

Value-added SIP-based VoIP Service

陳俊明

葉哲華

吳坤熹

國立交通大學

國立交通大學

國立交通大學

資訊工程學系

資訊工程學系

資訊工程學系

jinming@csie.nctu.edu.tw

chyei@csie.nctu.edu.tw

solomon@csie.nctu.edu.tw

摘要

VoIP 的優點除了可節省傳統電話的長途通話費用、降低成本之外，更在於它能夠利用 IP 網路的開放式架構，整合多種通訊方式，提供傳統電話無法提供的服務。本文中將以 SIP 為基礎，提出「可由第三方提供的語音信箱服務」與「整合式通信中心」兩項 VoIP 加值服務的新架構。

關鍵詞：ENUM, SIP, UMS, VoIP

一、前言

在 IP (Internet Protocol) 的應用上，IETF (The Internet Engineering Task Force) 在 RFC 3261 文件中所提出的 SIP (Session Initiation Protocol) [15]，是繼 H.323 之後，被公認在 VoIP (Voice over IP) 的信令 (signaling) 及通話控制 (call control) 上，極具潛力的一個通訊協定。相較於 SIP，ITU-T (the telecommunication standardisation sector of International Telecommunications Union) 所制定的標準 H.323 協定過於複雜、擴充性低、規則修改繁雜，因此要在其中開發新的服務或增加新的元件實在是相當不易。SIP 的其中一個優勢在於其擴充性，採用以文字為基礎的標頭 (header) 格式；它的格式類似 HTTP (Hypertext Transfer Protocol)，易於擴充。因此已被 3GPP (Third Generation Partnership Project) 採用作為其 IMS (IP Multimedia Subsystem) 的信令協定。

儘管 SIP 現今已被大量使用為 VoIP 的通話協定，但是 SIP 設計的本意並不只單純為了通話服務，有會議 (session) 性質的服務皆可以使用 SIP 作為協定。因此 SIP 不但用在 VoIP 系統中，像是 Instant Message Service (即時訊息服務) [2]，以及 Presence Service [6]，也可以利用 SIP 擴充出來的定義來架構。下面我們要介紹的兩個系統，可由第三方提供的語音信箱服務 (Third-party Provided

Voicemail Service) 和整合式通信中心 (iNotify Center)，都是以 SIP 為基礎的服務。

二、可由第三方提供的語音信箱服務

一般所熟知的語音信箱服務 (Voicemail Service)，是在使用者無法接聽來電時由 Voicemail Server (語音信箱伺服器) 代接，並錄下留言。過去語音信箱服務的作法，以行動電話網路的架構為例，必須由交換機處理，額外設定轉接到語音信箱。當交換設備被選定的同時，語音信箱服務也就受限於此設備提供的功能，使用者無法有自行選擇及擴充的自由。

SIP 的架構下的 Proxy Server (代理伺服器) 如同傳統的交換設備，可提供受話者通訊位址的轉接。然而 Proxy Server 的功能不僅如此，如善用 SIP 協定的彈性及擴充性，語音信箱服務可輕易加入 ITSP (Internet Telephony Service Provider) 已建置的 SIP Proxy Server。以下我們將提出一個架構，允許第三方 (third-party) 單獨開發更完善的語音信箱服務，讓使用者可自行與其他 Proxy Server 搭配。

(一) SIP 多重呼叫機制

因應一個電話號碼可以在不同電話機接聽的需求，SIP 提供一個便利的多重呼叫機制 (forking)。以下介紹一般 SIP 的註冊及撥打流程。如圖 2.1，Bob 的兩支電話設定為同一 SIP Public Address，啟動後會向 Proxy Server 註冊 (送出 REGISTER 訊息)，告知個別的 SIP Contact Address。註冊的結果使得 Registrar 存放 Bob 的 Public Address 與其兩支分機的 Contact Address。

