

# 以 HAVi 多媒體網路整合於 OSGi 家用閘道器之設計與實作

## The Design and Implementation of OSGi-Compliant

### Home Multimedia Network System

彭興國

逢甲大學資訊工程所

[Monkey329@gmail.com](mailto:Monkey329@gmail.com)

張晉源

樹德科大資訊工程系

[cychang@mail.stu.edu.tw](mailto:cychang@mail.stu.edu.tw)

薛念林

逢甲大學資訊工程系

[nlhsueh@fcu.edu.tw](mailto:nlhsueh@fcu.edu.tw)

#### 摘要

本研究所提出之家庭多媒體影音裝置服務共享架構，是以HAVi技術整合家庭多媒體裝置，並且符合OSGi規範所建構而成的家庭多媒體網路系統（Home Multimedia Network System）。透過service bundle的形式使系統更具有彈性，並能有效地降低功能擴充成本與操作複雜度、更具高度可靠性。除此之外，本研究更進一步提出智慧型家庭概念，將家庭的資訊、控制、多媒體作一整合，並且透過行動裝置來達到遠端的操作控制，藉此讓生活跳脫空間的限制，讓使用者可以盡情的發揮自己的創意，建構出屬於個人的生活空間。

**關鍵字：家庭網路、HAVi、OSGi、Service Bundle**

#### Abstract

In this paper, we propose an architecture of sharing services between multimedia devices, in which all home multimedia devices are integrated by the HAVi technology, and the network system conforms to the OSGi standard. Due to the benefits of service bundle, the system is more flexible and more reliable. Moreover, in this paper we also propose the idea of smart home in our system which integrates information, control and multimedia in a home network. Users can control the home network through mobile devices wherever they are.

**Keywords : Home Network, HAVi, OSGi, Service Bundle**

#### 一、前言

根據資策會 ACI-FIND[9]於 2004 年的最新調查結果顯示，由於國內網際網路的普及廣泛與數位寬頻的成長，國內上網普及率已逾六成，而由於資訊與生活的融合越見明朗，消費者對於網路的要求與期望也相對的越高。除了期望頻寬更快速、加值服務更多元以外，更期望能透過網路來作一居家的家電控制與影音多媒體的整合與管理，充分讓物質感官的享受與生活資訊融為一體。

為達到上述之目的，則必須藉由具擴充性與整合性之智慧型閘道器所提供的使用管理功能，使智慧型家庭中的各項系統技術能發揮其最大的效益。因此面對如此的環境需求，需要具有適切因應措施與擴充機能的設備，能夠整合家中的智慧型系統。

家庭寬頻設備的使用趨向具備與其他裝置交換訊息網路功能的資訊家電產品-家用閘道器(RG, Residential Gateway)[19][20]。其主要設置於建築物的網路或家庭網路的入口，扮演著家庭內部網路與外部網際網路唯一的橋樑，因此將是不可或缺的一項重要裝置。

然而單就現今家庭所普遍使用的傳統閘道器，並無法將這些異質性網路實際整合。當中原因在於眾多的閘道器之中，並無遵循一開放性的服務標準，因而常造成不同廠商自行訂定介面造成規格不合、無法互通等情況發生，使用戶無法使用其他廠商所提供的服務，造成服務差異化。

因此，符合OSGi[4][10]規範是現今家用閘道器能否降低成本以及具備開放性的主要因素，標準化將使網路服務提供商以及資訊家電提供商可簡易的開發通用的服務包，並且易於更新抽換，也方便家庭各個系統元件日新月異的功能要求，並能有效降低家庭網路服務提供商的服務成本[24]。消費者亦可依據自身的需求來選擇所需的服務，而遠端的服務供應商 (Service Providers) 可透過家用閘道器 (Residential Gateway, RG) [19][20]，將用戶端所需的各式生活資訊與服務，透過網際網路，動態的下載至用戶端的家中，提高人們生活的便利性。

在家庭網路的多媒體娛樂部份，過去傳統的類比影音裝置大都是採用 AV 端子、S 端子來作一傳輸的媒介，但由於新一代的數位影音多媒體潮流逐漸取代過去的影音裝置，進而成為市場主流商品的同時，其傳輸媒介亦逐漸偏向 USB [32] 及 IEEE1394 為其傳輸主流。而在 IEEE1394 的高傳輸速率及傳輸距離等優勢的促長下，成為現今影視訊媒體的主要傳輸媒介，其 IEEE1394 與 USB 的比較如下表 1。

表 1 IEEE1394 b 與 USB2.0 之比較

	USB2.0	IEEE 1394 b
最大傳輸速率	480Mbps	800~3200Mbps
隨插即用	有	有
傳輸距離	5m	4.5~100m
周邊裝置	鍵盤、滑鼠 電腦螢幕 數據機 低速 CD-ROM 行動碟 讀卡機 硬碟盒	DV/D8 數位式攝影機 HDTV 高畫質電視機 硬碟機 掃描器 印表機 DVD-ROM Driver 硬碟盒
未來應用方向	電腦週邊設備	家庭網路、資訊家電、無線傳輸

然而，儘管在家庭中能擁有快速的影音媒體娛樂，卻無法實際透過一個網路來作一個影音多媒體的整合與管理，因而造成媒體資訊流通的封閉與使用上的麻煩，當中諸如：無法在任一時間地點觀看所有的多媒體資訊、媒體資訊受到空間的拘束...等。

因此，本研究為了將影音多媒體裝置網路導入家庭網路的一環，使其具備影音多媒體整合的能力以及有效率的作一管理與統整，並且能夠兼具技術融合 (convergence) 與應用互通性 (interoperability) 等兩大特性。所以，本研究採用其核心建立在 IEEE1394[6][7][16] 底層協定基礎上的 HAVi (Home Audio - Video Interoperability, 家庭視訊音頻協同運作技術)，來達到數位影音媒體的互通轉換功能。並且以一個服務包的形式掛載至符合 OSGi 規範的閘道器上。藉此除了可以使得各個應用技術達到“隨插即用”的功能，更可建構出一個完整的家庭多媒體社群。

本文第二節中將敘述本研究之動機與目的，第三節中將進行本研究相關文獻與研究的探討，第四節中將針對本研究的各項技術之基本架構、應用及設計原則逐一說明。本研究之相關實作與測試將於第五節中詳細說明。最後，於第六節中簡要說明本研究代理機制架構應用於閘道器的優點與結論。

