術規範說明及議題討論，整理成如下幾點：			
就台灣有線寬頻產業協會提出五大議題(請參閱備註)：			
1、就第一點技術規範內容的前瞻性，是否能滿足未來國內有線電視技術需求，以及是否有競爭力，而不會造成將來技術發展的阻力。			
王鴻智博士：關於技術的前瞻性，這問題確實非常重要，訂定一個規範，必須考量到未來，因為 MPEG4 目前僅是一個潮流，還不是一個標準，一個技術規範是希望成為一個標準，必須是很確定的，而且 MPEG4 有非常多的 profile，不同產業所使用的 profile 都不一樣，這部分還沒有很確定穩定的時候，我們不建議把它放在標準裡面。			
石佳相博士：關於前瞻性的問題，這個基本上我們並不是想要定好一個強制的規範，使得一個市場沒辦法往前發展，所以我們分時程、分層次、分階段，之所以採用 MHP 也是這樣的考量，就是說考量到整個歐盟市場上，在多媒體的環境上 MHP 是主流，在整個發展上也考量到整個技術進展，所以我們是以推薦 MHP 的方式來做這樣的規範。			
焦信達博士：關於 MPEG2 方面，因為在 DVB 裡所使用的傳輸標準有開始採用新的 Audio Video coding 方式的趨勢，基本上這個部分現在還沒有被定義到 DVB 相關的 spec 裡面，所以基本上我們處理將來 Audio Video coding 變化的部分就跟共同規範的第 2.4 點一樣，當有新的影音串流標準出現的時候，本規範依據 DVB 相關的新標準定案之後再來增訂。			
2、有線電視的競爭者為電信業者的數位多媒體平台，能否相對比較電信相關法令的內容，以免造成有線電視業者在技術上的不公平競爭。			
石佳相博士：電信相關的法令，涉及到兩個產業競爭的問題，在歐盟他們強調的是 relation，並不是強調 zero sum 這種零合的競爭方式，所以歐盟希望整體的產業包括不同產業不同層次			

或者不同技術之間能夠像演奏一場交響樂一樣，大家能夠共存才可以，所以在選擇這些規範技術的時候，基本上我們是參考歐盟這種思考方式，原則上參考的規範以電信總局來說，技術中立，由業者決定，而衛星上已經是歐規 DVB-S，有線是 DVB-C，無線是 DVB-T，所以這部分大概是沒什麼好選擇的，一定就是 follow 歐洲 DVB 的部分來做選擇定義。

陳子聖簡任技正：關於電信是否有相同的規範需等未來訂定出來，自然將來有人來做這個產業，也要受共通的規範，這是一個大原則，這個在管制的原則是相當一致的，事實上我們也面臨了一些問題，無線跟有線部分，有線必載無線的這部分，也必須注意到將來他們的共通性，所以今天同時提出無線有線衛星的規範，是希望將來對業者之間也能達成一個互通互容的目標，以避免未來大家在競合上面出現一些障礙，也希望能藉由各位先進達成一個共識，使得大家能順利的營運。

3、世界各國相關規範參考及說明。

焦信達博士：關於世界各國規範參考說明，其實這部分我們在訂定的時候有參考了很多規範，例如芬蘭 Cable operator 的規範，參考他們的一些想法，也參考了 DTT 數位無線電視台，他們是一個 Network operator，同時並參考了他們的共通規範，另外還有荷蘭、義大利、澳洲...etc。

4、技術規範的條文內容及說明。

石佳相博士：首先解釋一下擬這個草案的基本精神，原則上這是針對 open market，凡是屬於 private system 尤其像是 cable 系統的東西，我們並不建議由這種草案的方式強制執行，所以這個草案所有談的是包括上下游頭端終端設備整體的協調，因此我們必須制定這樣的規範。