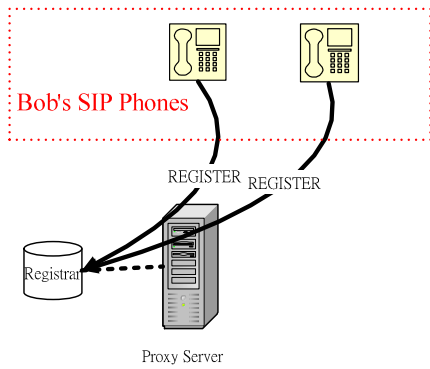


圖 2.1 Bob 兩支分機向 SIP Proxy 註冊其 Public Address

今若有使用者 Alice 欲與 Bob 交談，參照圖 2.2，Alice 在撥號後傳送 INVITE 訊息(1)，內含 Bob 的 Public Address。Proxy Server 以 Bob 的 Public Address 為索引，向 Registrar 查詢。Registrar 回傳兩個 Contact Address，分別是先前 Bob 註冊的兩個位址。故 Proxy Server 會將 INVITE 分送兩處(2a、2b)，此時兩支電話同時響起。(為求清晰起見，圖中省略 100 Trying 與 180 Ringing 等回應訊息)。

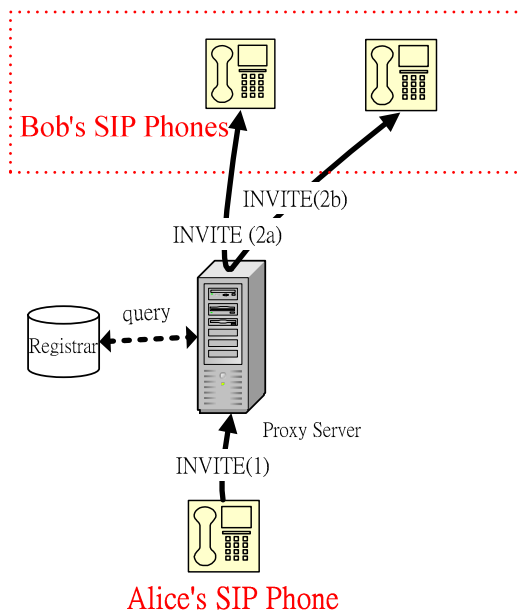


圖 2.2 Alice 呼叫 Bob，訊息經 Proxy Server 被 fork 到兩處

習慣上同號碼的兩支電話響起後，若接起其中一支，另一支則停止震鈴。SIP forking 也做得到同樣的效果。參照圖 2.3，若 A 電話被接起來，回傳 200 OK (3)，Proxy Server 一方面送出 CANCEL 到 B (4)，一方面轉送 200 OK 訊息至 Alice (5)。B 電話收到 CANCEL 便停止鈴聲，而 Alice 再送 ACK 至 A 電話，便可建立語音連線了。

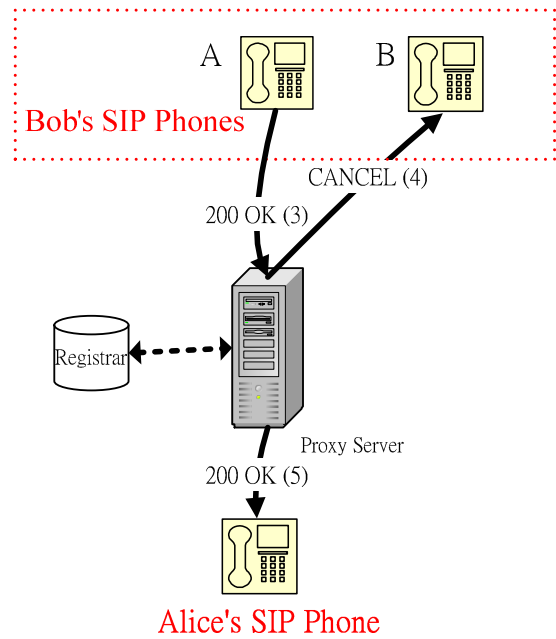


圖 2.3 Bob 接起 A 話機，B 話機停止鈴聲

以上就是所謂的 forking 機制。在這個架構下，Bob 如果另外設置第三台 SIP phone，只需由第三支電話再發送註冊(Register)訊息，ITSP 業者不必在 Proxy Server 上進行特別的設定調整。

(二) 語音信箱留言的信令流程

應用以上提到 SIP forking 的架構，Voicemail Server 亦可模擬 Bob 的 SIP Phone 的動作，向 Proxy Server 註冊。我們修改上個例子，將 Bob 的第二支分機，以 Voicemail Server 取代。請參照圖 2.4，Voicemail Server 亦使用 Bob 的 Public Address 向 Proxy Server 註冊，並將 Contact Address 指向 Voicemail Server 接收 SIP 訊息的入口。

