

以 SIP 為基礎多媒體會議系統之設計與實作

張昆旭、蔡昱航、陳威志、蘇暉凱*
南華大學資訊工程學系

摘要 — SIP 是因應多媒體傳輸所發展出來的通訊協定，但在多媒體會議控制尚無統一的標準規範，許多設計都是必須由參與者事先協議好會議 URI，再由會議參與者主動建立連線到會議伺服器，這種服務模型與我們一般撥打會議電話的習慣不同，而且使用上很不方便。因此，本論文以集中式會議服務架構為基礎，依據 IETF draft-ietf-sip-uri-list-conferencing-01 引入 URI-list 概念，設計與實作支援 URI-list 之多媒體會議伺服器與用戶終端應用程式，實現會議主席可以一次邀請多位參與者，共同進行多媒體會議。¹

一、前言

VoIP (Voice over Internet Protocol) 是一種將語音即時通訊資料透過網際網路來傳遞的技術，也被簡稱為「網路電話」。VoIP 將聲音的類比訊號數位化後，並透過分封交換網路進行點對點的即時通訊。近年來，由於傳輸與高速交換技術的進步，讓 VoIP 服務擺脫掉品質差的壞印象，但卻能帶來低通訊成本、多元化應用整合的好處，同時也讓更多需要高頻寬及高傳輸品質的即時多媒體應用普及化，多媒體會議即是網際網路殺手級應用之一 (Killer Applications)。

現今跨區或跨國企業，為了降低生產成本與專業分工，往往採取全球分工的策略，以獲得最大的經濟效益；依照各國原料與人力成本，以及專業層面上的考量，紛紛在各國成立分支機構。因此，運輸、差旅與通訊費用，往往是跨國企業的一大支出。多媒體會議即透過低成本之網際網路連接國內外不同地點之會議室，使與會人員能看見、聽到對方聲音，並且可以相互交換多媒體資料，猶如面對面的現場開會，藉以解決決策者分身乏術的困擾，無需出國，便能擁有專業人才參與會議，同時具有降低旅費開支，應付突發事件，緊急開會迅速解決問題。由此可知，多媒體會議的需求性與未來無限發展的潛力。

雖然 ITU-T H.323[1] 已明確制定多媒體會議的標準，但是基於系統複雜度與發展成本因素下，多媒體會議產品 MCU (Multipoint Control Unit) Server 價格相當昂貴，整套視訊會議設備動輒數十萬台幣，更甚者高達百萬以上。除此之外，H.323 設計上為一種多媒體廣泛協定，採用各種編碼方式，使得整體框架過於累贅、複雜，造成效率大幅降低。另一方面，H.323 缺乏支援 VoIP Signaling Protocol 的擴展性。基於以上理由致使 H.323 結構複雜，彈性以及擴充性低，因此無法大量推廣。

近幾年 SIP (Session Initiation Protocol) 會議的相關研究雖然較多人探討，但卻還沒有統一的標準與規範。許多設計都是必須會議參與者事先協議好會議 URI (Uniform Resource Identifier)，再由會議參與者主動建立連線到會議伺服器，這種服務模型與我們一般撥打傳統會議電話的模式相差很多，而且使用上很不方便。因此，本論文以集中式會議服務架構為基礎，依據 IETF draft-ietf-sip-uri-list-conferencing-01[2] 引入 URI-list 概念，設計與實作支援 URI-list 之多媒體會議伺服器與用戶終端應用程式，實現會議主席可以一次邀請多位參與者，共同進行多媒體會議。

二、相關研究

IETF 組織在近幾年的努力下，已完成了許多網路電話協定標準的制定，而目前正積極發展中的多媒體會議架構與會議控制協定標準，其模型可分為兩種，分散式架構及集中式架構。並定義 Focus 與 Mixer 兩個元件。