王鴻智博士：關於其它相關條文內容說明，我們做了一些 research 也有做一些分析，在此必須強調我們要找的共通平台還是以開放式，我們沒有辦法找一個 private solution 來去建議採用某一家公司，這樣對於國內上中下游的產業可能會有所偏頗，所以我們採用一個開放式的平台，另一方面我們採用的都是歐規 DVB 的標準，目前看來 MHP 是比較適合的選擇，之前石博士焦博士也一再強調這是一個推薦，並不是一個強制，至於推薦敘述的細節以及具體做法，還是需要各位先進提供寶貴的意見。

5、現有業者系統技術相容及轉移問題。

石佳相博士：其實很早在談這個問題的時候就 concern 到有些有線業者已經採用某一套 CA 及某一套 Middleware，原則上採用兩個精神，一個是推薦的方式，另一個是我們去制定這些規範主要針對 open market，就是由消費者自行採購去擁有這個終端設備，在這樣的一個市場上面，為了保護消費者與上下游的關係，因此勢必需要制定某些規範，而制定這些規範呢，必須盡量要跟 private 的系統能相容，所以在整個制定規範的考量之下，我們希望這個規範能夠跟既有的有線電視採用的系統相容共通，能夠並存在這個市場中，至於有線系統將來怎麼走向，其實應該有相當大的一個空間，並不需要強制去做系統的轉移，而仍然能有一個並存的空間存在，這就是 CA 在 MHP 的這些規範制定上所做的一些考量。

就業者提出其它相關議題部分：

1、針對劉兆悅博士提出有線系統是一個內網，不需規範來設限。

石佳相博士：第一個，基本上這不是強制性的規範，所以不用太緊張，第二個，之所以分層次，當初的考量，當然有線是一個內網沒有問題，但是當涉及到節目及終端家電商的時候，就不是單純內網問題了，因為你必須要求用戶端的設備要能夠解這些節目，要讓這些製作服務應用的廠商能夠在你的終端設備能被解，這不是一個內網的問題，應該是一個市場秩序的問題。

王鴻智博士：其實我們主要是針對如果將來組合無線有線衛星都會使用到一個共同規範多媒體共通平台的時候，那這一個平台這一個技術相關的技術規範應該是怎麼樣的，這是我們根據這個主軸來推薦出來的一個技術資料，整個草案絕對不是只是一個技術肯定還要考量到其它因素，今天這個技術草案只是一個草案，針對技術部分，並不是一個法律條文。

2、針對高正一協理提出相關查驗問題。

石佳相博士：如果是 private 系統當然是不需要查驗，但如果要採用這種 open market 的做法，則不只是頭端要提供查驗，連所有的家電商的 STB，數位電視機都需要查驗，否則無法確認能夠互相的相通，這是一個必備的過程，只要系統上的 operation 有採用這種 open market 的做法，勢必一定要有相當的查驗的過程，不然上下游就無法相通。

陳子聖簡任技正：提出這些規範，完全不是站在限制各位的經營，或是不管從技術面營運面，絕對不是在限制大家，而是希望說要找到一個共通平台，這個共通平台能讓業者有獲得最大利益，消費者也獲得最大利益，基本方向是沒有問題的，至於查驗部分，我們會再另外提出一份 proposal，這個部分其實是希望將來上中下游能有共通性的最大好處，這個查驗跟當初有線電視門檻，用意是不一樣的，這裡是希望能夠將來讓數位視訊產業不管有線無線都有一個共通性。

3、針對高正一協理提出 CA、Euro Loader 相關問題。

石佳相博士：

CA 的部分不是強調一個共同的 CA，只是一個共通的，能夠相通的介面叫做 simulcrypt，是一個介面而不是一個共同的 CA，就未來能夠相通之後，各公司仍然可用各公司的 Nagra、NDS 等等，都沒問題，只是有一個 simulcrypt 能夠通的介面，使節目可以跨系統的轉換，用戶端的設備也能跨系統的轉換。

loader 的問題，一樣這是一個 recommendation，不是一個 mandatory，尤其對於 cable，如果是一個 private 的系統的話，可以不用 follow 這整本的規範，之所以要定 loader 是針對一個 open market，換句話說，消費者今天買一個數位電視機 build-in 有中嘉系統的這樣的一個數位電視機，有一天要搬家到東森的系統裡面去的時候怎麼辦，勢必一定要有一個公開標準化的 loader 去作終端設備的變換。

