

在 SIP-based Push-to-Talk 通訊服務的通訊群組管理機制

許政穆 梁瑞志 連韋賓

國立中正大學資訊工程學系

{hsujm;ljc95m;lwp95m}@cs.ccu.edu.tw

摘要

多人群組通訊可分為多人同時講話與同時只限一人講話的通訊模式，對於多人同時講話稱為群組會議(Multiparty Conferencing)，而同時只允許一人講話稱為隨按即說(Push-to-Talk)。隨按即說的群組通訊通常是用於特定通訊用途，如計程車通訊、警查勤務通訊等。由於隨按即說是以群組作為通訊的對象，因此群組管理機制就顯得非常重要。本論文我們提出了一個遵循 OMA 提出的 Push-to-Talk over Cellular (簡稱 PoC) 規範所建置的 SIP Push-to-Talk 的群組管理機制，讓在 All-IP 網路上的 SIP Push-to-Talk 群組通訊能建立屬於自己的 Push-to-Talk 群組管理清單，並依群組清單進行合法的 Push-to-Talk 群組通訊。

關鍵詞：SIP, Push-to-Talk (PTT)，通訊群組管理。

Abstract

The Group communication service can be categorized into two parts: one is multiparty conferencing and another is one-way Push-to-Talk (PTT) service. The PTT service has been widely used in taxi communication and police services via traditional radio services. Nevertheless, the communicating target of the PTT is a specific group using a half-duplex way to communicate each other. Thus, a well-designed mechanism of the PTT group management is very important. In this paper, we have developed a SIP-based PTT service, which follows a PTT standard, push-to-talk over cellular (PoC) service standard proposed by the OMA. Our developed SIP-based PTT service can help PTT users to create and maintain their own PTT group lists, and make some PTT sessions based their organized group lists.

Keywords: SIP, Push-to-Talk (PTT), Group Management.

1. 前言

在多人的群組通訊模式中，隨按即說(Push-to-Talk, 簡稱 PTT)[2]是我們日常生活中很常見的服務，其功能就像對講機一對多的單向通訊。由於隨按即說有其通訊的特殊性與便利性，因此在電信網路與電腦網路上也都有隨按即說的群組通訊服務建置，而行動通信網路上的隨按即說的群組通訊服務則被稱為 Push-to-Talk over Cellular(簡稱 PoC) [6]。

由於 Push-to-Talk 是一種半雙工單向的群組通訊服務，跟以往傳統電話、手機等一對一全雙工雙工的通訊方式不太一樣。因 PTT 適用於同時時間內跟一群人進行通話，但是同一時間之內只能允許一位群組成員進行發話，而其他參與此 PTT 會談的成員都變為聆聽者。而圖 1 為一個以 PTT 方式作群組通訊的示意圖，其中 PTT 是利用發言權機制管控參與會談者的發言權利，當擁有發言權的會談者便可講話。但發言完畢或經過一段發言時間後，發言權便會釋放出來或被撤銷。所以透過此機制便能維持一對多且單向的 PTT 群組通訊特性。由於 PTT 群組通訊同時時間只允許一個參與者發言，假若同時間內有多位參與者請求發言權，就必須由 PTT 管控機制決定誰獲得發言權，其他請求人還是必須等待，通常對發言權的競爭是採取先到先得服務模式。

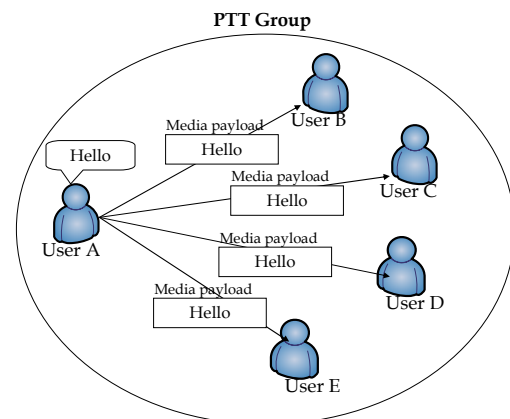


圖 1 PTT 群組通訊示意圖

由於 PTT 在傳統無線電通訊是相當常見的，但在 All-IP 網路則需要考量 PTT 群組通訊的建立與多媒體語音視訊內容的傳遞，因此在 All-IP 網路上對 PTT 群組的設計與建置研究，大多都採用 Session Initiation Protocol (SIP) [9]作為 PTT 服務中的協商信令，而語音封包則是使用 Real-time Transport Protocol (RTP)。此外 PTT 群組通訊是一個群組通訊與會談的概念，初期必沒有統一的標準規範，造成 PTT 在互通上的困難。因此 Open Mobile Alliance (OMA) 定義了許多 PoC 標準規範後，眾多研究所提出的 PTT 架構與運作模式皆已遵循 OMA 的 PoC 架構[12]。