- Focus：負責多媒體會議建立、釋放、控制與管理。
- Mixer：提供聲音與影像混合功能。

分散式會議架構將 Focus、Mixer 設計在各 UA (User Agent) 上，並應用於多點傳播的網路傳輸環境上。其優點在於可節省資源，不需額外配置設備，便可達到多人會議的目標。當多人會議連線時，每個人有各自的 Focus 及 Mixer，然而每一端的網路品質及電腦訊息處理速度皆有不同，極易導致會議品質的流失、資源管理不當。在此情況下，缺乏管理、安全性；另一方面技術門檻較高，目前發展進度相對也比較緩慢。

集中式會議架構，將 Focus、Mixer 集中在會議伺服器上，會議的進行與離開必須由會議伺服器管理與控制。以此為前提，則所有 UA 的會議品質及安全控管都有一定程度的保障。相對會議人數的增加，造成會議伺服器的負擔也加大。且 UA 無法得知其他 UA 的訊息，當其中某一個 UA 離開會議時，有可能其他 UA 根本不知道。但就資源管理層面而言，此架構的可行性極高。

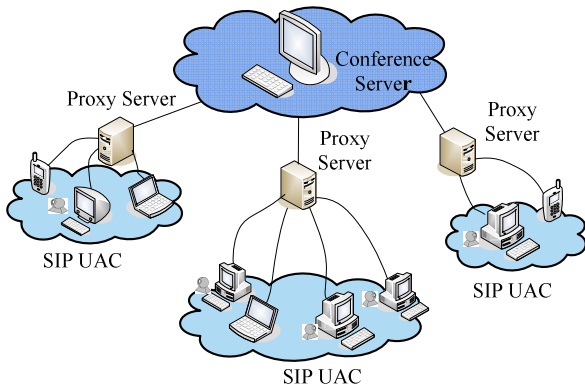
集中式會議架構相較分散式架構而言，除以上所述會議品質及資源控管容易之外，標準規範亦較為完整、可行性較高，使網路應用服務提供者對現有的 SIP VoIP 服務做整合較簡單，進而提供更高的多媒體會議加值服務。目前 IETF 所發表一系列 SIP 為基礎的相關議題有：會議需求[3]、多媒體會議架構[4]、會議控制協定[5]、會議發言控制權需求[6]、BFCP[7]、BFCP 串流的 SDP 格式[8] 等。

¹ 本研究由國科會贊助，計畫編號 NSC 95-2218-E-343-002。

三、以 SIP 為基礎多媒體會議系統

3.1 系統架構

本論文所討論之系統架構如圖一所示，採用集中式會議架構。由三個部份所組成，Conference Server、Proxy Server、以及 SIP UAC (User Agent Client)。Conference Server 主要具有維持、管理會議並且具備 Video Mixer 與 Audio Mixer 的功能，除此之外更需具備解讀本論文所討論的 URI-list 之能力，並且根據 URI-list 上的成員 URI 邀請至會議；而 Proxy Server 在此的功能為重新導向，將 UA 所傳送的訊息，傳送給 Conference Server；最後 SIP UAC 除了基本功能之外，另外還增加對 URI-list 的處理功能，與 Conference Call 的能力。



圖一：系統架構

3.2 設計原理

根據 RFC 4579[9] 所描述要建立一個會議，須由 UA 送出一個 INVITE 訊息需求給 CF (Conference Factory) 的 URI，當 CF 收到需求時會回傳給 UAC 實際 Conference Server 的 URI，而 UAC 得到 Conference Server 的 URI 後，即可進行會議，另外可以新增成員參與此會議。

一般會議的建立是由一個發起人，去邀請要參與會議的人員。所以本論文根據 IETF draft-ietf-sip-uri-list-conferencing-01[2] 所提出使用 URI-list 來初始會議，在 UAC 對 CF 提出建立會議需求的同時會夾帶 URI-list 給 CF，此 URI-list 包含 UAC 想讓 Conference Server 邀請的成員 URI，一旦 Conference Server 成功建立時應該嘗試將 URI-list 上的成員邀請到此會議。

