

# 整合網際服務與 OSGi 服務

## Integrating Web Services and OSGi Services

李允中 馬尚彬 張文芳

國立中央大學資訊工程學系

Email: {yjlee, albert,  
sff@selab.csie.ncu.edu.tw}

謝禎罔 賀光鵬

財團法人資訊工業策進會整合實驗室

Email: {cchsieh, phomarc@iii.org.tw}

### 摘要

本論文提出一個整合網際服務(Web Service)技術與開放式服務閘道器標準(OSGi)的方法，透過 OSGi 服務接合器(adapter)，將後端的 Web Service 包裝為 OSGi 服務，以 Web Service 作為後端服務提供機制，OSGi 標準作為前端服務遞送機制，或將 OSGi 服務轉包為網際服務，以提供遠端存取。

We propose an integration mechanism for Web Services and OSGi services. The focus of this research is two-fold: (1) to transit Web Services into OSGi services to enable an in-car service delivery, and (2) to wrap OSGi services into Web Services for remote access.

**關鍵詞：**OSGi、網際服務(Web Service)、服務接合器(Service Adapter)

### 一、緒論

回顧過去幾年，網際網路從無到有，使用者只要透過瀏覽器，就可以使用網際網路上各式各樣的資訊，而網際服務(Web Service)的興起更是將網際網路的應用帶向另一個紀元。網際服務以 XML 作為異質平台與系統間文件交換溝通的標準格式，而其模組化的特性，使得軟體之間的整合能夠鬆散地連接，大為提高軟體可用性，經由網際服務技術，可以讓開發者整合現有的軟體系統，或是組合數個網際服務以建構更強大的服務。在網際服務技術的帶領下，Web 不再只是一個資訊平台，同時也扮演著一個服務平台的角色。

在另一方面，我們再來觀察另一個在網路市場中佔有舉足輕重地位的重要技術：OSGi [12]。OSGi (Open Services Gateway Initiative) 這個組織成立於一九九九年三月，其成立的目的是在於建立一個開放式的標準介面，讓遠端的服務提供者(Service Provider)與本地端的設備

之間建立一個完整的點對點傳送服務。OSGi 定義了一個開放性的平台，使得遠端軟體服務供應者所提供的應用程式及額外的加值服務，能根據每位使用者不同需求，隨時下載最新資訊至靠近使用者的閘道器(Gateway)上，並且自動安裝執行。而這裡所指的閘道器通常是連接區域網路(Local Network)與廣域網路(Wide Area Network)間的一個裝置，如機上盒(Set-Top Box)、ADSL 數據機、纜線數據機(Cable Modem)等。透過這個開放性的平台，不同廠商所開發出的服務軟體及設備都能互相溝通及搭配使用。

由上述介紹可知，網際服務著重於服務的提供，並以 Internet 作為其服務傳遞的管道。而 OSGi 則是在區域性網路(local network)下，遞送特殊需求的服務，如家庭服務與汽車服務等。兩種架構的目的各異，但在資訊服務技術中皆扮演重要的角色，然而目前卻沒有提供一種技術可以將 OSGi 服務和網際服務做一個有效的結合與應用，因此，本研究將以服務接合器(service adapter)來整合網際服務與 OSGi 服務，以產出更貼近使用者需求的各式服務。

### 二、相關研究

#### (一) OSGi (開放式服務閘道器標準)

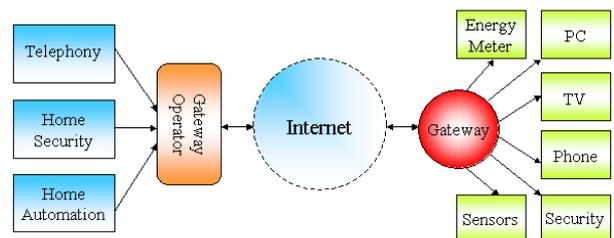


圖 1 OSGi 架構觀念圖

OSGi 標準制定主要目的在提供一個開放性平台，使得在遠端軟體服務供應商所提供的應用程式及加值服務軟體，能視使用者需求隨時下載至近端靠近用戶的閘道器上，接著自動安裝執行。為了達成這個目的，軟體服務供應商必須要根據 OSGi 所制定的規格去開發服務軟體，而靠近用戶端的閘道器也必須要將 OSGi 的功能整合至其中。