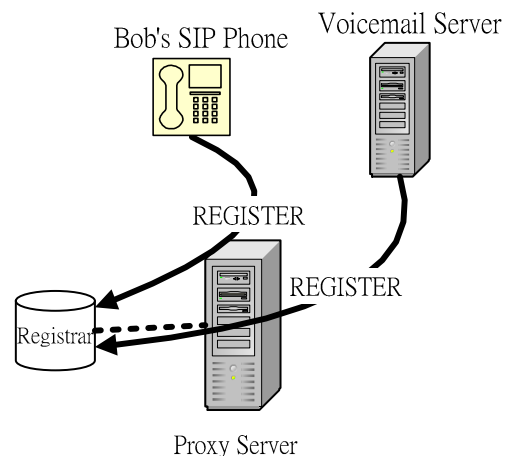


圖 2.4 Voicemail 註冊與 Bob 相同的 Public Address

同樣地，如圖 2.5，使用者 Alice 傳送 INVITE (1) 訊息至 Proxy Server 時，Proxy Server 由 Registrar

取得兩個 Contact Address，並將 INVITE 分送 2a、2b 兩處 (包括 Voicemail Server)。

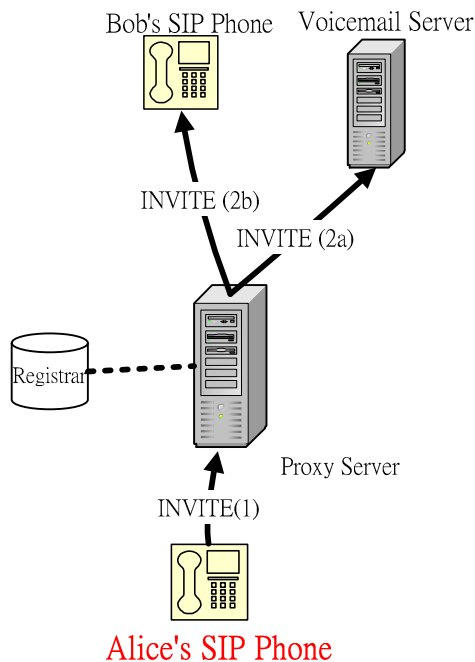


圖 2.5 Alice 呼叫 Bob，訊息經 Proxy Server 被 fork 到兩處，包括 Voicemail Server

Voicemail Server 在內部有計時器 (pick-up timer)，在等待固定秒數後如未收到 CANCEL，即回傳 200 OK 啟動通話。如同我們熟悉的留言過程，Voicemail Server 可先傳送一段歡迎辭給 Alice，接著啟動錄音，達成語音留言功能。詳細的訊息流程請參考圖 2.6。

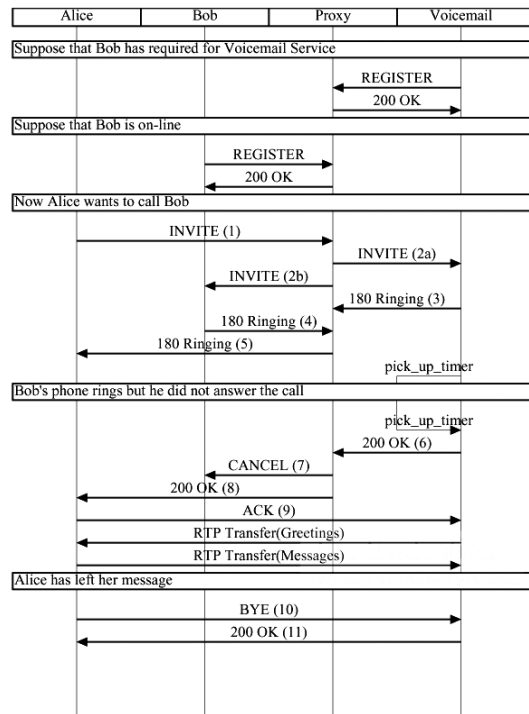


圖 2.6 流程說明如下：(此圖省略 100 Trying 之回應訊息)