由於傳統無線電對講機只能提供通話功能，而運行在 All-IP 網路上的 PTT 群組通訊服務是以 SIP 信令作為 PTT 群組通訊控制，因此能在 SIP 信令夾帶額外資訊，如使用者位址、使用者喜好，使用者裝置能力、以及使用者狀態資訊等，因此 PTT 群組

通訊也能整合其它服務如狀態呈現服務(Presence Service)[1]。

在 PTT 群組通訊中如何建立 PTT 通訊群組成員與群組成員管理是一項重要議題，因為通訊群組(Contact List)建立與群組通訊的權限管控(Access Control)，都必需受合宜的機制加以管理，才能確保 PTT 群組通訊的正常運作。然而在 OMA 所定義的 PoC 規範中，明確指出 Group List Management Server (GLMS)負責管理使用者所定義的群組清單，而 PTT 伺服器會根據使用者定義的群組清單建立群組通訊會談。雖然在[5]中，也同樣遵循 OMA PoC 的 GLMS 觀念，但在處理 GLMS 的群組信令是以 SIP MESSAGE 而不是 OMA 所使用的 XML Configuration Access Protocol (XCAP)[8]。本論文同樣是設計一套 SIP-based Push-to-Talk 群組通訊服務，但在 PTT 的群組管理則是遵循 OMA PoC 的 GLMS 管理機制，包括 PoC 的 XML 資料格式與使用 XCAP 作為群組資訊傳送等。透過此 OMA 的 PoC 群組管理機制，未來在 PTT 系統擴充與延伸上都有一定的程度的互通性，特別是在未來全面的 ALL-IP 網路上。

在本論文架構中，在第二節會探討 OMA 所提出的 PoC 架構稍作描述。而在第三節針對本論文所設計的 PTT 群組管理機制、架構、與 SIP 信令詳加討論，第四節則說明相關機制的實作，並提出一個 PTT 使用情境說明我們設計的 PTT 群組通訊在群組管理的運作方式。最後在第五節將作一總結說明。

2. 文獻探討

為了了解 OMA 所訂定的 PoC 群組通訊運作方式，首先必需對 OMA 的 PoC 群組通訊架構大略說明，而圖 2 為 OMA 所提出的 PoC 架構[6,12]，而 OMA PoC 架構內的元件功能說明，如圖 2 所示。

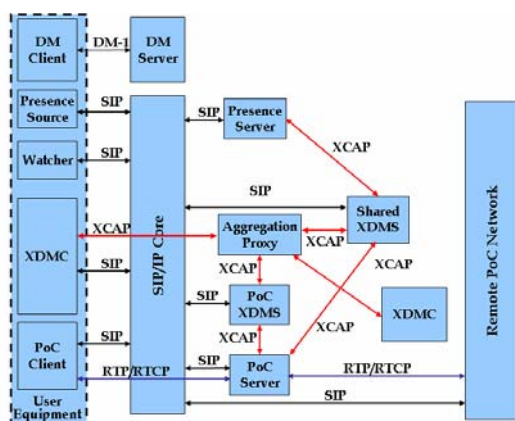


圖 2 OMA 的 PoC 架構

在 OMA PoC 架構中，SIP/IP Core 模組代表 SIP 環境中的代理伺服器(SIP Proxy)或註冊伺服器(SIP Registrar)，主要提供 SIP 使用者端註冊、認證、與 SIP 信令轉送。PoC XDMS (PoC XML Document

Management Server)與 Shared XMDs 是負責管理儲存使用者的群組通訊清單等資訊，而 Aggregation Proxy 是用來整合群組清單與使用者現狀狀態資訊呈現。

在 OMA PoC 群組管理方面，使用者端可以透過 XDMC (XML Document Management Client) 來進行群組通訊的管控，而這些群組資訊的管理皆是透過 XCAP (XML Configuration Access Protocol) 存取控制協定來傳輸存取的。對於專門負責管控儲存使用者所的群組通訊清單等資訊的伺服器，通常也被稱為 GLMS (Group List Management Server)。