底下就 OSGi 標準中的重要概念(如圖 1) [5, 7, 10, 11]予以說明：

1. 服務提供者(Service Provider)：在 OSGi 下，服務(Service)的定義是可以在 OSGi 閘道(Gateway，簡稱為 OSG)下執行的應用程式(application)模組，可存取區域網路(local area network)的裝置與廣域網路(wide area network)上的遠端伺服器(remote back-end server)。這些服務將由閘道操作器所測試(testing)與驗證(verification)。

2. 閘道操作員(Gateway Operator)：閘道操作員主要負責維護與管理閘道以及 OSGi 服務。所負責的事務包含管理閘道、提供安全的廣域網路連結、控制閘道資源、控制存取權限、管理服務，以及遞送、起始、結束、更新與移除服務等。

3. 閘道(Gateway)：閘道是一個內嵌式(embedded)的伺服器，負責連結外部的網際網路以及內部的客戶端(client)，OSGi 閘道可跨越各種不相同裝置以及傳輸協定(如 Jini、UPnP、802.11、HAVi、Bluetooth 等)，而一個閘道本身可存在於數位視訊轉換盒(Set-Top Box)、纜線數據機(Cable Modem)、DSL 數據機(DSL modem)、個人電腦等裝置中，其特性相當適合作為家庭網路或汽車網路的基礎架構。

而 OSGi 的規格[9]主要分為三大部份：

1. 服務閘道器框架規格(Service Gateway Framework Spec.)：其定義的是服務閘道器之基本架構，任何設備只要根據此標準去實作出框架規格的功能，就可稱之為一個服務閘道器。框架規格是以 Java 虛擬機器(Virtual Machine)為基礎，以帶來可攜性與跨平台等好處。而根據框架規格所設計出的服務閘道器，不同廠商所開發的服務均可在其上執行。

2. 服務閘道器應用程式介面規格(Service Gateway API Spec.)：OSGi 服務是以 Bundle 形式呈現的 Java 應用程式，Bundle 是一個 Java Archive (JAR)壓縮格式的檔案，可動態安裝於此服務框架中。而此規格即定義了開發 Bundle 所應遵循的介面，不同廠商所開發出的 Bundle，只要依據此介面設計，就能在所

有的服務閘道器上執行。一個 Bundle 通常包含了

- 應用程式的可執行碼(executable code)。
- 應用程式所需的資源(resource)，如原生函式庫(native library)、影像檔、與 HTML 網頁等。
- 此應用程式的自身描述，包含文件、函式庫(native library)的可用性需求。
- 啟動類別(activator class)：經由此類別來啟動與終止此應用程式。

Bundle 在服務閘道器中的生命週期可分為下列四個狀態：已安裝、等待啟動、啟動、取消安裝。Bundle 要先經過安裝與啟動後才能在服務閘道器中執行，安裝之後的 Bundle，其狀態會自動進入等待啟動狀態，而要將 Bundle 從服務閘道器中移走，要經過取消安裝的步驟。

3. 裝置存取規格(Device Access Spec.)：其主要定義是「裝置自動偵測(Device Automatic Detection)」與「驅動程式 Bundle 下載(Device Driver Bundle Download)」兩個功能。當有新的設備連接至區域網路時，服務閘道器要有能力偵測並分辨出連接的設備種類，以自動下載正確的驅動程式。

## (二) 網際服務(Web Service)

網際服務可視為是以 Internet 或 Intranet 為基礎的模組化應用程式，可以自動地被一個或多個應用程式所啟動而完成特定的工作。而各大廠商，包括 IBM、Microsoft、Sun 等公司，為使各種不同平台、不同程式語言實作出之網際服務能夠彼此相容，陸續向 W3C 提出各種以 XML 為基礎的標準協定作為網際服務的溝通語言與溝通工具，這些協定包括了 SOAP、WSDL 與 UDDI。目前這些協定已逐漸被廣泛的接受與使用，也使得網際服務的應用日益普遍，包含電子商務(e-business)、電子化政府(e-government)、電子化學習(e-learning)等領域。