- (1) Alice 發出 INVITE 到 SIP Proxy
- (2) SIP Proxy 同時送 INVITE 到 Voicemail Server(2a) 與 Bob 的 SIP Proxy(2b)
- (3) Bob 之 SIP phone 回應 180 Ringing，鈴聲響起
- (4) Voicemail Server 回應 180 Ringing，同時啟動計時器(Pick-up Timer)等待 Bob 先接起電話
- (5) Proxy Server 收到 Ringing，轉送回 Alice，Alice 因而得到等待回應的訊息
- (6)待計時器 Time out 後，Voicemail Server 便會接起電話。其做法同樣為先回應 200 OK 給 Proxy Server。Proxy Server 會送出一個 CANCEL(7)給 Bob，同時送 200 OK 給 Alice(8)。此時 Bob 的話機停止響鈴
- (9) Alice 發送 ACK 表示收到，現在可與 Voicemail Server 通話了。一般留言的動作是先聽到歡迎詞 (Greetings)，接著再開始留言。通話使用之協定為 RTP (Real-Time Protocol) [16]
- (10) Alice 留言完畢掛下話筒，送出 BYE 給 Voicemail Server，接著 Voicemail Server 停止錄音，回傳 200 OK(11)。另外 Voicemail Server 需額外的計時器避免 Alice 講話時間過久，此時便由 Voicemail Server 送 BYE 訊息給 Alice。

(三) 優點分析

目前致力於研發 SIP 通訊產品的廠商包括 Cisco [3]、Pingtel [13]、snom [17]，皆已實作出 SIP phone 與 SIP Proxy Server/Registrar，並包括語音信箱功能。但這些廠商提供的語音信箱功能並非獨立的產品，而被列為 Proxy Server 的外加功能。兩者的關係可以圖 2.7 表示。

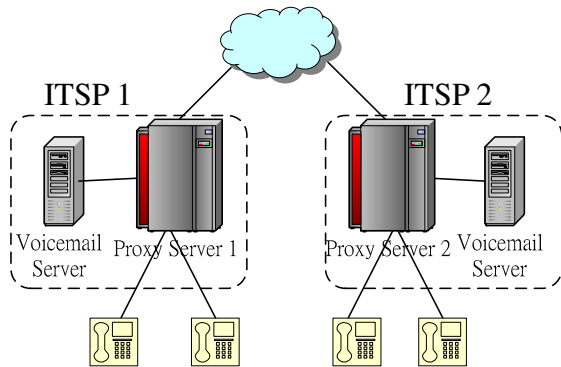


圖 2.7 目前廠商提供語音信箱服務的方式是與 SIP Proxy Server/Registrar 一同售出

各家提供的留言服務功能各異，如果使用者喜好 A 公司的 Proxy Server，但 B 公司的語音信箱服務較符合需求，則兩者都不能滿足此使用者。若利用 SIP 的擴充性，新的架構可以如圖 2.8，語音信箱服務提供者可開發獨具特色的語音信箱服務供使用者選擇，使用者不需要向 Proxy Server 提供者申請語音信箱，基本上只需將其 Public Address 告知語音信箱服務提供者，就可享有此服務。

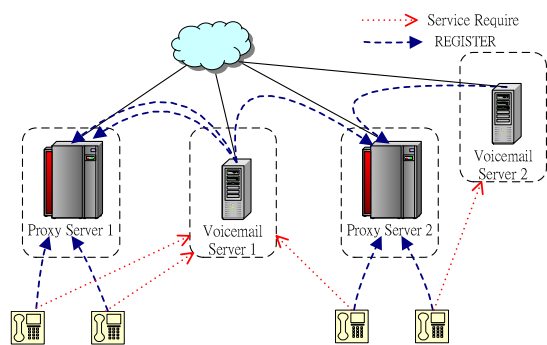


圖 2.8 不同廠商提供的 Proxy Server 可與他家廠商的 Voicemail Server 整合

(四) 語音信箱伺服器實作範例

Voicemail Server 端需要一長駐程式負責 SIP 訊息的收送及 RTP 的接收與存檔。由於其程式流程不同於一般 SIP phone，需提供多使用者同時連入、在收到 INVITE 後需等待使用者無法接起電話時才回應 200 OK，且不需將語音送至喇叭撥放，故使用一般單人使用且與語音整合的 SIP library