## 二、研究動機目的

近年來，隨著智慧型家庭的潮流發展迅速，其家庭內部資源網路的周邊價值也相對提高，根據 In-Stat[2] 針對家庭網路的市場預估 (圖 1)，由於家庭網路頻寬的成長，影音多媒體等相關消費性週邊設備的產值將會從 2004 年的 83 億美金逐年成長到 171 億美金。

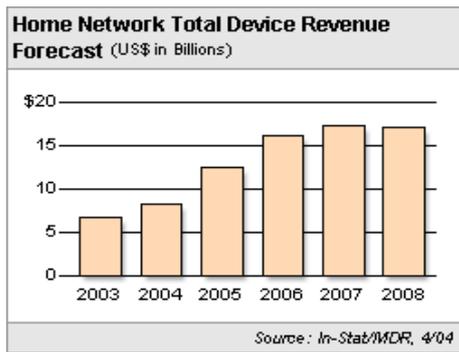


圖 1 全球家庭網路週邊設備市場預估

新一代的數位影音多媒體潮流已逐漸取代過去傳統類比的影音裝置，進而成為市場主流商品，當中對於傳輸速率的需求也相對的提高。因此在 1995 年，電機電子工程師協會 (IEEE) 制定 1394 成為正式的傳輸標準，以解決傳輸速率不足的問題。而伴隨著 1394 裝置的市場逐年穩定成長，飛利浦、SONY、松下及三星...等八家國際家電大廠，意識到未來智慧型家庭將會面臨到的多媒體影音互通作業性問題，因而共同擬定出一個標準並稱之為 HAVi (Home Audio - Video Interoperability, 家庭視訊音頻互通技術)[5]，其核心建立在 IEEE1394[6][7][16] 的底層協定基礎上，進而實現 HAVi 設備之間的互相溝通存取，形成一個裝置型態的 HAVi 網路鏈結。

概括以上所提及的相關資訊可知，未來家中的各項影音多媒體傳輸媒介，都將以 IEEE1394 為主流。相對的，對於一個家庭資訊網路而言，以 IEEE1394 所形成的裝置網路也將越顯得其重要性及必要性。

而在數位科技與網際網路普及化的驅使之下，形形色色的網路服務不斷的演進，以及數位內容與裝置間相互溝通的能力產生極大的關聯，導致服務提供者面臨到如何整合不同介面的軟硬體所帶來的問題。因此，本研究以一般使用者與服務供應廠商的角度去探討歸納出現今家庭網路的問題癥結點。

1. 傳統網路服務與設備所提供與網際網路所連結的能力已逐漸無法滿足一般使用者各式各樣的需求。

2. 傳統的網路架構將逐漸被隨插即用 (plug-and-play) 的裝置連結方式所取代。
3. 各家廠商所推出的資訊設備產品缺乏依循一共同標準。
4. 眾多的服務與架構缺乏有效地整合與管理。
5. 影音多媒體裝置缺乏一個互通性的服務及裝置間功能的整合，進而侷限其延展性及發展。

然而就目前而言，現今市面上之家庭網路整合技術諸如：UPnP、Jini、HAVi、LonWorks、OSGi ... 等，可說是如同雨後春筍般的不斷推出。但是如何同時具有技術融合 (convergence) 與應用互通性 (interoperability) 兩大特性才是未來智慧型家庭成敗的關鍵要素。

因此，本研究利用一個開放式服務平台 (OSGi) [4][10][11][12][13][23] 為整體家庭網路的基礎平台，當中不管是何種技術、語法或協定，只要遵循 OSGi 所提供的規範去開發、撰寫，即可在任一具有 OSGi 規範之閘道器裝置上執行。

而為了將影音多媒體設備作一整合，本研究則是採用了 HAVi [5][14][15] 整合技術，當中用以整合所有具 IEEE1394 規範之設備裝置，並且以一個服務包的形式掛載至符合 OSGi 規範的閘道器上。藉此除了可以使得各個應用技術達到“隨插即用”的功能，更可建構出一個完整的家庭多媒體社群。

### 三、相關研究探討

國外對於 HAVi 與其他中介軟體之間的結合應用及開發研究，最早期由 Sun 與飛利浦、SONY 兩家世界影音設備大廠，於 1999 年共同開發 Jini 和 HAVi 的閘道器軟體。初步的試驗成果，就是將 HAVi 及 Jini 的閘道器軟體植入視訊轉換器 (set-top box) 中，達成兩者的拓撲得以相互變換處理的效果。正式的商品化將始於一九九九年九月發表的 Sharp PDA Zorus 及 AV 設備，其使用紅外線傳輸，讓 AV 設備的音樂資料下載到 Zorus，使

Zorus 兼具音樂播放機的功能。除此之外，部份同業的個人電腦週邊設備業者，對於引進 Jini 表達相當高的興趣，其動機卻反而不是在以個人電腦為核心的相關產品角色，反而是欲藉 Jini 取得 HAVi 的连接能力[8]，進而使 AV 設備能與印表機和 HDD 相連，以擴展此兩產品在非個人電腦領域的市場。據悉，印表機和 HDD 業界擬將 Jini 控制的此兩項產品 API 各自在一九九九年夏天予以標準化，不過 AV 業界因有鑑於 Java 環境下的動作不夠快，正待昇陽設法解決以提高速度，故暫時不會積極搭載 Jini。

同年，SONY 亦提出以 HAVi 建構之 Home Network File System [17]，此系統主要以 IEEE1394 為家庭的影音傳輸媒介，進而將家庭的各項多媒體影音資源作一統整與管理，實現一家庭多媒體網路的構想，當中並可透過家庭伺服器(Home Server)來做為與外界網路的連結。

此系統雖然能夠利用 HAVi 來整合家庭多媒體，使其成為一個多媒體網路，但對於整個家庭網路來說仍然不夠健全，因為對於未來的數位家庭而言，單一的技術整合並無法實質上的滿足多元化的家庭網路發展。而後於 2001 年，Roli G. Wendorft 等人提出以遠端的型式連取家中的 HAVi 裝置網路[18]，並對裝置做控制、資料擷取等動作。其研究是以 Java RMI 為傳輸的型式，並透過 Web 來作為使用者的登錄介面。

同年5月，NOKIA 亦提出一個 VHE (Virtual Home Environment, 虛擬的家庭環境) 的架構 [3][21]，利用 OSGi、Jini、HAVi、CORBA... 等整合一個家庭內部的各項資源，進而透過行動裝置能達到及時的連結、監控保全、裝置互通... 等功能。