底下我們對網際服務相關的標準協定，分別描述如下：

- SOAP (Simple Object Access Protocol)：(SOAP) [3] 是一種公開標準，讓應用程式與應用程式能透過 Internet 相互溝通，但不需要知道彼此所在的作業平台，或是各自如何實作等細節資訊。而就技術的層面

而言，SOAP 是一種以 XML 為基礎的訊息交換機制，以 XML 的方式來達成文件訊息交換與遠端方法呼叫。

- WSDL (Web Services Description Language) [4]：WSDL 是一個以 XML 為基礎的網際服務描述語言，描述那些是可透過 internet/intranet 使用的應用程式？要用什麼資料格式與溝通協定才能與這些程式互動？它讓網際服務應用程式能以一種標準方法來描述自己擁有那些能力，以便讓互動更容易進行。而 WSDL 分為兩個部分：服務描述介面(Service Interface Definition) 與實作服務定義 (Service Implementation Definition)。
- UDDI (Universal Description, Discovery and Integration) [2]：UDDI 提供了實作一個服務登錄資料庫(Registry)所需的應用程式介面(APIs)。藉由此登錄資料庫，服務提供者能夠公佈(publish)他們提供的網際服務，而服務要求者可以經由登錄資料庫尋找(find)得到所需的網際服務。

而根據服務導向架構(Service-Oriented Architecture, SOA [6])所參與角色的觀點，我們可以定義出三個基本角色：服務提供者(Service Provider)、服務要求者(Service Requestor)、與服務登錄資料庫(registry)。

- 服務提供者：就商務的角度，服務提供者是網際服務的擁有者。而就技術的角度而言，服務提供者是一個使用服務的平台。
- 服務要求者：就商務的角度，服務要求者扮演的角色是向服務提供者要求服務。而就技術的角度而言，服務要求者可以由一個真實的人來扮演，經由瀏覽器來達成目的；而服務提供者也可以是一個程式，經由自動執行方式來達成要求。
- 服務登錄資料庫：提供一個目錄服務讓服務提供者可以公佈其網際服務至目錄上，而讓服務要求者可以藉由此服務尋找到適當的網際服務。

這些角色的運作必須建立於上述三個標

準協定上(如圖 2)，其中服務登錄資料庫是以 UDDI 來實作，而任兩方之間的溝通都是透過 SOAP 訊息機制來交換資訊。底下我們進一步解釋任兩方之間的操作。

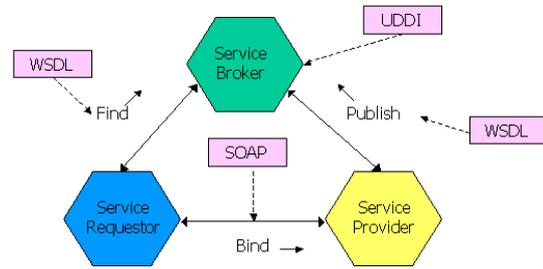


圖 2：服務角色與標準協定關係圖

- 服務提供者對服務登錄資料庫：服務提供者負責提供網際服務讓別人來使用。藉由使用 WSDL，服務提供者可描述在這項服務中所含的功能，要使用此功能所需要的輸入資料，以及預期的輸出結果，並提供 URL 連到這支服務程式以便可以真正去呼叫它。最後，服務提供者透過 UDDI 把這些資訊公告(publish)至服務登錄資料庫。
- 服務要求者對服務登錄資料庫：服務要求者透過 SOAP 訊息處理，服務要求者可以送出格式化的查尋字串(find)給服務登錄資料庫。而服務登錄資料庫會將查詢結果傳回給服務要求者。
- 服務要求者對服務提供者：服務要求者已得到適當的 WSDL 檔，即可將需要的輸入資料，透過 WSDL 提供的 URL 位址，啟動(bind)該服務提供者所提供的網際服務。