由於 OMA PoC 主要是應用在行動通訊環境下，故在 User Equipment 端包括了 Presence Source 與 Watcher 作為提供使用者狀態資訊與訂閱狀態資訊，而 DM (Device Management) Client 主要是讓使用者可以透過遠端傳輸方式與 DM Server 互動，對嵌入式設備(如手機)進行相關更新、管理、控制與資料處理備份等工作。同樣 User Equipment 端的 XDMC 為 XML Document 管理的對應模組，讓使用者用來管理自己的群組通訊清單。

此外 PoC Server 負責使用者的 PTT 群組會談與會談發言權控制，而 OMA PoC 架構中發言權的控制是被包含在 RTCP (RTP Control Protocol) 中，所以該模組會同時間處理信令與語音部份，在處理程序上就顯得相對複雜。

3. SIP-based PTT 群組通訊系統架構

在 OMA PoC 中的 PTT 通訊模式有分 Dial-out Groups 與 Join-in Group，而 PTT 會議(Session)建立則分 On-Demand 與 Pre-established Sessions 兩種 [12]。為了讓 PTT 的群組通訊比較偏向使用者的使用習慣，在本論文是以 Dial-out Group 的通訊模式，並配合 On-Demand 的會議建立方式為主，而其相關的 PTT 流程運作，將於後續加以詳述。圖 3 為本論文所採用的 PTT 群組通訊的系統架構，其包括了 PTT Server、GLMS、RTP Relay Server。

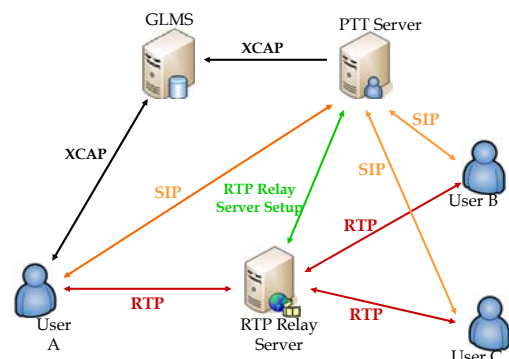


圖 3 PTT 系統架構圖

在 SIP 通訊架構中，當要建立一個通話時，需要知道對方的 URI 以便發出一個 INVITE 的請求，才能建立通話。同理在 Dial-out Group 模式的 PTT

群組通訊模式中，PTT 伺服器也必須知道 PTT 群組通訊中每位使用者 URI 才能進行 INVITE 發送。然而 PTT Server 要如何知道使用者所定義的群組通訊資訊，因此需要擁有一個 PTT 群組管理模組管理。由於 OMA PoC 提出是以 XML 描述群組通訊內的 Group List 資訊，並為每一群組定訂一個獨特的 URI 方便 PTT Server 取得群組資訊。這些以 XML 描述的 PTT 群組通訊的成員清單最終是儲存於 GLMS 內。為了簡化每位使用者對其所擁有的 PTT 群組通訊清單管理，我們將每一位使用者群組清單設定為單一檔案。而這些以 XML 描述的 PTT 群組通訊清單則是以 XCAP 協定在 PTT 使用者端、GLMS、PTT Server 之間的 PTT 通訊群組清單的傳遞。

由於 XCAP 協定是架構於 HTTP 協定上，對於 XML 文件儲存方式是利用網頁樹狀結構來進行儲存。而 XML 文件存放位置的 URI 會包含一個 AUID (Application Unique ID)，其作用是用來辨識不同服務型態，有可能是現狀狀態資訊、好友清單等。依據 OMA PoC 所提出的 AUID (Application Unique ID)，其定義為 org.openmobilealliance.poc-groups，並且會加上使用者的 URI，便可用來辨識不同使用者的群組清單。因此在 PTT 使用者的群組資訊儲存，就如同圖 4 中所列 URI 表示：

```
http://net11.cs.ccu.edu.tw/xcap/services/AUID/users/USER_URI/group_list.xml
AUID : org.openmobilealliance.poc-groups
USER_URI : sip:user1@net11.cs.ccu.edu.tw
```