並不適合。開放原始碼 Josua (Jack's Open SIP User Agent) v0.9.2 為一包含 SIP signaling 而不包括語音傳輸的 SIP User Agent (代理人) 程式，內部使用 GNU oSIP library [10]。Josua 並提供很大的修改彈性，於程式碼中保留各 SIP 事件的 Callback Function，可輕易地增加個別需求的處理程序，因此很適合於其上開發 Voicemail Server。

由研究到實作 Voicemail Server，我們花費約半年的時間，目前已可註冊至指定的 Proxy Server，並將留言語音存成 WAV 格式，使用者可以自行由網頁以 HTTP 存取語音留言。而通知使用者取得新留言的部分，我們已規劃更多元化的形式，先以 SIP Instant Message 送至目前開發中的 iNotify Center，iNotify Center 可選擇 Instant Message、手機簡訊、或 E-mail 等方式，以當時最佳的方式通知使用者。細節會在下一章介紹。

三、整合式通信中心

UMS (Unified Message System, or Universal Message Service) [7] 是針對現在快速增加各種訊息服務而形成的概念。我們經常用的的訊息服務，像是 Internet 上的 E-mail、Instant Message (即時訊息)、手機使用者常用的簡訊服務 (SMS) 等，使用者必須透過不同的介面來收個別的訊息；也就是手機簡訊只能透過手機、E-mail 只能透過收信軟體來接收。而 UMS 的概念就是將這些在不同介面上的服務加以整合，讓送訊者及收訊者都可以透過單一介面就可以收送所有的訊息。

我們提出的 整合式通信中心 — iNotify Center，就是基於 UMS 的概念所發展出來提供給訊息發送者的一項服務。iNotify Center 相當於各種不同訊息服務之間的 gateway。對於信息的發送者來說，iNotify Center 就是一個訊息中心，不論是透過何種介面，發送者都是將訊息交由 iNotify Center 處理，接著由 iNotify Center 來判斷，利用最有效率的方式來送給收訊者。(例如：發送者手邊只有手機，而收訊者在電腦前，此時 iNotify Center 可能就會考慮將從發送者收到的簡訊以 Instant Message 的方式送給收訊者)。

(一) 相關產品及服務

ICQ.Com [11] 提供了類似 UMS 的服務。ICQ.Com 提供 "The ICQ Unified Messaging Center" 供使用者來使用，這是一個網頁介面的訊息中心，透過這個網頁，使用者可以送訊息到 ICQ、送出手機簡訊、以及送 E-mail。對使用者來說，只要透過網頁單一介面，就可以發送許多不同介面上的訊息服務。

Concord Technologies [5] 提供了 "Universal

Mailbox”，讓使用者可以將所有的訊息，不論是語音訊息或者是 Fax，都可以由這個信箱來接收。使用者只需要透過這個 Mailbox，便可以存取不同介面的訊息。

(二) iNotify Center

本節所介紹的 iNotify Center，是一個以 SIP 協定為基礎的通訊服務。當送訊者有訊息要通知接受者時，可以將訊息以 Instant Message、手機簡訊、E-mail 等各種方式，傳送到 iNotify Center。iNotify Center 則根據收訊者的上線資訊及其預設之喜好，選擇最佳方式傳送給收訊者。如此，送訊者就不必煩惱該在眾多的通訊方式中選擇何者來通知接受者，而可以將此一工作留給 iNotify Center 來負責。