URI-list 的格式如圖二所示，由兩大部分所組成：一是根據 IETF draft-ietf-simple-xcap-list-usage-05[10] 以 XML (Extensible Markup Language) 的「resource-list」為格式，列出初始成員的 URI 成為 URI-list 訊息本體；二則是根據 IETF draft-ietf-sipping-capacity-attribute-03[11] 使用 XML 擴充 Copy Control 屬性，加在 URI 後面用來描述 URI 的屬性，如圖二內「copyControl」。而屬性集合有「to」，「cc」，及「bcc」用來告知 Conference Server 所送出的 INVITE 訊息是否應夾帶該成員的 URI 於 URI-list 內，當 URI 後面屬性為「to」或「cc」時表

示 Conference Server 送出的 INVITE 在 URI-list 本體必須列出該成員的 URI；而屬性為「bcc」則否。

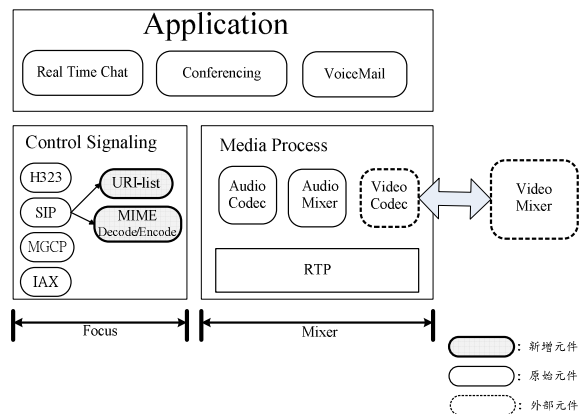
```
<?xml version=" 1.0" encoding=" UTF-8" ?>
<?resource-lists xmlns=" urn: ietf: params: xml: ns: resource-lists"
xmlns: cp=" urn: ietf: params: xml: ns: copycontrol" >
  <list>
    <entry uri=" sip:bille@example.com" cp: copyControl=" to" />
    <entry uri=" sip:joe@example.org" cp: copyControl=" cc" />
    <entry uri=" sip:ted@example.net" cp: copyControl=" bcc" />
  </list>
</resource-lists>
```

圖二： URI-list 範例

3.3 Focus 和 Mixer 元件設計

在 Conference Server 方面，本論文採用開放軟體 Asterisk[12] 作為基礎架構，Asterisk 是一個功能強大，並且彈性極高的閘道器軟體，除此之外更支援多種 VoIP 協定，對於 Client 端的各式各樣協定能夠很簡單的處理，如 H.323、SIP 協定，甚至於具備維持會議與管理。

為了支援本論文所討論之 URI-list，Conference Server 必須具有能力去解讀 UAC 所傳送的 INVITE 去建立會議，並區分 INVITE 訊息內的多重本體 — SDP 及 URI-list；此外根據 URI-list 上的成員 URI 由 Conference Server 主動傳送 Outgoing INVITE 訊息給各個會議參與者，由 Conference Server 統一邀請會議成員參與會議。本論文設計、實作之 Conference Server 架構如圖三所示。



圖三： The Scope of Server

下列是 Conference Server 所實作出的功能。

1. MIME：
Conference Server 收到含有 INVITE 時，如果具有多重本體，也就是 INVITE 除了 SDP 協商多媒體的本體之外，還有 URI-list 本體，而 Conference Server 為了區分這兩個本體的不同，使用 MIME 去解碼，藉此以分出 SDP 本體與 URI-list 本體。
2. URI-list：
UAC 所傳送建立會議需求 INVITE 夾帶的 URI-list 是用 XML resource-list 來表示會議成員 URI。所以 Conference Server 具備解讀 XML 的能力，從 URI-list 將每位需要邀請的成員 URI 逐一解析