在這有一個建議，我們增列第一條，去說明這個規範適用的方式，是在一個開放市場的狀況之下，不會去限制到有線跟衛星私領域的部分，希望把這一條增列上去，則之前很多的問題及疑慮是不存在的。

4、針對高正一協理提出“把學理上的明日之星(MHP)看作市場的明日之星”。

王鴻智博士：這個研究案主要是剛剛陳簡任技正有特別提到一月二號已經有要求工研院電通所來研究 MHP 未來，及可能是一個軟式導入，或是有線無線等等需要一個共通平台，在已確定是 MHP 的情況下，其技術規範內容應該為何？這是研究案的目的，當然本人很贊成高協理的意見，確實很多技術上非常好的東西，不見得市場就能接受，這可能是定行政策等等我們才要去考量的。

備註：

台灣有線寬頻產業協會提出五大議題內容：

一、 技術規範內容的前瞻性，是否能滿足未來國內有線電視技術需求，以及是否有競爭力，而不會造成將來技術發展的阻力。

由於全世界數位電視的技術相當多，在不同的應用上，各有不同的優勢，也沒有任何一項技術是公認的標準，因此，對於制定國內『有線數位電視多媒體共通平臺技術規範』內容，應該先讓業者了解世界各國有線電視數位電視的技術發展現況，比較各種技術發展優勢，進而再決定國內制定規範的涵蓋範圍是否具有技術的前瞻性。

二、 有線電視的競爭者為電信業者的數位多媒體平台，能否相對比較電信相關法令的內容，以免造成有線電視業者在技術上的不公平競爭。

有線電視與無線電視發展平台不同，政府在制定有線數位電視的規範時不應把對無線數位電視的規範，直接從「無線」，文字修正成「有線」，然後就對有線電視做規範。同時政府在產業公平發展的角色上，不應只針對有線電視作做技術規範，而電信業者確無相對的技術法規，這將會形成不合理的市場競爭，要求能有相同的法令標準，使有線電視與電信業者應享有公平對待的原則。

三、 世界各國相關規範參考及說明。

有關『有線數位電視多媒體共通平臺技術規範（草案）』所涵蓋的技術範圍主要可分為 4 類，第一類為頭端與傳輸規範 (Headend & Transmission)，第二類為應用服務平台 (Application Platform)，第三類為加密系統 (Conditional Access)，第四類為機上盒軟體更新機置 (STB firmware-upgrade mechanism)，內容相當廣範。是否能在規範制定之前，先詳細提出其它各國政府對相關技術法令制訂的參考研究報告，再來決定國內的技術規範條文。

四、 技術規範的條文內容及說明。

全世界在這數位技術所使用的標準相當多，例如 Headend & Transmission 在有線方面就有 ARIB, ATSC, 以及 DVB-C 等；而 Application Platform 有 BetaNova, Liberate, MHP, MS TV, NDS, OpenTV, Wink 等；Conditional Access 則有 BetaNova, Irdeto, NagraVision, NDS, Viaccess 等；STB firmware-upgrade mechanism 而各家 DTV 經營者又有其獨特平台管理方式。因此，建議能針對現有草案各條文內容，明確說明制訂的原因及技術等考量因素。

五、 現有業者系統技術相容及轉移問題。

目前國內各家有線電視業者的數位電視推廣，早在 2 - 3 年前就已經投入相當多的成本建置先進的數位電視平台，並花許多時間在進行數位化的工作，對國內數位電視產業的推展不遺餘力。當此新法規制定的同時，是否會引響現有的數位電視推展，而且對於目前業者已提供給客戶的數位機上盒等產品，政府應考慮相關的配套措施，若會造成業者及用戶之損失，政府將如何彌補，以保障應有之權益。