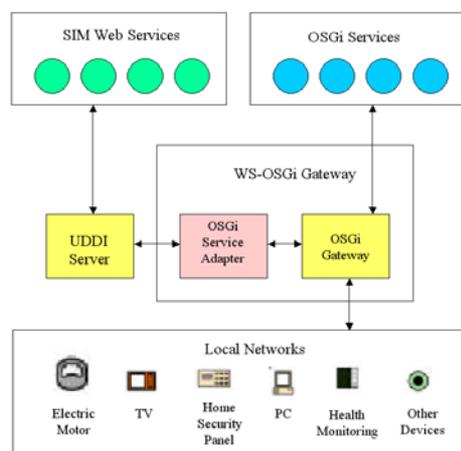


圖 3：OSGi 服務接合器架構圖

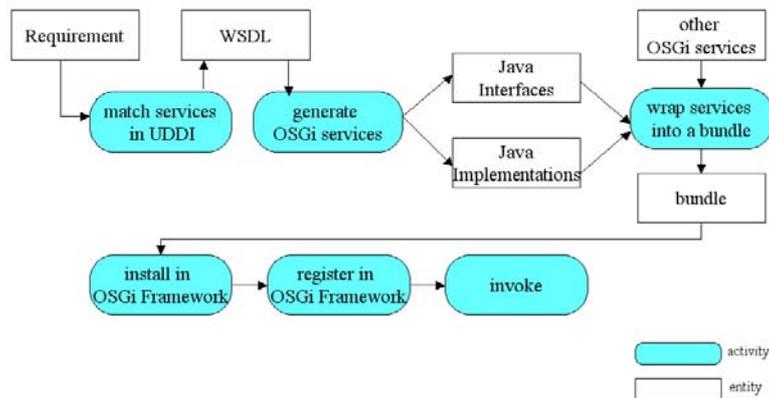


圖 4：網際服務轉換為 OSGi 服務

### 三、網際服務與 OSGi 服務的整合機制

#### (一) 服務接合器(Service Adapter)

在本研究中，我們主要目標是為達成網際服務與 OSGi 服務的整合，我們認為可以透過一個服務接合器(service adapter)來達成(如圖 3)。此接合器主要要達成的功能包含兩個部分，一是將 OSGi 服務轉換為網際服務，讓使用者可以透過網際網路存取 OSGi 服務；二是將網際服務轉換為 OSGi 服務，讓服務在區域網路(Local Area Network，如汽車網路、家庭網路等)下使用各式各樣的網際服務。

#### 網際服務轉換成 OSGi 服務

OSGi 架構下的服務為一個 JAR 格式的檔案(Bundle)，內部包含應用程式的可執行碼(executable code)、所需的資源(resource)、此應用程式的自身描述(manifest)、以及如何啟動與終止此應用程式的方式等。因此可看出 OSGi 服務的運作方式與網際服務的運作方式大異其趣，因此，若要將一般的網際服務轉換為 OSGi 服務，我們認為應該包含以下的程序(如圖 4)：

- 1 取得網際服務：開發者可在 UDDI 伺服器根據需求進行服務配對(match services)，並取得符合需求的 WSDL 文件，進一步進入產生 OSGi 服務的步驟。
- 2 建立 OSGi 服務所需要的檔案：此轉接器提供使用者可以藉由一份 WSDL 文件，進一步產生出 interface、implementation、activator、manifest、complex type 檔案。
  - 2.1 產生 OSGi 服務介面(interface)：取得所需的 WSDL 文件後，我們提

供一個可產出 Java interface 的自動介面產生器，利用 WSDL4J [13] 及 WSIF [14]此兩項工具剖悉相對應的 WSDL 文件，擷取所需要的資訊，根據這些資訊產生出一個 OSGi 服務的 java interface。此 interface 所呈現出來的 operation 可一一對應到 WSDL 所描述的 operation。

- 2.2 產生 OSGi 服務實作(implementation)：建立出服務介面後，我們以一個可產出 Java implementation 的服務實作產生器，讓開發者可以根據取得的 WSDL 文件進一步產生出此網際服務的 Proxy，Proxy 內包含了上述介面的 implementation 及 OSGi 服務的相關功能。
- 2.3 產生 OSGi 服務驅動器(activator)：建立出服務實作介面後，我們將產生一個驅動此 OSGi 服務的驅動程式，稱之為 activator。透過此驅動程式的協助，OSGi Framework 可管理相對應的 OSGi 服務。在本研究中，我們採用 Oscar [8] (Open Service Container Architecture) 作為底層的架構，其中 Oscar 為一套遵循 OSGi Framework 規格的實作，目前之版本已符合 OSGi 2.0 大部分之功能。
- 2.4 產生 OSGi 服務說明文件(manifest)：建立出服務驅動程式後，我們將產出 Jar manifest，作為此 OSGi 服務的相對應說明文件。在此文件中除了說明此 OSGi 服務的一些相關資訊外，並詳細記載了此 OSGi 服務驅動程式的所在，以