圖 4 XCAP 的 URI 表示

由於 XCAP 協定是在 HTTP 上所延伸，用來存取位於 GLMS 上的 XML 檔案共有 PUT、GET、DELETE 等三個指令，提供使用者新增、存取、刪除自己的群組清單。

```
<?xml version="1.0" encoding="iso-8859-1"?>
<group xmlns="urn:oma:params:xml:ns:list-service">
  <list-service uri="sip:Alice_group_Lab@net11.cs.ccu.edu.tw"
    display-name="Lab">
    <allow_anonymity>false</allow_anonymity>
    <list>
      <entry uri="sip:Tom@net11.cs.ccu.edu.tw"
        description="Tom" />
      <entry uri="sip:John@net11.cs.ccu.edu.tw"
        description="John" />
    </list>
  </list-service>
  <list-service uri="sip:Alice_group_mana@net11.cs.ccu.edu.tw"
    display-name="mana">
    <allow_anonymity>true</allow_anonymity>
    <list>
      <entry uri="sip:Bob@net11.cs.ccu.edu.tw"
        description="" />
      <entry uri="sip:Mary@net11.cs.ccu.edu.tw"
        description="" />
    </list>
  </list-service>
</group>
```

圖 5 群組管理清單

對於 GLMS 內所儲存的群組管理清單，其格式

如圖 5 所示。由於每位 PTT 使用者可以建立多個 PTT 群組通訊清單，因此主要是以<list-service>標籤來區別使用者所建立的每一個群組，並且標明一個唯一的群組 URI。此外 PTT Server 也會以<allow_anonymity>標籤作為判別在進行 PTT 群組會談時是否允許不屬於群組清單中的成員加入。其目的是為了限制 PTT 群組會談的私密性，當不希望本身的 PTT 會談給外人參與時，將可以設置此標籤來拒絕非群組內的成員加入。此外<list>標籤則是用來標識相同群組名稱的群組成員，而每一<entry>是記錄每位使用者的 URI 與相關得資訊描述。

所以當 PTT 使用者上線並對 PTT Server 註冊後，就會利用 XCAP 的 GET 從 GLMS 上取回 PTT 群組通訊清單，供 PTT 使用者選擇要進行 PTT 群組通訊之用。當然 PTT 使用者亦有權利修改 PTT 群組通訊的清單內容，便可使用 XCAP 的 PUT 將修改群組清單上傳至 GLMS 上，以取代舊有的 PTT 群組通訊清單。而 PTT 使用者若要 Dial-out 建立一個 On-Demand 的 PTT 會談，便會將儲存於 XML 檔案的 PTT 群組通訊 URI 透過 INVITE 送給 PTT Server，而 PTT Server 便會產生一連串的 INVITE 將此 On-Demand 的 PTT 會談予建立，並通知 RTP Relay Server 開啟相關的 Relay Port 進行相關的 PTT 會談時的 RTP 語音封包轉送處理。

本論文所採用以 Dial-out 與 On-Demand 形式的 PTT 群組會談，當 PTT 群組會談發起者使用 SIP INVITE 訊息邀請 PTT 群組成員進行會談，也許會有部份 PTT 群組因當下某些因素無法回應接受會談，而允許這些 PTT 成員在某些條件下能加入先前已建立的 PTT 群組會談。因此 PTT Server 必須紀錄每一個 PTT 群組會談，才能輔助 PTT 使用者找到相關已進行的 PTT 群組會談。例如圖 6 中使用者 A 經由 PTT Server 取得目前正在進行的 PTT 群組通訊：Group 1 與 Group 2，其中 Group 1 允許 User A 加入 PTT 會談的話，User A 便可加入該會談；而 Group 2 為私人 PTT 會談並不允許其它不屬此群組清單成員的人加入，故 Group 2 會拒絕 User A 的事後加入。

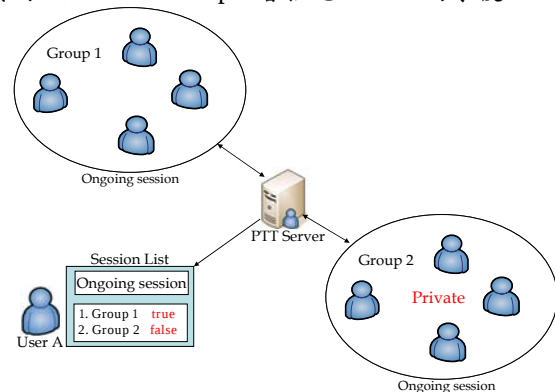


圖 6 PTT Server 維護進行中的 PTT 群組會談

為了讓 PTT 使用者取得目前進行的 PTT 群組會談資訊，PTT Server 使用 SIP MESSAGE 信令夾帶 PTT 會談資訊給 PTT 使用者，而其 SIP