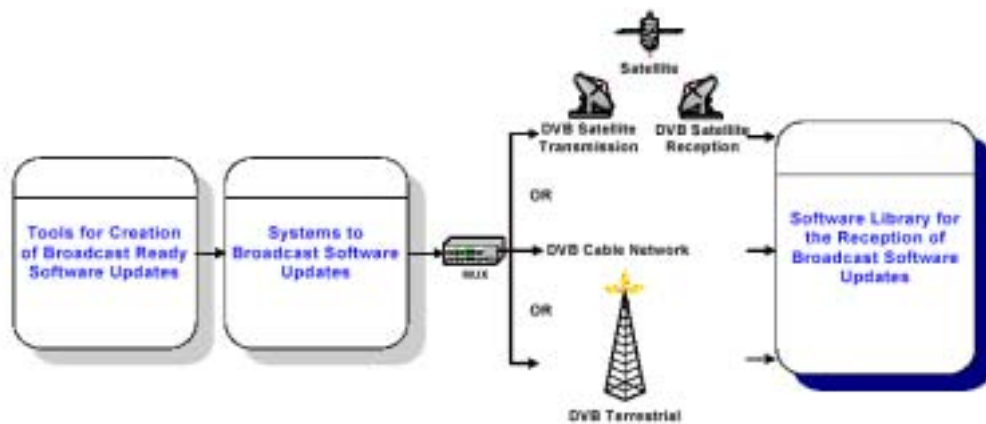
附件（四） DVB SSU 與 EuroLoader

1. DVB SSU 與 EuroLoader

附 4-1

DVB-SSU 與 EuroLoader

要更改數位電視接收機內的軟體，通常有兩種方式。第一種是透過 RS-232 之類的實體介面，以及 X-Modem 之類的通訊協定，以一對一的方式下載軟體至數位電視接收機。此種方式適合在數位電視接收機出廠前，或是在指定的維修站內，更新數位電視接收機內的軟體。但若要更新大量已經販售的數位電視接收機，此種方式就不是這麼地合適了。第二種方式，則是透過數位電視的頭端設備，播送欲下載至數位電視接收機的軟體。此種方式的優點是可以一次同時更新許多的數位電視接收機，只要數位電視接收機在頭端播送軟體更新資料時開機即可。DVB-SSU (System Software Update) 與 ECCA (European Cable Communication Association) 的 EuroLoader，即是支援此項服務的兩組標準，分別由 DVB 與 所制訂。