VHE 主要在於建構一個開放式的平台，並在之上加入一分散式的管理運算機制，藉其來整合家庭的各個中介軟體及不同訊息場之技術。目的在於建立一個全自動化、智慧型並具有遠端登錄存取功能的家庭。

另外，日本的東京大學及神奈川縣慶應義塾大學，也有著手於家庭影音裝置網路與行動運算裝置的連結測試與相關開發應用，其主要核心概念為

Smart Connect [22]，透過其自行研製的 API 來將所有的 AV 設備作統一的發配管理。此研究雖然能夠達到 AV 裝置的整合及管理，但卻缺乏一個標準規範，以供各界遵循甚至是相關的延續開發。

而國內對於 HAVi 的開發研究主要由台灣大學資訊工程學研究所針對 HAVi 與 Jini 之間的軟體轉換開道器[27]以及 HAVi 與 Java 的訊息溝通 [28]。當中，除了針對於 HAVi 與 Java 等不同訊息場之間的溝通研究以及 HAVi 軟體元素的基礎設計研究之外[29]，是藉由 Jini 裝置整合及跨平台的特性再配合上 HAVi 的影音媒體設備整合，使其多媒體整合網路能夠加入一分散式的運算環境之中，用以達到 HAVi 與 Jini 相互溝通與存取的目的。

然而 Jini 固然可以透過 Lookup Service 機制來整合各項服務[25]，以供使用者存取，但相較於 OSGi 的 Service Bundle，其所能夠整合的中介軟體層級以及可擴充性、功能性、使用性... 等，都優於 Jini 的整合能力。因此，本研究以 OSGi 為其開道器的標準平台，其上來整合各具質性的中介軟體技術[26]，藉此達到開道器的多元整合目的。

由上述得知，目前市面上的相關產品以及所發表的相關研究中，大部分都是針對裝置的互通性連結及行動裝置的應用來實作，並未達到家庭資訊網路的整合及提供完善的服務功能。在未來家庭網路的生活中，家庭內部網路、傳輸、通訊技術勢必走向多元化及多功化。因此，家庭內部裝置、媒體必須要有一個完整的整合規劃，使其內部資訊網路不僅具有隨插即用、隨機存取的特性，更要具備有開放式的整合機制，以迎乎瞬息萬變的資訊網路潮流。

基於上述等問題，本研究之 HAVi 多媒體網路在設計與功能方面，主要是依循 OSGi 的規範來開發，並利用當中的服務包註冊機制來提供給各項不同標準技術的裝置網路去註冊本身的狀態資訊，進而形成一個技術社群化的模組，而遠端服務使用者則可藉由登錄該開道器來下載並存取所需的多媒體服務。

## 四、系統架構

如下圖 2 所示，本研究主要探討、實作的部份，是如何在一個共同的平台協定之中(OSGi)加入 HAVi 群組網路做一連結[2]，進而在家庭資訊網路中，”達到多媒體影音裝置即時共享”，讓多媒體裝置網路能夠以一個群組服務的型態實現在家庭資訊網路之中。

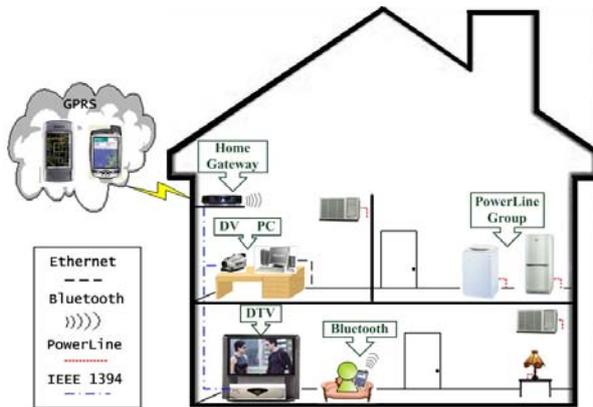


圖 2 基本架構

### 4.1 系統架構

本研究之基礎架構如圖 3 所示，在硬體之下為通訊介面，其分別為 Ethernet、IEEE 1394、Wireless LAN 三種，在硬體之上則是採用 Linux 作業系統之嵌入式系統[30]，在其上是由 J2ME 版本 + CDC 作為 Java 系統的虛擬機器[1]。接著是 Framework 建置，在當中本研究採用符合最新板本 OSGi 3.0 的 OSCAR 為其平台介面，並在此 Framework 中架構 HTTP、LOG、Device Manager、LonWorks Service 等基本服務。以下將探討本計劃之家用開道器與多媒體整合機制之設計與運作方式：

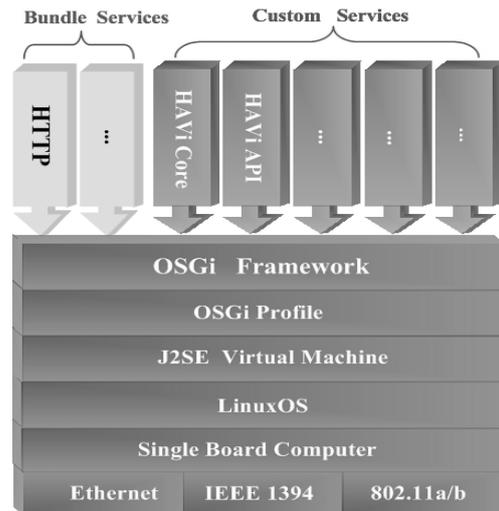


圖 3 家用開道器基本架構

### 4.2 HAVi 規範架構

HAVi 的協定架構的組成，是基於硬體介面層到軟體介面層之間的關係及連結。其協定結構圖如圖 4 所示。

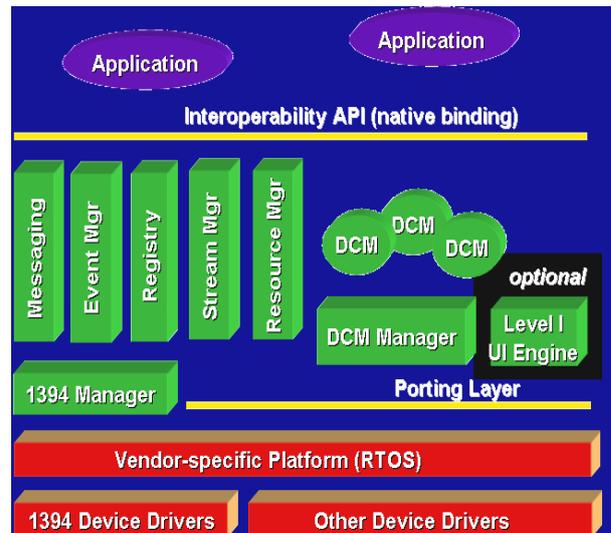


圖 4 HAVi 架構模型

為要實現 HAVi 之媒體裝置網路，因此在 HAVi System 的部份將規劃實作 Communication Media Manager、Messaging System、Registry、Event Manager、Stream Manager、Resource Manager、DCM、DCM Manager 等八類機制。