出來再做邀請的動作。

3. Outgoing INVITE :

當 Conference Server 收到 URI-list 並且解析出每位需要邀請至會議成員的 URI 之後，Conference Server 除了要建立會議之外，還要主動去邀請 URI-list 上的成員，因此 Conference Servers 能夠主動送出 Outgoing INVITE 給每位成員，而這也是一般 Conference Server 所缺乏的能力。

4. Audio Mixing :

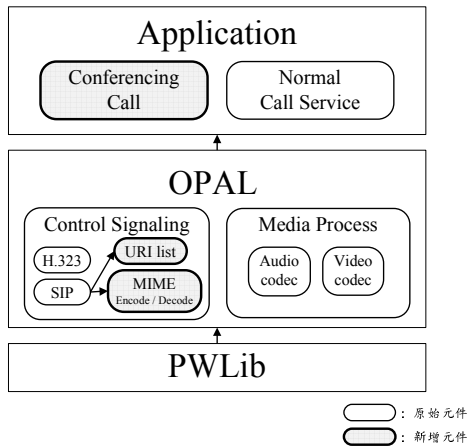
Audio Mixing 會處理會議成員送來的語音串流，它會負責語音的編碼和解碼，混合語音後，再將混音後的語音串流傳至會議內的各個成員。

5. Video Mixing :

因為 Asterisk 本身並沒有影像混合功能，所以要額外掛載 Video Mixer，由 CONFIFANCE[13] 所提供的 Video Mixer，來支援影像混合的功能，它會處理會議成員送來的影像串流，並負責影像的編碼和解碼；另一方面，將會議成員的影像混合後，再回傳給 Conference Server，由 Conference Server 將混合後的影像串流回傳給會議內的各個成員。

3.4 Client 端設計

就現有 SIP 協定中，並無支援 URI-list 本體的能力爲了達到這項功能，本論文以 Open H323[14] 該組織，所研究 OPAL 專案下提供的 SimpleOpal — 具有 SIP Client 端功能的開放軟體做延伸，以達到本論文的需求，程式架構如圖四所示。



圖四：The Scope of SimpleOpal

其中 PWLib (Portable Windows Library) 主要做爲 Operating System 與 OPAL 之間 I/O 需求的溝通介面；而 OPAL 的函式具有支援 SIP 及 H.323 協定，此外本論文在 Control Signaling 方面新增 URI-list 功能，以達到在 URI-list 本體所需的具有 XML 的 Decode 與 Encode 的能力，並且爲了區分 SDP 與 URI-list 本體加入 MIME 處理；而應用層因爲 SimpleOpal 本身並不具有

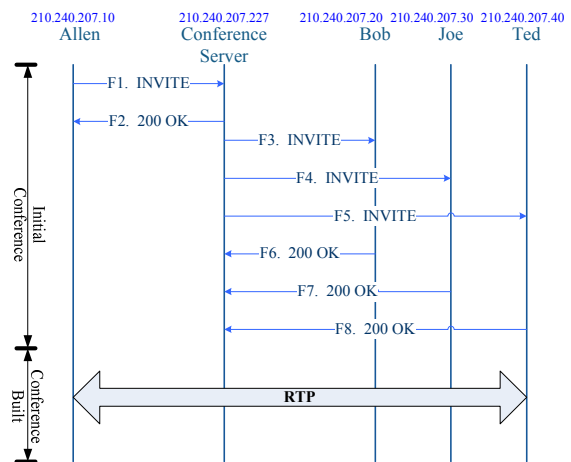
Conferencing Call 的能力，因此加入了 Conferencing Call 的元件；另外更加強邀請會議參與者 URI 的屬性設定，並編寫部份程式，使送出的 SIP 訊息內具有以 XML resource-list 方式來描述 URI-list 本體。