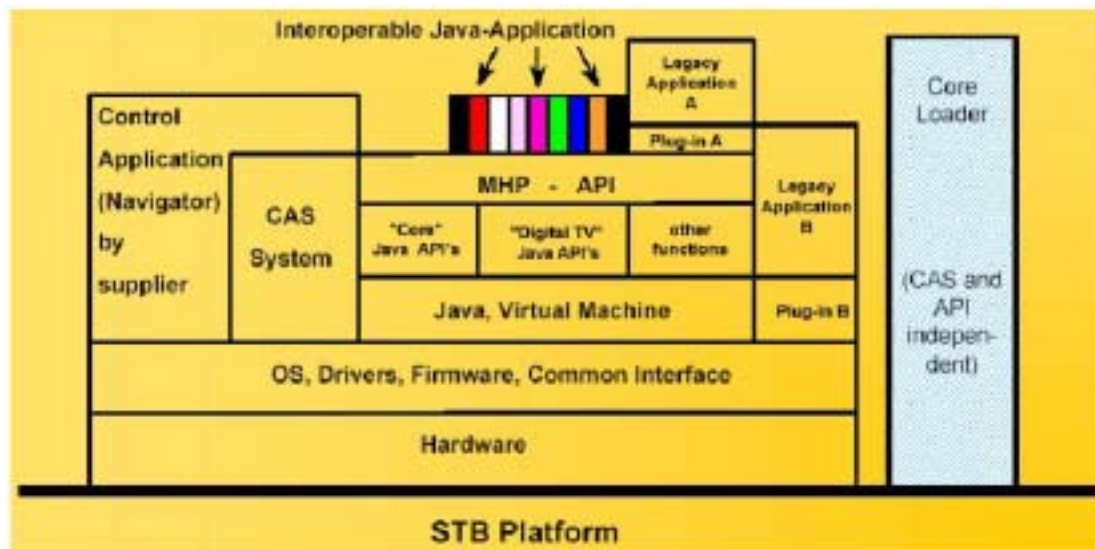
圖一、DVB-SSU 系統架構圖

如圖一所示，要支援 DVB-SSU 與 EuroLoader，必須於數位電視頭端及數位電視接收機內，提供以下的三種元件：

- 轉換格式之工具軟體：將欲更新軟體轉換為可以被數位電視頭端廣播出去的格式。DVB-SSU 支援兩種不同的資料廣播格式，第一種是定義於 DVB-SSU 標準 (ETSI TS 102 006) 內的 standard carousel 格式；第二種則是由數位電視接收機廠商，所自行定義之格式。不過，前述的兩種格式，在播送前都會被轉換成 MPEG-2 的 transport stream。因此，為了方便起見，有些電視台只接受數位電視接收機廠商，以 MPEG-2 transport stream 的格式，遞交所欲更新之軟體。
- 播送欲更新軟體之頭端設備：如果數位電視接收機廠商所交付之更新軟體格式，為 DVB-SSU 所定義之 standard carousel，則電視台於頭端

必須提供符合 DVB data broadcasting 標準之 carousel play-out system。但如果電視台收到的軟體格式為 MPEG-2 transport stream，則電視台僅需使用 MPEG-2 generator 播送欲更新之軟體。

- 數位電視接收機內的軟體程式庫：為了接收由電視台所播送之軟體更新資料，數位電視接收機內必須提供一套軟體程式庫負責此項功能。EuroLoader 所定義的即是負責此項功能之軟體，是數位電視 boot loader 的一部份。



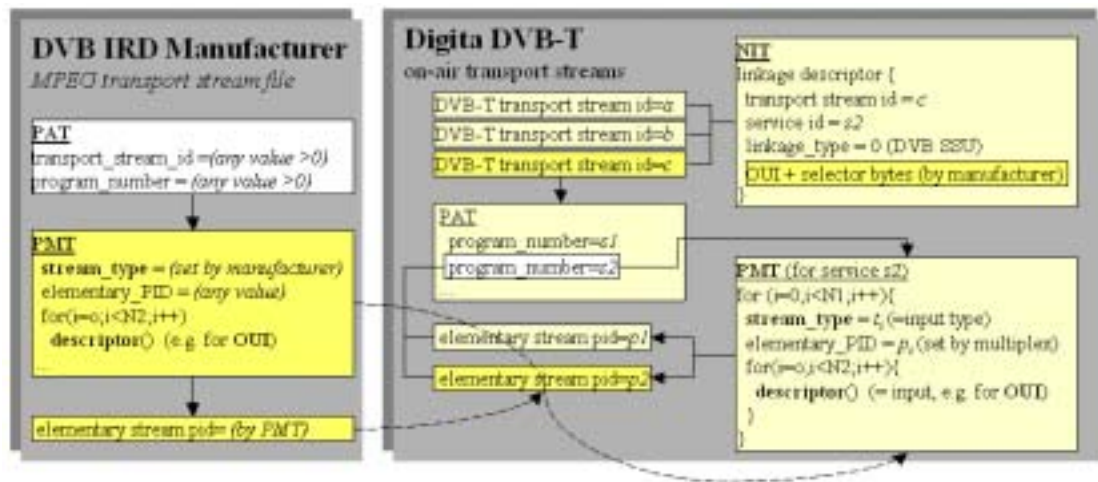
圖二、採行 EuroLoader 之數位電視接收機之系統架構圖

圖二為採行 EuroLoader 之數位電視接收機之系統架構圖，EuroLoader 獨立於 CA 與 middleware 之外。由 DVB-SSU 與 EuroLoader 的組合，可以更新數位電視接收機內以下類型之軟體，包括了：