### 4.3 裝置分類

而就 HAVi 的協定架構中，多媒體裝置又被定義成四個不同的層級，其層級的劃分，則依據裝置本身所具備的軟體、硬體條件來做一決定，分別是 FAV (Full AV)、IAV (Intermediate AV)、BAV (Base AV)、LAV (Legacy AV)等四種層級[10]，其裝置型態的詳細內容可參照下表表 2 所示。當中，本研究規劃實作的 HAVi Core 即須屬於 FAV 或是 IAV 的裝置型態。

表 2 HAVi 裝置型態類別

Software element	Device class			
	FAV	IAV	BAV	LAV
Java	x			
DDI controller	(x)	(x)		
Resource manager	x	(x)		
Stream manager	x	(x)		
DCM manager	x	(x)		
Registry	x	x		
Event manager	x	x		
Messaging system	x	x		
Communication media manager	x	x		
Device control module	x	(x)	x	x
IEEE 1394	x	x	x	(x)
IEEE 1212r	x	x	x	(x)
IEC 61883.1 FCP	x	x	(x)	(x)
IEC 61883.1 CMP	x	(x)	(x)	(x)

## 五、軟體架構

本研究在功能模組設計方面，共分為 Lookup Service Module、Information Manager Module、Device Control Module、Graphics Interface Module 等四個功能模組。在下面則將會介紹這些模組的功能定義以及各個模組之間的關係。如下圖 5 所示，顯示各模組間的交流互動及功能性。

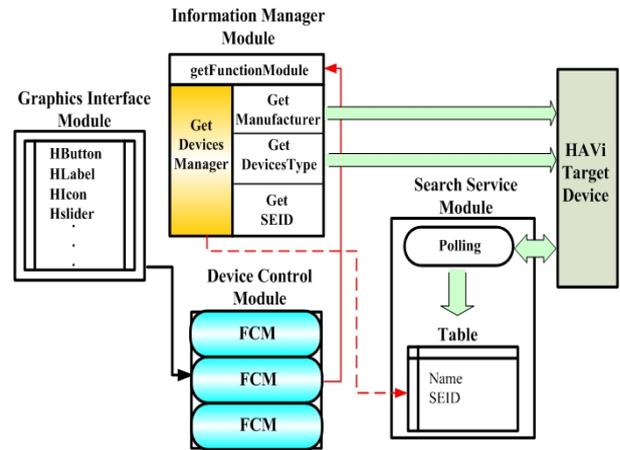


圖 5 模組關係結構圖

### 5.1 搜尋服務模組(Lookup Service Module,LSM)

搜尋服務模組在此一 HAVi Core 中所扮演的角色，是一個針對於 HAVi 裝置網路做即時監聽的機制。透過 HAVi 所訂定的 HaviListener() 方法來實作，來達到 HAVi 網路系統的即時監測，隨時更新裝置資訊、狀態，讓使用者可以確實而有效率的掌握 HAVi 社群網路之中的所有裝置資訊及功能服務。

### 5.2 訊息管理模組(Information Manager Module)

訊息管理模組主要在於裝置網路之中，取得所有已註冊登錄的裝置資訊，當中包含了裝置的名稱、類別、製造廠商以及裝置之相關功能元件...等資訊。讓 Client 端在要求 HAVi Core 提供裝置服務之前，可以預先知道該裝置的服務資訊為何，進而依照相關的需求提出服務的請求。

### 5.3 裝置控制模組 (Device Control Module)

裝置控制模組主要的作用在於結合前幾節所提到的模組功能，進而建構出一個完整的使用者控制服務介面。Client 端透過此一服務控制介面，利用當中所提供的控制元件 (如按鍵、調節器...等)，來達到操控遠端裝置的服務。

而在 HAVi 的規範所定義的 DCM 是由不同的功能零件模組(Functional Component Modules, FCM)所組織而成的。換言之，DCM 即是由許多的 FCM 元件所結合的一個完整操作控制介面。

本研究對於裝置控制模組，共提出了 getSynDCM 與 getSynFCM 兩個方法來達到同步的裝置控制。但就功能性而言，getSynFCM 所提供的功能都較屬於基本的操控元素，如 Play、Stop、Exit...等；而 getSynDCM 所能提供的服務則是透過許多個 FCM 的功能元件所共同建構而成。因此，除了 FCM 所定義的基本功能之外，還可額外對於該裝置做電源、資料儲存...等動作。也就是說，DCM 功能性是由多個不同的 FCM 所建構而成，其功能更可能因為 FCM 的增加而更加多元化。如圖 6 所示，FCM 與 DCM 的關係。

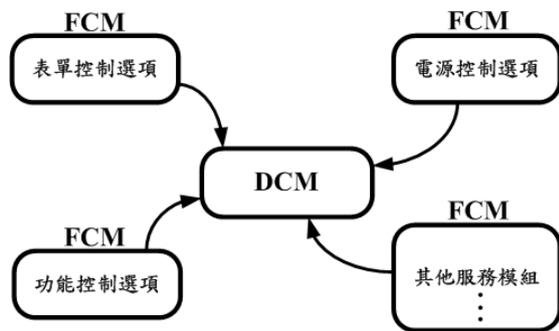


圖 6 FCM 與 DCM 之關聯圖

#### 5.4 圖形介面模組(Graphics Interface Module)

由於 HAVi 所制定的 org.havi.ui package，並沒有被詳細的定義出來，而這也將造成使用者在使用上的困擾與麻煩。因此，本圖形化介面模組主要是詳細的定義 HAVi 所屬的 UI (user interface)，將各個圖形元件功能一一詳細定義，並且將之封裝成新的 package “stu.havi.GI” 以供其他 method 或是使用者來開發應用。