原本的 SimpleOpal 軟體，SIP 訊息只允許傳遞一位對象，選定對象後，便直接做出 INVITE 動作。而透過修改過 SimpleOpal 軟體以達到使用者在軟體介面可選取多位對象參加會議，經程式彙整成 URI-list，使用 MIME 多重本體做 SIP 內容的調整成初始會議需求的 INVITE，再送出 INVITE 的動作。

四、設計結果與範例

4.1 會議起始

整個會議建立流程如圖五所示，包含兩個部份— Initial Conference 到 Conference Built。



圖五：The Sequence of Conference Built

- 一開始 Conference Server 處於 Waiting 的狀態，等候 Client 需求，而 F1 INVITE 則是由 Allen 所送出的建立會議需求，INVITE 需求內容如下頁圖六所示，第一行表示送出的 SIP 方法爲 INVITE，對象一開始爲 Conference Factory (第 5 行)，而在 Require Header (13 行) 會註明需要 Server 了解本體上包含 URI-list 必須用 INVITE 去邀請所含的成員 URI，另外會在 Content-Type Header (14 行) 註明含多個內容型態組合，並使用「boundary1」爲分界，需使用 MIME 去解析，第一個內容本體則是 SDP (18 行至 28)，與 Server 協商 Allen 的多媒體能力：Audio 與 Video Port，此外還有使用何種 Audio、Video Codec；而另外一個本體則是本論文所討論的 URI-list (31 行至 42)，Allen 將要邀請的成員之 URI 使用 XML 來表示，並設定 URI 屬性皆爲「to」，使會議參與者皆可得知參與會議之成員。
- 當 Conference Server 收到需求後會回傳 F2 200 OK 給 Allen，內容如圖七所示，在 Contact Header (第 8 行) 告知 Allen，Conference Server 的 URI 並且會在後面註明 isFocus；Allen 可從該訊息

知道 Conference Server 具有解讀 URI-list 的能力，並且已經成功建立 Conference Server，除此之外此訊息會夾帶 SDP 本體 (14 行至 23)，與 Allen 協商 Conference Server 的多媒體能力。

```

1 INVITE sip:conf-fact@210.240.207.227 SIP/2.0
2 Via: SIP/2.0/TCP 210.240.207.10:5060
3   ;branch=z9hG4Khjhs8ass83
4 Max-Forwards: 70
5 To: "Conf Factory" <sip:conf-fact@210.240.207.227>
6 From: Allen <sip:allen@210.240.207.10>;tag=32331
7 Call-ID: d432fa84b4c76e66710
8 Cseq: 1 INVITE
9 Contact: <sip:allen@210.240.207.10:5060>
10 Allow: INVITE, ACK, CANCEL, BYE, REFER
11 Allow-Events: dialog
12 Accept: application/sdp, message/sipfrag
13 Require: recipient-list-invite
14 Content-Type: multipart/mixed;boundary=" boundary1"
15 Content-Length: 690
16
17 --boundary1
18 Content-Type: application/sdp
19
20 v=0
21 o=allen 2890844526 2890842807 IN IP4 210.240.207.10
22 s=A conversation
23 c=IN IP4 210.240.207.10
24 t=0 0
25 m=audio 20000 RTP/AVP 0
26 a=rtpmap: 0 PCMU/8000
27 m=video 20002 RTP/AVP 31
28 a=rtpmap: 31 H261/90000
29
30 --boundary1
31 Content-Type: application/resource-lists+xml
32 Content-Disposition: recipient-list
33
34 <?xml versino=" 1.0" encoding=" UTF-8" ?>
35 <resource-list xmlns=" urn:ietf:params:xml:ns:resource-lists"
36   xmlns:cp=" urn:ietf:params:xml:ns:copyControl" >
37   <list>
38     <entry uri=" sip:bill@210.240.207.10" cp.copyControl=" to" />
39     <entry uri=" sip:joe@210.240.207.20" cp.copyControl=" to" />
40     <entry uri=" sip:ted@210.240.207.30" cp.copyControl=" to" />
41   </list>
42 </resource-list>
43 --boundary1--
  