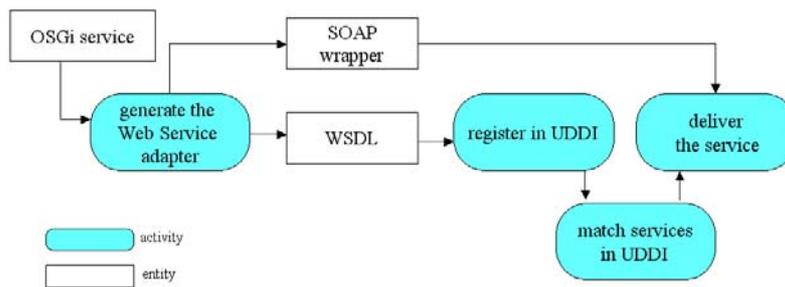


圖 5：OSGi 服務轉換為網際服務

確保 OSGi Framework 能明確的載入相對應的 OSGi 服務驅動程式。

- 3 編譯相關檔案：當 OSGi 服務的相關檔案建立完成後，我們將其動態編譯成.class 檔案。
- 4 包裝 OSGi 服務：將以上完成的 Java 實作與 Java 介面之.class 檔為輸入，並加入其它相關的元件，以建立可達成目標需求的 OSGi 服務。我們以一個 Bundler Constructor，將相關的資訊(包含服務間的依存關係)依據 OSGi 規格，包裝成為一個 OSGi Bundle。
- 5 安裝(install)、註冊(register)：在包裝完成此 OSGi 服務後，依據使用者需求安裝並註冊其 service 至 OSGi 框架(OSGi framework)上，完成部署工作。
- 6 呼叫(involve)服務：OSGi 服務一但註冊完畢，此服務即可被使用者透過 LDAP 等目錄服務來尋找、使用此服務。

### OSGi 服務轉換成網際服務

在網際服務轉換為 OSGi 服務方面，我們規劃了下面的方法，讓一個 OSGi 服務能夠轉換成一個網際服務，讓使用者藉由網際服務存取此服務，其程序如下(如圖 5)：

- 1 產生網際服務相關檔案：此轉接器提供使用者可以藉由一個 OSGi 服務，進一步產生出 soap service。
  - 1.1 產生網際服務：在給定一個 OSGi 服務後，此轉接器提供使用者可以藉由一個 JAR 剖析器擷取出 OSGi 服務內的重要資訊，並根據此資訊產生一個 SOAP 服務。此 SOAP 服務可視為 OSGi 服務的 wrapper，透過以 RMI 為底層之 Remote Controller 來存取 OSGi 服務。

1.2 產生 WSDD (Web Service Deployment Descriptor)文件：產生網際服務後，我們以一個能產生 WSDD 文件的產生器建立出與此網際服務相對應的 WSDD 文件。

- 2 部署網際服務：在此，我們利用 Axis [1] Soap Server 當作存放網際服務的伺服器，藉由以上產生出來的 WSDD 文件，將此網際服務部署至 Axis 上，以提供遠端使用者使用。
- 3 產生 WSDL 文件：在網際服務的部署至 Axis 上之後，我們利用 Axis 所提供的工具建立出與此網際服務相對的 WSDL 文件，並將其註冊至 UDDI 上。使用者在取得此 WSDL 文件後，可以根據此文件的資訊撰寫此網際服務的呼叫程式。
- 4 配對與遞送網際服務：當網際服務以註冊完畢，即可利用服務導向架構(Service-Oriented Architecture, SOA)來進行後續的動作。

### (二) 案例探討 (Case Study)

我們提出一個虛擬的劇本(如圖 6)：偷竊追蹤的電子化服務。在此複合服務(composite service)中，需要數個網際服務與 OSGi 服務共同合作才能達成目標，我們可利用這個例子來突顯本論文所提出之整合機制的重要性。而在此範例中，其服務運作可分為下列幾個步驟：