## 六、系統實作

### 6.1 執行環境建置

本研究的系統環境建置方面，大致可分為硬體平台及軟體平台兩類，而硬體平台為顧及需有 IEEE 1394 傳輸介面以供連結裝置，因此採用 X86 系列的單板。軟體平台部份則是採用一貫的 JVM，其細部說明如以下小節所述。

### 6.1.1 硬體平台規格

本研究之硬體平台採用 VIA EPIA M 系列的 Mini-ITX 主機板。EPIA M 系列採用 VIA Apollo CLE266 晶片組，其內嵌式 MPEG-2 解碼器可使 DVD 更加流暢地播放，並整合 2D/3D 繪圖器，用以提供更豐富的多媒體體驗。EPIA M 系列具備 DDR266 SDRAM 的龐大記憶體頻寬與 ATA/133 高速資料傳輸，可確保現今高效能需求的數位媒體應用。其內建的 VIA VT1616 晶片更支援到六聲道的音頻解碼。EPIA M 主機板內建 IEEE 1394 和 USB 2.0 插槽可支援高頻寬的資料傳輸，且對 S-Video、RCA TV-Out (NTSC & PAL) 和 10/100 快速乙太網路的連結也有流暢寬頻的支持。

### 6.1.2 軟體平台建置

本研究在硬體之上則是採用 Linux 作業系統之嵌入式系統[31]，往上包含了 J2ME CDC Java Virtual Machine、OSCAR Framework 以及 Service Bundle。

OSCAR Framework 所提供的 Service 包含 System Service 與 Custom Service 兩種。System Service 提供 HTTP 與 LOG，HTTP Service 提供使用者能夠設計提供 Web Service Bundle，LOG 等。為了強化閘道器的功能，將提供自行開發的 Custom Service 提供給使用者應用，包含了 FTP、Mail、HAVi、Mobile Agent...等服務。除了在閘道器上的應用，我們也將把這些服務運用在行動裝置上，讓使用者可以透過手機與 PDA 上網控制閘道器上的服務，使我們的生活更加的舒適便捷。其軟體建置的規格如下表 3 所示。

表 3 軟體平台建置規格

Development environment	Fedora 1
Cross - Toolchain	GNU Binary Utilities GNU Compile Collection GNU C Library
Flash Memory Programmer	JFlash

Bootloader	LILO
common UNIX utilities	BusyBox
Operating System Kernel	Linux Kernel
OSGi Framework	OSCAR
Java Virtual Machine	cdcfoundation

## 6.2 HAVi System 實作

為要實現HAVi 之媒體裝置網路，因此在HAVi System 的部份將規劃實作Communication Media Manager、Messaging System、Registry、Stream Manager、Resource Manager等五類機制，以作為裝置底層的運作基礎。

其實作是以C++ 來建構底層裝置間的Communication Media、Message connection、Stream Channel，當中將使用IEEE1394 官方提供的函式庫raw1394.h來作一驅動程式間的溝通及相關指令存取的動作，並依據本研究架構之需求來開發其他的函式庫(如圖 7 所示)，藉由JNI (Java Native Interface)的連結以提供上層由Java 開發的應用層式作裝置硬體的驅動與連結操作。

```

havi_system.c
#include<stdio.h>
#include<jni.h>
#include<unistd.h >
#include< errno.h >
#include<raw1394.h>
#include< kernel-raw1394.h >
#include< raw1394_private.h >
#include"rominfo.h"
#include"simpleavc.h"
#include"topologyMap.h"
#include"topologyTree.h"

TopologyTree *topologyTree;
raw1394handle_t handle;

```

圖 7 HAVi System 之相關函式庫

HAVi System 所實作之功能性為取得裝置在1394 網路拓撲之中的實體位址與其裝置之裝置型態、GUID (裝置識別ID)、裝置製造商...等相關裝置基本資訊，以作為上層應用程式的訊息取用及統一管理。

### 6.2.1 C source code 的開發與建置

利用 C 語言在控制硬體上的長處，本研究的裝置控制部份乃採用 C 語言開發，其目的為搭配底層裝置驅動程式，取得控制裝置的能力。在程式的規劃部分，分為裝置拓撲管理(topology tree)、裝置控制函式以及 bus 初始化。

在開發初期，需先探討底層 IEEE1394 API(libraw 1394) [33][34]，其對於本研究有應用到的函式加以分析、應用，並且引入其相對應的標頭檔，重要的標頭檔包括：

1. "libraw1394/raw1394.h"
2. "libraw1394/csr.h"

在拓撲管理方面，其重要的標頭檔包括 topologyMap.h、topologyTree.h，拓撲的方式乃依照 IEEE1394 的標準配置。以下為 topologyTree 部份程式片段：

#### *topologyTree 重要的資料結構*

```

typedef struct TopologyTree_t {
    SelfIdPacket_t          selfid[4];
    Rom_info                rom_info;
    char                    label[256];
    struct TopologyTree_t *parent;
    struct TopologyTree_t *child[MAX_CHILDS];
}

```

在裝置控制函式方面，本研究採用的實驗裝置為 Digital Video，其基本功能為播放(Play)、暫停(Pause)、停止(Stop)、快進(Forward)、倒轉(Rewind)、錄影(Record)，每一項功能都由其對應的底層函式在處理，透過一組控制碼使得相對的功能動作。底層 API 部分使用 raw1394\_write 函式，傳入由拓撲得到的裝置 ID，以及功能控制碼，即可控制指定裝置動作。

## 6.3 HAVi Core 實作

本研究所開發的 HAVi Core 目的在於透過HAVi Core 來補足原先HAVi API 裡面所缺乏的部份以及使用者介面端的開發問題，因此，除了根據HAVi specification 所定義的功能內容來實作出HAVi

API以外，更透過本研究所提出的功能模組來強化原API 在資源管理、分配等方面的功能，並且能夠透過使用精簡化的HAVi Core API 來讓使用者可以更輕易的著手開發相關的HAVi 應用。

HAVi Core 之實作目的是為了能夠取得並規劃整理所有 1394 裝置的分配及管理，以及對目標裝置作一指令動作的存取與控制甚至是裝置對裝置間的資料、影像互通操作...等功能。當中透過 JNI (Java Native Interface , Java 原生介面) 來作為與 HAVi System 溝通連結的橋樑，並進而與底層硬體作一連結驅動及取得其裝置之相關資訊，以提供使用者介面顯示及相關的動作所需。