如圖 3.1 所示，iNotify Center 包含幾個部分：

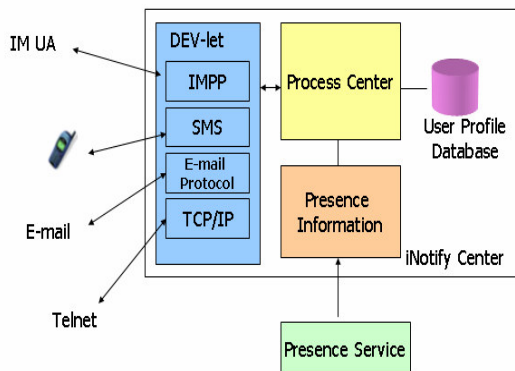


圖 3.1 iNotify Center 的架構

1. DEV-let

DEV-let 是用來控制各個介面的裝置，也就是提供各種介面給使用者來使用，同時也利用定義好的介面來和 Process Center 來溝通。DEV-let 接受發訊者所送出的文字訊息，並且發送由 Process Center 處理過的訊息給收訊者。就如圖 3.2 所示例子，DEV-let 可以提供：IMPP (Instant Messaging and Presence Protocol)、SMS (Short Message Service)、SMTP (Simple Mail Transfer Protocol) 和 TCP/IP。IMPP DEV-let 提供的是 Instant Message 的通訊協定，可以接受 Instant Message 要求 (request) 以及發送 Instant Message 給接受者，而目前我們採用的 Instant Message 通訊協定為 SIP。SMS 是提供 GSM 網路上的簡訊服務 [1]。E-mail DEV-let 則是透過 SMTP 來接受及發送 E-mail。TCP/IP DEV-let 則是可以透過 socket 連線傳送訊息。Process Center 負責管理各個 DEV-let 並且透過 DEV-let 和各個裝置互動。

當 iNotify Center 啟動時，DEV-let 便負責開始接受外來的要求，經由 Process Center 處理過後，

再由特定的 DEV-let 將訊息傳送出去。例如發送 Instant Message 到 iNotify Center，再由 iNotify Center 發送簡訊到收訊者手機。

2. Process Center

Process Center 負責管理 DEV-let 以及作資料的處理。DEV-let 在接受到發訊者所傳資料後，會送到 Process Center 來處理。此時 Process Center 會將收到的資料先轉換成 SIP MESSAGE 格式，接著會查詢 Presence Information (使用者的上線狀況，下一節會詳述)，根據收訊者現在的狀況來決定以何種介面傳送訊息給收訊者。Process Center 將 SIP 格式的訊息轉換成相對應的格式，然後會將資料送到 DEV-let 來發送。

3. Presence Information

Presence Information 是由 Presence Service 所提供的資訊。Presence Service 的應用，以最為廣為人知的 Microsoft Windows Messenger 來做例子。Windows Messenger 在加入其他使用者到好友名單上後，便可以看見這些使用者的上線資訊：像是線上、離線、忙碌、離開等等 (如圖 3.2)。



圖 3.2 Microsoft Windows Messenger

Presence Service 的架構在 RFC 2778 [8]、RFC 2779 [6] 中定義如圖 3.3 所示：

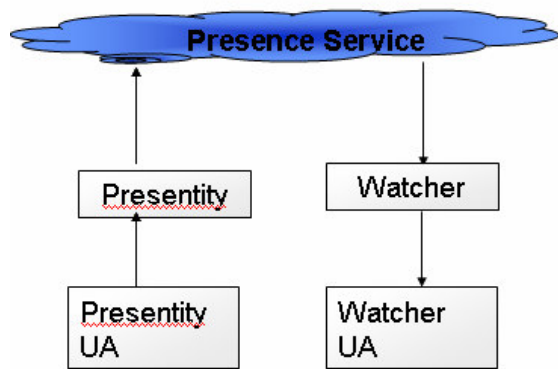


圖 3.3 Presence Service Model

其中 Presentity 是提供 Presence Information 的來源，而 Watcher 是接受 Presence Information 資訊。舉個例子來說，原本 Presentity 是處於離線的狀態，此時 Watcher 從 Presence Service 得到 Presentity 的 Presence Information 便是「離線」。而 Presentity 上線後，會發 Presence Information—「線上」之類的資訊給 Presence Service，而 Presence Service 會再將這資訊傳給 Watcher，此時 Watcher 便知道 Presentity 在線上。

我們採用 SIP 為 Presence Service 的通訊協定。SIP 在 RFC 3265 “Session Initiation Protocol (SIP)-Specific Event Notification” [14] 中定義了兩種 METHOD—SUBSCRIBE 和 NOTIFY，可以利用在 Presence Service 的搭建上面。

在 iNotify Center 的架構中，我們採用 IPTel.org 的 SIP Express Router 提供 Presence Service，將收訊者的 Presence Information 提供給 Process Center，來判斷以何種方式傳送給收訊者最為有效率。例如今天有封訊息要送給收訊者，而根據 Presence Service 所提供的資訊，如果收訊者正在線上，那 Process Center 可以考慮利用 Instant Message 的方式，直接透過 Internet 傳送給收訊者；如果發現收訊者並不在線上，Process Center 便會依據使用者預設的方式(手機簡訊或者 E-mail) 傳送給收訊者。