1. 當車主發現車輛失竊後，使用者可透過網際網路叫用個人資訊網際服務(Personal Information Service)，經由此服務來呼叫警察網際服務(Police Service)。
2. 警察服務將會先叫用車輛追蹤網際服務，由車輛追蹤服務來使用監理站網際服務(MVDIS Service)，以查詢車輛是否被拖吊，若無被拖吊，則將透過開道器

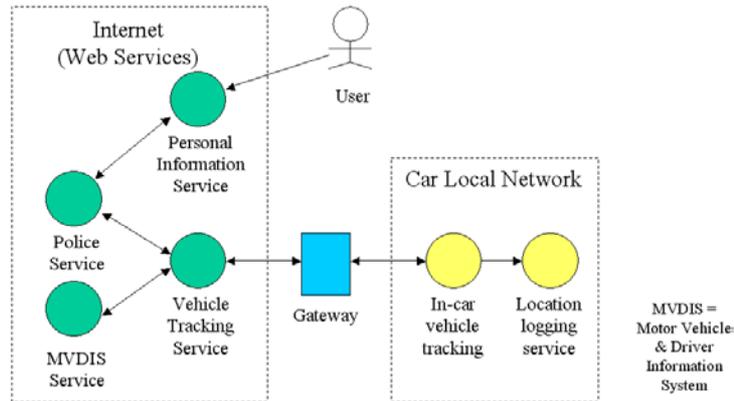


圖 6：虛擬劇本：偷竊預防服務

(Gateway)呼叫車輛內的追蹤服務(In-car vehicle tracking)。

3. 車輛內的追蹤服務(屬於 OSGi 服務)會呼叫定位記錄服務(Location logging service)來查詢目前汽車所在位置，並將此資訊透過閘道器回傳給車輛追蹤網際服務，並進一步回傳給警察網際服務。
4. 警察服務將根據回傳結果進行竊案處理，並將處理結果回報給個人資訊網際服務讓車主了解目前情況。

#### 四、結論

本論文提出一個整合機制，透過服務接合器(adapter)來整合網際服務(Web Service)與 OSGi 服務。此接合器可將後端的網際服務包裝為 OSGi 服務，讓網際服務作為後端服務提供機制，OSGi 標準作為前端服務遞送機制，或將 OSGi 服務轉包為網際服務，以提供遠端存取。經由本文所提出的整合機制，可結合網際服務與 OSGi 技術，進一步擴大電子化服務的應用範圍。

#### 五、致謝

本研究由經濟部委託(或補助)財團法人資訊工業策進會航空資料鏈路通訊系統技術研發四年計畫分包辦理。

#### 六、參考文獻

- [1] Axis Homepage: <http://ws.apache.org/axis/>
- [2] T. Bellwood, L. Clement, and D. Ehnebuske,

et al, "UDDI version 3.0 published specification", W3C, 19 July 2002. Available at <http://uddi.org/pubs/uddi-v3.00-published-20020719.htm>

- [3] D. E. D. Box, G.Kakivaya, A. Layman, H. F. N. N. Mendelsohn, S. Thatte, and D. Winter, "Simple Object Access Protocol (SOAP) 1.1", W3C, 2000. Available at <http://www.w3c.org/TR/SOAP>
- [4] R. Chinnici, M. Gudgin, J. J. Moreau, and S. Weerawarana, "Web Service Description Language (WSDL) version 1.2 w3c working draft". W3C, 9 July 2002. Available at <http://www.w3.org/TR/2002/WD-wsdl12-20020709/>
- [5] T. Honkanen, "OSGi – Open Services Gateway initiative".
- [6] H. Kreger, "Web Services Conceptual Architecture (WSCA) 1.0", IBM Software Group, May 2001.
- [7] D. Marples and P. Kriens, "The Open Services Gateway Initiative – An Introductory Overview".
- [8] Oscar Homepage: <http://oscar-osgi.sourceforge.net/>
- [9] OSGi, "OSGi Service Gateway Specification, release 2.0", October, 2000. Available at <http://www.osgi.org>
- [10] OSGi, "The last mile of software deployment", 2002.
- [11] OSGi, "The OSGi – Java connection", 2002.
- [12] OSGi Homepage: <http://www.osgi.org>
- [13] WSDL4J Homepage: <http://ws.apache.org/wsif/>
- [14] WSIF Homepage: <http://www-124.ibm.com/developerworks/projects/wsdl4j/>