而 HAVi Core 的組成架構如下圖 8 所示，共由 DeviceInformation、InformationManagers、LookupService、DeviceControls、DeviceChannelService 等十餘支程式組成，主要目的在於實現本研究所規劃的四個模組，以作為裝置資訊的匯集、管理及控制之用途。

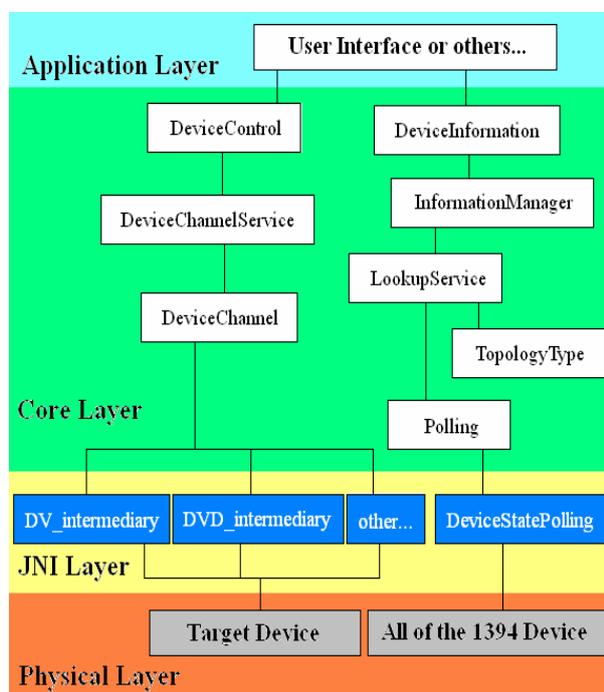


圖 8 HAVi Core 組成架構

#### 6.4 Java Native Interface 的建置與開發

為了補足於原本 HAVi API 所缺乏的擷取裝置狀態規範更具完善，本研究在 CoreLayer 中的

LookupService 提出一個專屬的 Polling 機制，並且建立一程式介面命名為 DeviceStatePolling.java，當中透過載入 HAViMiddleCore 的形式來取得 JNI 的訊息，並從中作一指令的傳遞與控制。

而在 DeviceStatePollin.java 的程式介面當中定義了以下幾個介面方法：

原生介面名稱	型泰	功能解釋
plug_in()	int	作為判斷旗標用，判斷是否有新裝置加入
plug_in_over()	void	作為判斷旗標用，判斷是否取完裝置狀態
get_GUID()	String	取得裝置的 GUID
get_Type()	int	取得裝置的型態
getManufacturer()	int	取得裝置的廠商名稱

透過以上這幾個介面方法，便可以及時取得各裝置的基本狀態，而 Core Layer 也可以利用這些基本的資訊，針對以連接上來的裝置做基本的資源管理與搜尋，例如：LookupService 所衍生出來的 TopologyType 功能，可以藉由 plug\_in() 與 getDevice\_GUID() 來達到目前現有裝置的拓撲型態。

#### 6.5 HAVi Service Bundle 實作

本研究在 Service Bundle 的設計部分主要關鍵架構在於將 HAVi 技術嵌入 OSGi 的環境後，如何讓一個共通平台之間的所有中介軟體等不同訊息場之間能夠相互取用其服務，進而能夠達到更深層的應用。在技術層面方面，除了使用 Sun 所開發的相關 Java 技術(如 RMI、servlet)，並且遵循 OSGi 規範來實作出本研究之家用開道器嵌入式服務模組，而這些架構在 Service Framework 上的 custom services 都是以 bundle 的形式執行測試，其測試畫面如圖 9 所示。

```

-> ps
START LEVEL 1
  ID State      Level Name
[ 0] [Active] [ 0] System Bundle (1.0.3)
[ 1] [Active] [ 1] Shell Service (1.0.0)
[ 2] [Active] [ 1] Shell TUI (1.0.0)
[ 3] [Active] [ 1] Bundle Repository (1.2.0)
[ 4] [Active] [ 1] HTTP Service (1.1.1)
[ 5] [Active] [ 1] Servlet (1.0.0)
[ 6] [Active] [ 1] HAVi Core
[ 7] [Active] [ 1] HAVi System
[ 8] [Active] [ 1] HAVi Agent
[ 9] [Active] [ 1] Jini Lookup Service
->

```

圖 9 Service Bundle 執行畫面

### ■ HAVi Service Bundle 實作與註冊

HAVi\_Demo應用程式主要是實作HAViService所定義的介面，取得HAVi device狀態值、控制HAVi device等動作都是在該程式執行。

如圖 10 所示，在 Service Framework 的運作過程中，本服務模組中的 Service bundle 會 export stu.havi.information 、 stu.havi.lookup 、 stu.havi.GUI、 stu.havi.DCM...等 package 和註冊 HAVi System 服務至 Service Framework 上。

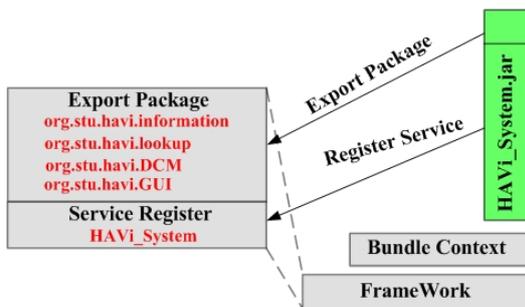


圖 10 export package 和註冊 HAVi\_System

在 HAVi Core 實作方面，與 HAVi System Bundle 溝通時，HAVi Core 模組會 import Service Framework 中的 stu.havi.lookup package 與 stu.havi.information package，來實作出 HAVi 裝置之資訊取得與控管。如圖 40 所示。此服務模組的 service bundle 在本研究中命名為 HAVi Core.jar，包含 JHAVi\_Control 應用程式、HAVi\_Network.jar 等兩支檔案，在 JHAVi\_Control 應用程式裡直接呼叫

HAVi\_Demo 應用程式的物件來引用，因此 HAVi Core 模組等於是轉而交付給 HAVi\_System 服務模組處理 HAVi device 控制與管理的部份。如下圖 11 所示。