MESSAGE 訊息格式如圖 7 中所示。在圖 7 中的 SIP MESSAGE 信令是以<ongoing_session>標記表示此訊息夾帶會談訊息，而內層的<list>標籤所包含的<session name>標記則為說明 PTT 群組的 URI，其後面跟隨的<allow_anonymity>標記則是依據群組建立者所規範是否允許非群組成員加入。

```
MESSAGE sip:callot@net11.cs.ccu.edu.tw SIP/2.0
Via: SIP/2.0/UDP 140.123.105.108;branch=z9hG4bK3A9B977A
CSeq: 6290 MESSAGE
To: <sip:callot@net11.cs.ccu.edu.tw>
Accept-Contact: *,+g.poc.talkburst;require;explicit
Content-Type: text/plain;charset=UTF-8
From: <sip:net11.cs.ccu.edu.tw>;tag=6BCBF8B3
Call-ID: 622449533@140.123.105.108
Content-Length: 72
User-Agent: kphone/4.2
Contact:<sip:net11.cs.ccu.edu.tw;transport=udp>

<?xml version="1.0" encoding="iso-8859-1"?>
<ongoing_session>
  <list>
    <session name = "alice_group_manana@net11.cs.ccu.edu.tw"
      allow_anonymity = "false"/>
    <session name = "ray_group_family@net11.cs.ccu.edu.tw"
      allow_anonymity = "true"/>
  </list>
</ongoing_session>
```

圖 7 以 SIP MESSAGE 夾帶進行通話的 PTT 群組資訊

雖然 OMA PoC 已在 RTCP 封包內訂定了 PTT 會談發言權的控制機制，然而 RTCP 封包內的 PTT 群組發言控制的處理，無法立即反應給 PTT Server，因此為了讓 PTT 發言控制能快速明確受到 PTT Server 管控，因此我們借助 SIP 會談中的 SIP INFO Message [3]作為夾帶 PTT 發言權控制之用。由於 PTT 群組通訊服務的特性是在同時時間內只允許一人能發話，因此 PTT Server 必須維持每位參與通話的使用者狀態，不能同時讓兩位 PTT 使用者同時擁有開會談的發言權，所以 PTT Server 在 SIP INFO 信令中夾帶必須夾帶 PTT 使用者的發言狀態通知該會談的參與者。

在 OMA PoC 中所定的發言權管控總共有十多種，包括 TB_Request、TB_Granted、TB_Taken、TB_Deny、TB_Release、TB_Idle、TB_Revoke、TB_Ack、TB_Position、TB_Queued、TB_Disconnect、TB_Connect 等，其中最主要 6 個 PTT 會談的發言權控制如表 1 中所列，而本論文所使用的 PTT 會談發言權控制是以此 6 個類型為主。

表 1 PTT 使用者發言權取得狀態之顯示

類型	說明
Granted	獲得發言權
Taken	只有聆聽的權利
Request	要求取得發言權
Release	釋放發言權
Revoke	PTT 伺服器主動撤銷發言權
Idle	目前 PTT 會談沒有人取得發言權

對於 SIP INFO 訊息內所會夾帶會談參與者的資訊如圖 8 中所示，其中<token>代表自己目前狀態，<speaker>則代表目前擁有發言權的使用者，最後在<list>之中夾帶的則是目前參與群組通話的每位使用者的資訊。

```
INFO sip:Bob@140.123.105.126;transport=udp SIP/2.0
Via: SIP/2.0/UDP 140.123.105.111;branch=z9hG4bKd124.7262a761.0
To: "Bob" <sip:Bob@net11.cs.ccu.edu.tw>;tag=1F452764
From: <sip:Bob_group_manana@net11.cs.ccu.edu.tw>;tag=a6a1c5f60fa
CSeq: 1 INFO
Call-ID: 1622065351@140.123.105.126
Content-Length: 111
User-Agent: Sip EXpress router(0.9.4 (i386/linux))

<?xml version="1.0" encoding="iso-8859-1"?>
<ptt_tbcap>
  <token>Granted</token>
  <speaker img="Bob@net11.cs.ccu.edu.tw.png">Bob</speaker>
  <list>
    <member name="Bob" token="Granted"/>
    <member name="Mary" token="Taken"/>
    <member name="John" token="Taken"/>
  </list>
</ptt_tbcap>
```

圖 8 以 SIP INFO 訊息封裝 PTT 發言權之範例

4. SIP-based PTT 群組通訊與 GLMS 的群組管理機制實作

對於本論文所用的 PTT 群組通訊架構中，我們主要參考[10]的實作模式，以在 Linux (Fedora Core 4)的作業系統上，使用 Apache 2.0 加上 PHP 撰寫動態網頁，並結合資料庫功能來完成 GLMS 的群組管理機制上 XCAP 協定功能，以達到群組清單 XML 檔案的基本存取。