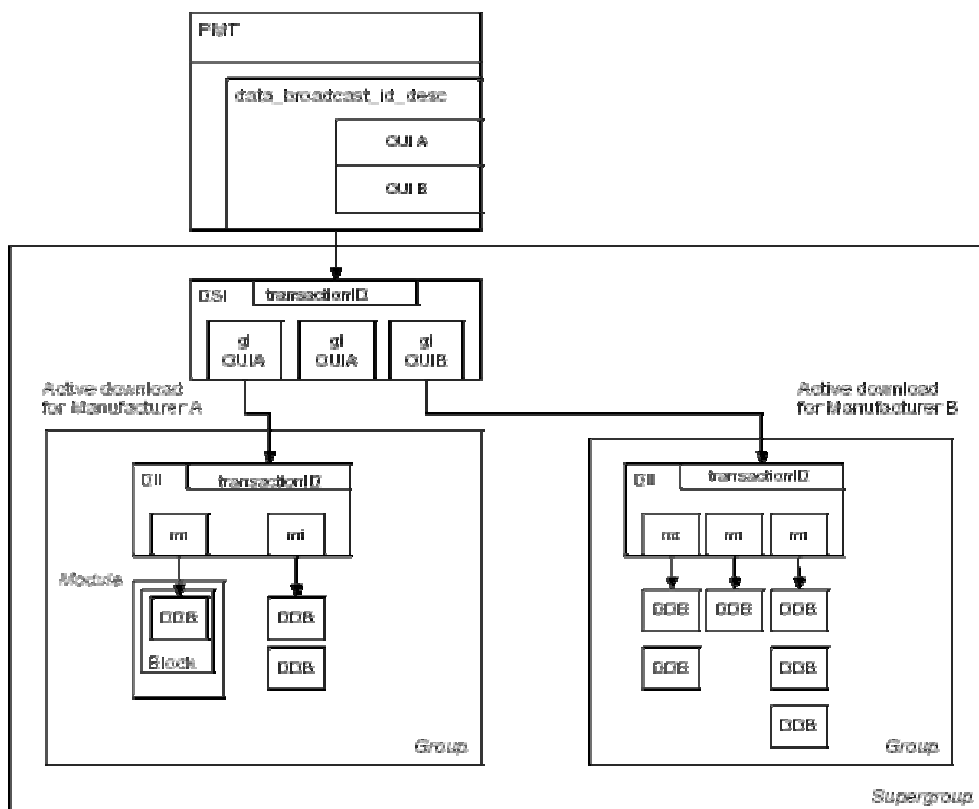
- Device firmware
- Operating systems
- Middleware
- Applications
- CA

DVB-SSU 的服務定位方式有兩種，第一種是所謂的 simple profile，也就是以 NIT/BAT 來定位提供 DVB-SSU 下載軟體的 PID。第二種則是採用 DVB-SSU 標準內定義之 UNT (Update Notification Table)，來達成服務定位的目的。採用 UNT 能提供一些額外的功能，例如：軟體更新排程、額外的選擇 (selection) 及

指定目標 (targeting) 的資訊 以及 filtering descriptor...等。



圖三、DVB-SSU simple profile 的服務定位方式



圖四、DVB-SSU 的 standard carousel

圖三是 DVB-SSU simple profile 服務定位方式的示意圖。在 NIT 內，linkage type 為 0X09 的 linkage descriptor，紀錄了某個 broadcasting network 內，所提