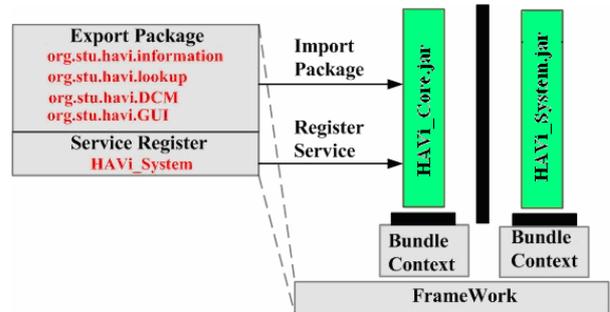


圖 11 import package 和取得服務

## 七、系統測試與驗證

在使用者面板的設計中，本研究是利用 Falsh 2004MX 來規劃設計使用面板，並且從事相關的圖形元件的設計，以配合本研究所開發的 API 配套取用。

相關的介面及相關的控制元件功能在以下將逐一作介紹，並請參照下圖 12。



圖 12 系統使用介面

1. 關閉系統：關閉系統所有正在執行中的程式及程序。
2. 功能列表：順序由左至右分別是一開啟所點選之裝置資源、觀看 IEEE 1394 拓撲型態的分布狀況(圖 13)、依照裝置型態顯現其對應之裝置

資源、依照製造廠商顯現其對應之裝置資源、系統簡介說明。

3. 裝置表單：顯示 IEEE 1394 網路中可供使用者使用的裝置資訊(開道器、Hub...除外)，並可藉由功能列表中的機制，來設定為依據何條件來搜尋、顯現裝置資訊，進而選取之後可進行控制、存取等動作。



圖 13 IEEE 1394 網路拓撲狀態介面

4. 控制選項：系統將依據使用者所選擇的裝置，提供該裝置所對應的相關控制面板及其控制的指令程式，讓使用者對其裝製作操作等動作。而本研究在操作面板及 API 的設計上，也開發出客制化面板風格，以供使用者、設計師選擇使用。其風格大致規劃有可愛版、正常版、豪華版三類型(圖 14)，以供多元的選擇。

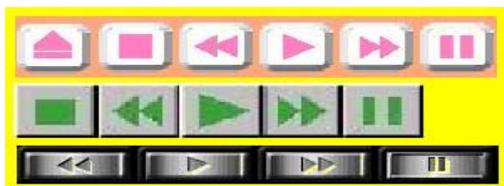


圖 14 控制面板風格

5. 顯示窗口：提供各種狀態的輸出，例如影像的輸出、拓撲型態的狀態輸出。

而使用者透過 HAVi Core 的影音管理機制可即時的取得家中任一 IEEE1394 設備裝置上的影音資源並且執行撥放、儲存...等動作。其多媒體在呈現方式如下圖 15、16 所示。



圖 15 DVD 影像顯示畫面



圖 16 DV 影像顯示畫面

使用者可從使用者介面得知 HAVi Core 所提供的裝置訊息管理表單，並進而選擇所要取用的裝置服務，當中藉由 HAVi System 來對於目標裝置做操作及影像擷取等動作，而使用者介面實作方面，主要是著重於以 Java Application for JMF(Java Media Framework)的技術來處理影像輸出的部份。

此外，本研究除了在一平台上作一測試外，另外也實作出符合行動裝置的連取服務，使得行動裝置能在遠端的情況下，仍然能透過連取家用開道器來取得服務包的資訊，但由於受限於無線傳輸的速率問題，因此，本研究在行動裝置的功能測試上，僅開發出支援裝置的操控(不包含影音的播放)，如下圖 17、18 所示。



圖 17 行動裝置操作介面

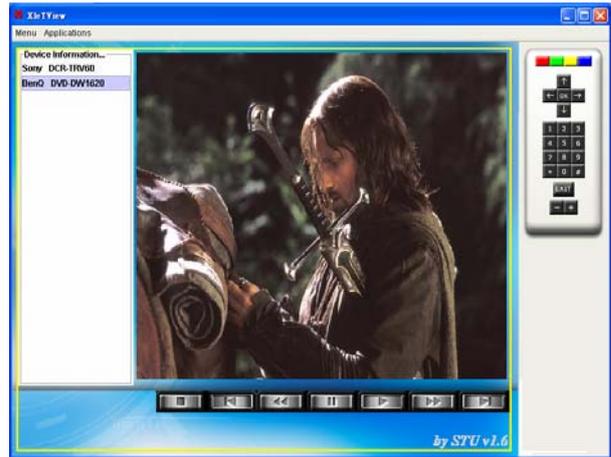


圖 19 HAVi Service Bundle 於 MHP 執行畫面

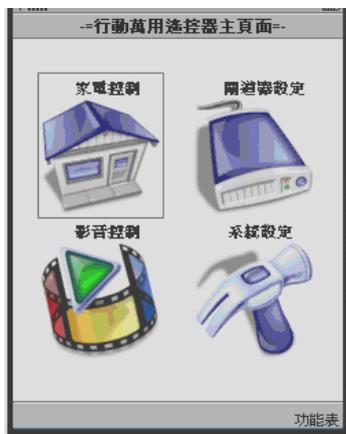


圖 18 行動裝置操作系統主頁面

而本研究的多媒體整合技術同時也應用在數位電視的相關測試中，在本實驗室還在測試階段的 MHP (Media Home Platform, 家庭多媒體平台) 數位電視技術之中，移植測試了本研究所開發的 HAVi 多媒體整合技術，並且透過 MHP 平台來取用 HAVi 的相關資源服務。

當中即是透過 OSGi 的服務包機制 (Bundle)，在一個共通的開道器平台介面之上來整合所有的資源服務。各種不同訊息場的技術皆能夠透過 Bundle 的形式來取得同一平台中所註冊的服務，並下載安裝執行，其成果展示如下圖 19。

## 八、結論與未來發展

由兩方面可以來簡述本研究的目的，在整合方面提出以影音裝置社群化服務導向之資訊整合介面，將能提供一個支援多媒體裝置的整合應用，在控制方面希望能改善原本一成不變的單一影音裝置單一控制為目標。

在居家生活方面，各類技術所整合的裝置、媒體網路都能透過群組的方式，來提供使用者一個舒適便捷的使用環境，並且讓裝置社群之間的管理達到簡易而不繁雜，社群相互溝通存取，資源分配共享的資訊整合服務，真正讓多媒體資訊網路達到技術多元融合、應用共通整合的境界。

此外，透過本研究在 HAVi 這樣的多媒體架構中所開發出的 API 套件，更可以讓原先官方所公開的測試版 API 功能更趨於完整，其系統介面及使用者面板、客制化選擇等方面也是有別於各方研究所沒有的成果，在 HAVi 的相關研究領域中可說是樹立一個新的里程碑。