4. User Profile Database

User Profile Database 儲存使用者相關資訊，例如：E-mail Address、行動電話號碼之類的資訊。在 Process Center 要將訊息轉送給收訊者的時候，便可由這裡查詢收訊者的相關資料。除了使用者的基本資料外，也可以提供一些基本的控制權來給使用者設定，例如指定將訊息一律轉送到 E-mail，不論使用者是否在线上；或者何種狀況下應該送至某處之類的設定，這都是 User Profile Database 會儲存的資訊。

(三) 信息流程範例

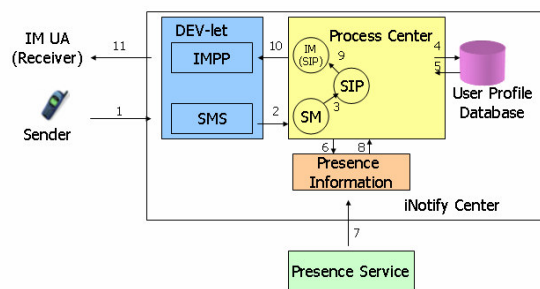


圖 3.4 Flow Example

(1) 發訊端以手機簡訊送訊息到 iNotify Center，由 DEV-let 的 SMS 介面負責接收。

(2) DEV-let 收到簡訊之後交由 Process Center 來做處理。

(3) Process Center 將此封簡訊轉換成標準格式(SIP MESSAGE)。

(4) Process Center 向 User Profile Database 查詢有關收件者的基本資料。

(5) User Profile Data 傳回收訊者的 SIP URI (Uniform Resource Identifier)。

(6)~(8) Process Center 根據 SIP URI 查詢收件者的 Presence Information，假設回傳時確認收訊者在线上。

(9) 將被轉換成 SIP 標準格式的訊息轉換成 Instant Message 通訊協定對應的格式 (iNotify Center 中的 Instant Message 通訊協定內定為 SIP)

(10) Process Center 將被轉換過後的 Instant Message 送至 DEV-let。

(11) DEV-let 送出 Instant Message 至 Internet 上，送給收訊者。

(四) 優點分析

iNotify Center 的理念，是針對多種不同介面的訊息服務加以整合。其中訂定 SIP 為標準格式的目的，第一、為了不同介面訊息之間的轉換，不需要將每個一對一轉換都實做出來，否則一旦多了一種新介面，就必須針對每個既有的介面去做對新介面做一對一轉換的實作。第二、使用 SIP 的原因，因為在格式定義上，SIP 在 headers (標頭) 上的定義足夠涵蓋該有的資訊，像是 From、To、Subject 等 header。加上 SIP 是內容採用純文字編碼，因此在字串處理上較為簡單方便。第三、SIP 本身就可以當作 Instant Message 的通訊協定 [2]，因此我們

採用 SIP 為各個訊息之間的標準格式。

iNotify Center 可以和其他系統加以整合，使用其“Notify”的功能。例如前半部分介紹的語音信箱服務，在使用者有了新留言後，Voicemail Server 便可以發送通知訊息給 iNotify Center，讓 iNotify Center 以最有效率的方式通知使用者。此外，iNotify Center 另一個重要的應用實例，是和網管系統合作，當網管系統發現網路出現一些狀況，需要即時通知網管人員，卻又不確定人在何處。此時便可以發一個訊息到 iNotify Center，讓 iNotify Center 利用最合適的方式即時通知網管人員。如果網管人員正在線上，就可以收到 Instant Message，進行即時的處理。若是網管人員不在線上，則可根據其預設的條件，將事件依優先程序，判別應以簡訊或 E-mail 通知網管人員，讓網管人員能做出適當的處置。傳統上，這類功能都被綁死在特定的網管系統中。利用 SIP 的開放式架構，我們所設計的整合式通信服務，可以輕易地整合至管理者自行開發的管理工具中。