供的 DVB-SSU 服務的 OUI (Organization Unique Identifier) 及 selector。另外，如圖三所示，NIT 內每個 linkage type 為 0X0A 的 linkage descriptor，包含了提供 DVB-SSU 服務的 service 的 transport stream ID 及 service ID。透過 transport stream ID 及 service ID，則可以透過 PAT 找到提供 DVB-SSU 服務的 service 的 PMT。再透過 PMT 內的 data_broadcast_id descriptor，則可以找到提供 DVB-SSU 下載軟體的 PID。至於每個 DVB-SSU 服務所佔用的頻寬，則相當的有彈性，由欲下載軟體之大小及輪播時間的長短來決定。

圖四是 DVB-SSU 標準所定義之 standard carousel，它是一個兩層式的架構，由 DVB data broadcasting 標準所定義之 data carousel 衍生而來。一個 DVB-SSU standard carousel 由一個到多個 group 所組成。每個 group 所包含的是一個數位電視接收機製造商所提供的下載軟體，也就是說，一個 DVB-SSU standard carousel，可以包含來自多個數位電視接收機製造商所提供的下載軟體。為了達到上述的目的，DVB-SSU 的 standard carousel 對原本的 DVB data carousel 做了以下額外的規定：moduleID 的 bit 8 ~ bit 15 內的值必須為 DSI 內 groupID 的 LSB。moduleID 的 bit 0 ~ bit 7 則為 group 內專用的 unique ID，因此，每個 group 內最多只能有 256 個 module。基於上述的限制，在整合來自不同數位電視製造商之 DVB-SSU group 時，可以不需要考慮 group 間 moduleID 相衝突的問題。

附件（五） 其他

1. DVB's email for Original_Network_Ids and Network_Ids in Taiwan

附 5-1



DT/4433/DVB
pma
By E-MAIL

Hong Jen-Kwey,
Director, Department of Radio and TV
Broadcasting Technology,
Directorate General of
Telecommunications
Min. of Transportation & Communication,
16, Sec. 2, Chi-Nan Road, Taipei 100,
Taiwan R.O.C.

Geneva, 23rd June 2004

Original_Network_IDs and Network_Ids in Taiwan



Dear Mr. Hong,

I would like to thank you for your letter dated 14th June 2004, which was kindly delivered by Mr. Wong (ITRI) during the course of our 56th Technical Module meeting in Geneva.

Your request for an original_network_id and network_ids has duly been noted, and I shall proceed at the earliest convenience with such a registration. You should hear from me by e-mail during the course week 27, i.e. from Monday, 28th June 2004. I will endeavour to ensure that the IDs are made available prior to this.

I have made the following registration:

Original_network_ID	0x209E
Network_Ids	0x3301-0x3400

You can view these registrations on: <http://www.dvb.org/index.php?id=16>

Once again, I thank you for your letter and wish you great success with DVB-T in Taiwan.

Yours sincerely,

Peter MacAvock,
Executive Director, DVB Project

DVB Project Office, c/o EBU, 17a Ancienne Route, CH-1218 Grand Saconnex (GE), Switzerland

Tel: +41 22 717 2719

Fax: +41 22 717 2727

URL: <http://www.dvb.org>

國家圖書館出版品預行編目資料

『數位廣播電視系統整合之研究』研究報告/
交通部電信總局，財團法人工業技術研究院編
著． - - 台北市：電信總局，民 93
面；公分

ISBN 957-01-9107-4 (平裝)

1. 數位電視

448.88

93024882

題名：「數位廣播電視系統整合之研究」研究報告
編著者：交通部電信總局、財團法人工業技術研究院
出版機關：交通部電信總局
電話：(02)2343-3723
地址：臺北市濟南路二段 16 號
網址：<http://www.dgt.gov.tw>

出版年月：中華民國 93 年 12 月
版次：第一版
刷次：第一刷
工本費：NT\$1000
展售處：

GPN: 1009304088
ISBN: 957-01-9107-4 平裝

統一編號 GPN

1009304088

NT\$1000

ISBN:957-01-9107-4



9 789570 191073