而在目前政府正極力推動的數位電視之相關計畫中，可預期的家庭影音多媒體資源將被整合在其中，進而成為一個完美的家庭多媒體網路中心，所以將家庭多媒體相關之影音裝置整合於數位電視當中，相信也是未來的發展重點之一。

因此本研究未來將會繼續朝向更深入的相關探討與開發應用，並廣泛的學習新知，以豐富所研

究的領域，期待能結合更多的理論與實作讓研究更佳的務實有效率。

### 誌謝

本論文由國科會研究計畫「目標導向與樣式輔助之物件導向設計方法之研究」/NSC 93-2213-E-035-037 與「開放式服務平台嵌入分散式運算環境之研究」/NSC 93-2213 -E-366-013所補助，特此感謝。

### 參考文獻

- [1] Sun Microsystems, <http://www.sun.com>
- [2] Cahners In-Stat, Service Provider Infrastructure, <http://www.instat.com>
- [3] NOKIA, <http://www.nokia.com/>
- [4] OSGi, <http://www.osgi.org>
- [5] HAVi, <http://www.havi.org>
- [6] IEEE, <http://standards.ieee.org>
- [7] IEEE1394 , <http://www.1394ta.org>
- [8] The Community Resources for Jini Technology,<http://www.jini.org>
- [9] Institute for Information Industry, <http://www.iii.org.tw>
- [10] OSGi , "The Home Network Market-Current Status and Solutions" , 2003 , [http://www.osgi.org/documents/news\\_events/events/OSGi\\_Korea\\_Home\\_Network\\_Industrial\\_Forum\\_final.pdf](http://www.osgi.org/documents/news_events/events/OSGi_Korea_Home_Network_Industrial_Forum_final.pdf)
- [11] D.Marples and P. Kriens, "The Open ServicesGateway Initiative: An Introductory Overview,"IEEE Commun. Mag., vol. 39,no.12, Dec.2001.
- [12] Joong-Han Kim, Sung-Su Yae and R.S.Ramakrishna, "Context-Aware Application Framework based on Open Service Gateway," Info-tech and Info-net, vol. 3, pp. 209-213, 2001.
- [13] L. Gong, "A software architecture for open service gateways," IEEE Internet Computer, vol.5, pp. 64 -70, Jan.-Feb. 2001.
- [14] T.Saito, I.Tomoda, Y.Takabatake, J.Arni and K.Teramoto, "Home gateway architecture and its implementation" , Consumer Electronics, IEEE Transactions on , Vo.46,no.4 , pp.1161 -1166, Nov.2000.
- [15] R.Lea, S.Gibbs, A.Dara-Abrams and E.Eytchison, "Networking home entertainment devices with HAVi",IEEE Internet Computer , vo.33 , pp. 35 - 43 , Sep.2000.
- [16] Don Anderson , "FireWire System Architecture: IEEE 1394a, 2/e" , Addison Wesley ,1999
- [17] T.Igarashi, K.Hayakawa, T.Nishimura, T.Ozawa and H.Takizuka, "Home network file system for home network based on IEEE-1394 technology", Consumer Electronics, IEEE Transactions on, Vol.45, pp.1000 - 1003, Aug. 1999.
- [18] R.G.Wendorft, R.T.Udink, M.P.Bodlaender, "Remote execution of HAVi applications on Internet-enabled devices", Consumer Electronics, IEEE Transactions on, Vol.47, pp.485-495, Aug.2001.
- [19] D. Valtchev and I. Frankov, "Service gateway architecture for a smart home," IEEE Commun. Mag., vol. 40, pp. 126-132, Apr. 2002.
- [20] K.Wacks, "Home systems standards: achievements and challenges." IEEE Commun. Mag., vol.40, pp. 152-159, Apr. 2002.
- [21] Titos Saridakis, "VHE : an Approach to the System Architecture ", [http://www.vtt.fi/ele/new/vheseminar/saridakis\\_presentation1.pdf](http://www.vtt.fi/ele/new/vheseminar/saridakis_presentation1.pdf)
- [22] 徳田 英幸、村井 純、楠本 博之、中村 修、南 政樹 "機器の連携による 動的なメディアデータ通信経路生成機構の構築"，日本應義塾大學環境情報學部，平成 13 年
- [23] 張碩文、林文瑋、林政良、張晉源"應用家用

開道器建構 LonWorks 介面之智慧型家庭控制網路”，2004 數位生活研討會，成功大學，台南，台灣，民國 93 年

- [24] 陳邦正 “以 OSGi 為基礎之嵌入式 LINUX 網路服務開道器設計與實現”，國立交通大學碩士論文，民國 90 年
- [25] 林正良、彭興國、張俊輝、張晉源“支援 Jini Community 與家用開道器之設計與實現”，第十五屆物件導向技術及應用研討會，長榮大學，台南，台灣，民國 93 年
- [26] 陳弘翔、楊欽雄、管世達、張晉源“OSGi 平台上 HAVi 與 Jini 協同運作之機制設計”，2005 數位生活研討會，成功大學，台南，台灣，民國 94 年
- [27] 徐國偉”HAVi 與 Jini 之間軟體開道器的設計及實作”，國立台灣大學資訊工程學研究所碩士論文，民國 90 年
- [28] 余定穎”HAVi 訊息系統之設計與實作”，國立台灣大學資訊工程學研究所碩士論文，民國 92 年
- [29] 李務誠” HAVi 軟體元素設計方法”，國立台灣大學資訊工程學研究所碩士論文，民國 92 年
- [30] 陳邦正” 以 OSGi 為基礎之嵌入式 LINUX 網路服務開道器設計與實現”，國立交通大學碩士論文，民國 90 年
- [31] Alessandro Rubini,Jonathan Corbet，2004，Linux 驅動程式 第二版，歐萊禮，台北。
- [32] 郭士秋，2005，USB 2.0 理論與規範，儒林，台北。
- [33] 郭嘉龍，2001，IEEE-1394 AV 標準介面應用，全華科技，台北。
- [34] 龐丹鴻，2002，IEEE 1394 設計與程式開發—使用 C++語言，文魁資訊，台北。
- [35] 拓璞產業研究所，2004，引爆資訊、通訊與家電產業版圖重整之數位家庭，拓璞科技，台北。