四、結論

本篇文章我們提出了以 SIP 為基礎，可由第三方提供的語音信箱與整合式通信中心兩服務。

語音信箱服務提供者能獨立開發系統，供使用者選用，並能輕易與任何其他開發者開發的 SIP Proxy Server 整合。在實際應用上為了安全性的考量，VoIP 使用者在註冊到 SIP Proxy 的同時，是需要經過認證的。SIP 目前的認證方式與 HTTP 的認證機制相同，是採取 RFC 2617 [9] 的使用者身份認證機制。然而此認證方式下雙方擁有相同的金鑰，今使用者如申請第三方語音信箱服務提供者的服務，必須給提供者相同的金鑰內容，將更容易發生密碼外洩的可能。是否能利用 Federated Authentication [4] 的概念，在保護使用者密碼的前提下，又能由第三方註冊上 Proxy Server，實為第三方語音信箱服務實際應用上一個重要的課題。

目前的 iNotify Center，主要是針對送件者可以透過單一介面發送各種訊息，可是收件者依然還是必須透過對應的介面來接受訊息。未來可以加上整合式訊息信箱 (Unified Message Box) [7] 的概念，讓訊息接受者也可以只透過單一的介面來處理各種不同類型的訊息服務。

由於 iNotify Center 是一個開放的架構，未來也可以在此架構下加上更多種不同的訊息服務，提供更多樣的功能給使用者。像是整合多媒體訊息 (MMS)，讓使用者可以傳送及接收多媒體訊息。

此外，目前使用者的各項聯絡資訊，像是 E-mail address 和手機號碼，是儲存在 iNotify

Center 自己的 User Profile Database 中。未來更理想的應用方式，則是透過 ENUM [12] 機制，利用 E.164 number 至 DNS 查詢。如何有效結合 ENUM 與 VoIP 加值型服務，是未來在學術研究和商業應用上都相當熱門的主題。

五、誌謝

本專題成果由國科會計畫 NSC 91-2219-E-009-053 贊助。

六、參考文獻

- [1] 林風, 饒仲華, 林一平, 呂明坤, 塗冠驊, 皇甫建君 “Short Message Service Programming for Wireless Internet” :維科圖書, October 2002
- [2] B. Campbell, Ed., J. Rosenberg, H. Schulzrinne, C. Huitema, D. Gurle, “Session Initiation Protocol (SIP) Extension for Instant Messaging,” RFC 3428, December 2002
- [3] Cisco Corp. “Cisco SIP Proxy Server”, <http://www.cisco.com/univercd/cc/td/doc/product/voice/sipproxy/index.htm>
- [4] Clickshare Service Corp. “Federated Authentication – Identity Management”, <http://www.clickshare.com/>
- [5] Concord Technologies, <http://www.concordfax.com/>
- [6] M. Day, S. Aggarwal, G. Mohr, J. Vincent, “Instant Messaging / Presence Protocol Requirements,” RFC 2779, February 2000
- [7] David A. Zimmer, “Universal Message Services”, <http://www.csie.nctu.edu.tw/~chyei/voip6/ums4/ums4.html>
- [8] M. Day, J. Rosenberg, H. Sugano, “A Model for Presence and Instant Messaging,” RFC 2778, February 2000
- [9] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, L. Stewart, “HTTP Authentication: Basic and Digest Access Authentication”, RFC 2617, June 1999
- [10] GNU Project, “oSIP Library”, <http://www.fsf.org/software/osip/osip.html>
- [11] ICQ.Com “Unified Messaging Center,” <http://web.icq.com/wwp/default/0,,00.html>
- [12] J. Klensin, “The History and Context of Telephone Number Mapping (ENUM) Operational Decisions: Informational Documents

Contributed to ITU-T Study Group 2 (SG2)",
RFC 3245, March 2002

- [13] Pingtel Corp., "SIPxchange IP PBX,"
http://www.pingtel.com/pr_sipx_overview.jsp
- [14] A. B. Roach , "Session Initiation Protocol
(SIP)-Specific Event Notification," RFC 3265 ,
June 2002
- [15] J. Rosenberg, H. Schulzrinne, G. Camarillo, A.
Johnston, J. Peterson, R. Sparks, M. Handley, E.
Schooler , "SIP: Session Initiation Protocol,"
RFC 3261 , June 2002
- [16] H. Schulzrinne, S. Casner, R. Frederick, V.
Jacobson, "RTP: A Transport Protocol for
Real-Time Applications," RFC 1889, January
1996
- [17] snom technology AG , <http://www.snom.com/>