此 GLMS 管理機制是以資料庫方式來儲存群組清單資訊，其優點在於當使用者新增一個群組時，系統可以方便與現有的群組 URI 核對，查詢是否會造成重複命名的群組 URI。而 XCAP 的通訊協定是架構於 HTTP/1.1 協定之上，因此我們可以利用現成的網頁伺服器來支援，此部份我們使用 Apache 2.0 網頁伺服器，並利用 Apache 中所擁有的 Rewrite 模組，將 XCAP 協定操作的 URI 導向到一個專門處理 XCAP 請求的動態網頁。此動態網頁乃是用 PHP 所撰寫而成，並用來解析請求的 URI 並判斷此請求為 PUT、GET、DELETE 哪一種。對於群組管理系統的請求，其 AUID 則設定為 org.openmobilealliance.poc-groups 以符合 OMA PoC 所定的群組管理的前序(prefix)。而請求的 URI 剩下部份就可以取出使用者的 URI 資訊，以決定此 XCAP 的請求來自哪一位使用者，以便存取出相對應的資訊。

在判斷完 AUID 與使用者 URI 資訊後，GLMS 將會處理 HTTP 所挾帶的群組資訊。其群組資訊中可分為群組 URI 與群組中的使用者 URI，因此在將 PTT 群組通訊清單在儲存進資料庫之前，會先利用 PHP 所提供的 XML 文件解析器功能取出群組

URI，與資料庫中所儲存的 URI 進行比對，若是沒有重複則可將此清單存進資料庫中。

在資料庫方面，我們採用了 PHP 所支援的 MySQL 資料庫來儲存資訊。在 GLMS 中的資料庫內建立了兩個資料表，分別為“group_uri”與“group_data”，如表 2 所示。其中“group_uri”主要用來儲存使用者與群組 URI 的對應之用，而建立此資料表目的主要用來快速比對資料庫中現存的群組 URI 是否衝突，以及提供 PTT 伺服器查詢此“group_uri”的擁有者為誰；而“group_data”資料表則主要是儲存使用者的 PTT 群組通訊清單的 XML 檔案之用。

表 2 GLMS 所用的資料表內容

資料表：group_uri	
欄位名稱	描述
user_uri	使用者 URI
group_uri	使用者所定義 PTT 群組通訊的 URI
資料表：group_data	
欄位名稱	描述
user_uri	使用者 URI
file_name	使用者上傳所指定的檔案名稱，在系統中統一為“group_list.xml”
xcap_data	主要儲存使用者群組資訊(XML 檔案內容)

在本論文所建置的 PTT Server 主要負責 PTT 會談建立與維持，並且負責維持 PTT 發言權一致性的管控。在此我們是以開放原始碼的 SIP Express Router (SER)[11]作為 PTT Server 的基幹。SER 本身設計為一個開放式的架構，允許系統開發者可以依照自己的需求修改程式碼來建立各式各樣的加值服務，甚至可以結合 SER 本身所提供模組加以應用，用以建置出新的功能模組，因而我們在現有的 SER 伺服器上面，開發 PTT 模組，用以進行 PTT 通訊與發言權的管理。

在本論文所擴充 SER 的 PTT 模組(PTT Server)主要可以分為三個主要功能：提供 PTT 的會談資訊、PTT 會談的建立，以及發言權的控制。當使用者上線完成註冊程序後，PTT Server 便會定期發送 MESSAGE 信令給 PTT 使用者，告知使用者目前 PTT Server 上正在進行 PTT 會談的群組資訊，以便 PTT 使用者加入到目前正在進行的會談。

當 PTT Server 在收到一個建立 PTT 會談請求(SIP INVITE)時，PTT Server 會先判斷此建立會談請求，是為第一次請求還是使用者想要加入一個進行中的會談，若是為新建立 PTT 會談，則 PTT Server 會利用 XCAP 協定從 GLMS 上取回使用者的群組資訊以建立 PTT 會談；並利用 SIP MESSAGE 信令更新 PTT 會談相關訊息。對於正在進行的 PTT 會談，PTT 會談發起者或其它已參與會談成員可以使用 SIP REFER 信令邀請其他人加入目前的會談。

對於 PTT 會談的發言權控制，就如先前說提到是參考 SIP INFO Message [3]作為發言權的管控，故

在 RTP Relay Server 上面並不會真正的抑止 RTP 封包轉送。然而在 OMA PoC 的發言權管控機制是交至 RTCP 封包內，因此 RTP Relay Server 會負責發言權的款管控處理。此一模式 RTP Relay Server 雖能快速有效處理 RTP 封包轉送與發言權控制的同步處理，但沒法讓 PTT Server 快速了解目前 PTT 會談的發言已經交給那位 PTT 使用者，較無法快速呈現 PTT 會談的立即性。

本論文所設計的 PTT Server 對於多人同時請求同一 PTT 會談的發言權是採取先到先服務的機制，並且會設定每位使用者握有發言權的時間，以免某位使用者不正常離線或發言過久，而造成發言權無法釋出與交予其它 PTT 參與者使用的情況發生。

為了說明我們設計的 SIP-based PTT 群組通訊與 GLMS 管理機制的運作可行性，我們將會以圖 9 內的流程說明一個一對多單向的 PTT 會談建立過程與相關的處理，而 SIP-based PTT Client 則以[4]中所提出的支援圖片示的 SIP-based PTT Client 為主。

在圖 9 中共有三個 User Agents 分別為 UA1、UA2、與 UA3，其中當 UA1 上線註冊完成(1)時，PTT Server 會送出一個 SIP MESSAGE 信令給 UA1 告知目前 PTT 會談狀況(2)，並且利用 XCAP 協定向 GLMS 取回 UA1 的 PTT 群組通訊清單(3)，並將群組資訊在使用者畫面上顯示出來(4)。接著 UA1 便可藉由群組清單邀請 UA2 來建立一個 PTT 會談(5)，當 UA1 與 UA2 的 PTT 會談成功建立時，系統將會發送 MESSAGE 來更新所有使用者的進行中會談資訊(6)。

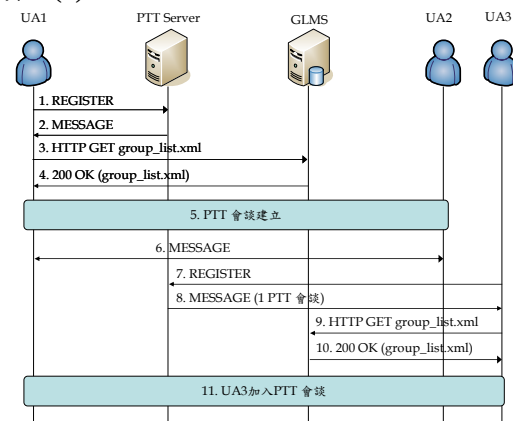


圖 9 SIP-based PTT 會談建立與維護的運作流程

接著 UA3 上線並且完成註冊(7)，此使用者端所顯示的畫面將會如圖 10 中的 SIP-based PTT Client 畫面顯示。在圖 10 的右半邊是 SER 本身所提供的狀態呈現服務，而左上方顯示的則是現在系統有那些 PTT 群組正在進行 PTT 會談通訊資訊，而左下方顯示則是使用者的群組資訊，讓使用者可以選擇加入現有的 PTT 會談，或是建立一個新的 PTT 會談。而圖 11 為在 PTT 使用者端新增新的群

組與編輯相關成員資訊畫面，經由此使用者介面設定新的 PTT 群組清單資訊後，便能更新到 GLMS 上。

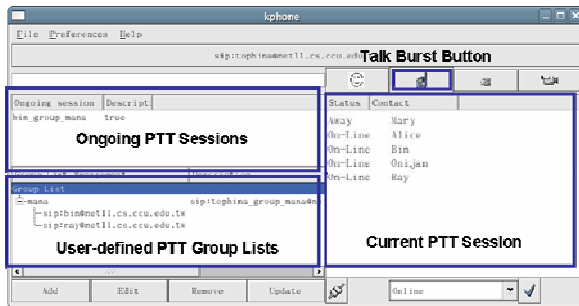
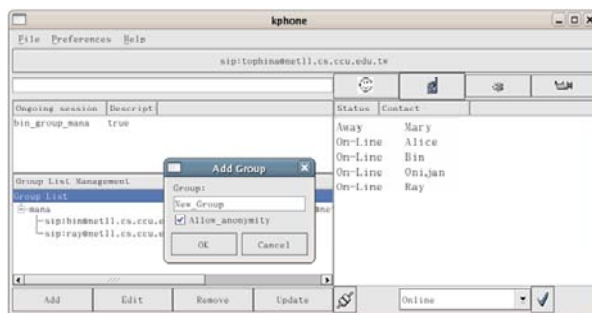
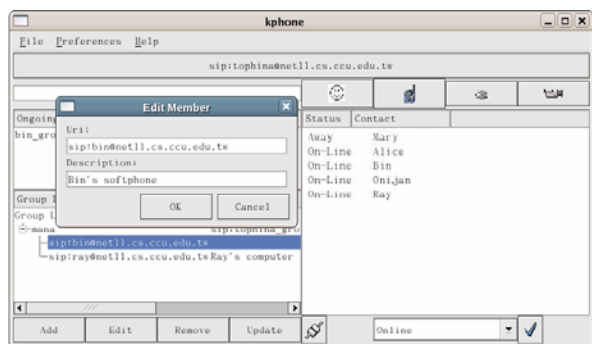


圖 10 PTT Client 呈現 PTT Session 資訊畫面



(a)加入新的使用者自訂的 PTT 群組通訊清單



(b)編輯 PTT 群組清單的成員 URI

圖 11 PTT Client 新增與編輯 PTT 群組通訊清單

5. 結論

本論文主要是提出一個遵循 OMA PoC 標準實作出的 SIP-based PTT 群組通訊服務，包括使用 GLMS 與 XCAP 作為群組管理與管理資訊的存取協定之用。在我們提出的 PTT 群組通訊架構支援 Dial-out 與 On-Demand 形式的 PTT 會談建立與管理，讓 PTT 使用者能依照自己的偏好設定 PTT 的群組通訊清單，以 SIP INVITE 方式邀請 PTT 使用者參與 PTT 的會談。而此 SIP-based PTT 群組通訊服務可應用於 All-IP 網路上，能與現存的 Internet 與 Telecom 有很好的互通性。

此外本論文所實作的 GLMS 管理機制也能整

合情境感知資訊並提出對應服務，如 Presence 整合，甚至也能達成情境過濾(Context Filter)的情境感知應用。如此一來，能讓 PTT 群組通訊與 GLMS 管理機制，能與其它 SIP-based 的應用服務整合，而提供更多樣化的 PTT 群組通訊服務。

參考文獻

- [1] N. Blum and T. Magedanz, "PTT + IMS = PTM - towards community/presence-based IMS multimedia services," in Proceedings of the 7th IEEE International Symposium on Multimedia, 2005.
- [2] L. A. DaSilva, G. E. Morgan, C. W. Bostian, D. G. Sweeney, S.F. Midkiff, J. H. Reed, C. Thompson, "The resurgence of push-to-talk technologies," IEEE Communications Magazine, VOL. 44, NO. 1, pp. 48-55, 2006.
- [3] S. Donovan, "The SIP INFO Method," Internet RFC 2976, Internet Engineering Task Force, Oct. 2000.
- [4] J. M. Hsu, W. B. Lain, and J. C. Liang, "A Picture-based SIP Push-to-Talk Service," in Proceedings of 2007 Symposium on Digital Life Technologies, CDROM, June 7-8, Tainan, 2007.
- [5] W. Hyun, S. Park, I. Lee, and S. Kang, "A study on design and implement of XCAP Server," in Proceedings of the 8th International Conference on Advanced Communication Technology (ICACAT 2006), Feb. 2006.
- [6] P. Kim, A. Balazs, E. van den Brock, G. Kieselinann, and W. Bohm, "IMS-based push-to-talk over GPRS/UMTS," in Proceedings of 2005 IEEE Wireless Communications and Networking Conference (WCNC 2005), pp. 2472-2477, 2005.
- [7] A. Parthasarathy, "Push to talk over cellular (PoC) server," in Proceedings of 2005 IEEE International Conference on Networking, Sensing and Control (ICNSC 2005), pp. 772-776, 2005.
- [8] J. Rosenberg, "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)" Internet RFC 4825, Internet Engineering Task Force, May 2007.
- [9] J. Rosenberg et al, "SIP: Session initiation protocol," IETF RFC 3261, June 2002.
- [10] L. Y. Wu, M. H. Tsai, Y. B. Lin, and J. S. Yang, "A client-side design and implementation for push to talk over cellular service," Wireless Communications and Mobile Computing, July, 2006.
- [11] SER (SIP Express Router), online available at <http://www.iptel.org/ser>.
- [12] Open Mobile Alliance. Push to Talk over Cellular (PoC) – Architecture, Candidate version 1.0, OMA-AD-PoC-V1_0-20060127-C, online available at http://www.openmobilealliance.org/release_program/poc-archive.htm.