```

圖六： INVITE request received at the conference server

```

1 SIP/2.0 200 OK
2 Via: SIP/2.0/TCP 210.240.207.10:5060
3   ;branch=z9hG4Khjhs8ass83
4 To: "Conf Factory" <sip:conf-fact@210.240.207.227>
5 From: Allen <sip:allen@210.240.207.10>;tag=32331
6 Call-ID: d432fa84b4c76e66710
7 Cseq: 1 INVITE
8 Contact: Conference Server <sip:conf34@210.240.227>;isFocus
9 Allow: INVITE, ACK, CANCEL, BYE, REFER
10 Allow-Events: dialog, conference
11 Accept: application/sdp, message/sipfrag
12 Content-Type: application/sdp
13 Content-Length: 690
14
15 v=0
16 o=conf 31643 31643 IN IP4 210.240.207.227
17 s=session
18 c=IN IP4 210.240.207.227
19 t=0 0
20 m=audio 40000 RTP/AVP 0
21 a=rtpmap:0 PCMU/8000
22 M=video 40002 RTP/AVP 31
23 A=rtpmap:31 H261/90000
  
```

圖七： OK responded by the conference server

- 當 Conference Server 收到具有 URI-list 的 INVITE 之後，會根據 URI-list 上的成員 URI 逐一的邀請到會議內。Bob 即為 URI-list 上的其中一位，所以 Conference Server 會傳送 F3 INVITE 給 Bob，而內容如圖八所示，會在 Contact Header (第 9 行) 告知 Bob，Conference Server 的 URI 並註明

isFocus；此外，在 Require Header (13 行) 會註明此 INVITE 包含會議成員的名單，而在 Context-Type (14 行) Header 可知需用 MIME 解析本體：第一個本體為 SDP (18 行至 29)，與 Bob 協商 Conference Server 的 Audio 與 Video Port 此外還有使用何種 Audio、Video Codec；而另外一種內容型態則是 URI-list (31 行至 42)，Bob 可以由此 URI-list 本體上的 URI (38 行至 40) 得知參與會議的成員有哪些人。當 Bob 收到後也會傳送夾帶 SDP 本體的 200 OK 訊息給 Conference Server，協商自身的多媒體能力。另外 F4、F5 也是 Conference Server 所收到 URI-list 內的成員 (Joe、Ted) 之邀請過程，訊息內容與圖八自然相同，收到 INVITE 後 Joe 與 Ted 同樣會傳送類似圖七之 200 OK 訊息，以協商各自的多媒體能力。

```

1 INVITE sip:bill@210.240.207.20 SIP/2.0
2 Via: SIP/2.0/TCP 210.240.207.227
3   ;branch=z9hG4bKhjhs8aa454
4 Max-Forwards: 70
5 To: <sip:bill@210.240.207.20>
6 From: Conference Server <sip:conf34@210.240.207.227>;tag=234332
7 Call-ID: 389sn189dasdf
8 Cseq: 1 INVITE
9 Contact: <sip:conf34@210.240.207.227>;isFocus
10 Allow: INVITE, ACK, CANCEL, BYE, REFER
11 Allow-Events: dialog, conference
12 Accept: application/sdp, message/sipfrag
13 Require: recipient-list-invite
14 Content-Type: multipart/mixed;boundary=" boundary1"
15 Content-Length: 690
16
17 --boundary1
18 Content-Type: application/sdp
19
20 v=0
21 o=conf 2890844343 2890844343 IN IP4 210.240.207.227
22 s=session
23 c=IN IP4 210.240.207.227
24 t=0 0
25 m=audio 40000 RTP/AVP 0
26 a=rtpmap: 0 PCMU/8000
27 m=video 40002 RTP/AVP 31
28 a=rtpmap: 31 H261/90000
29
30 --boundary1
31 Content-Type: application/resource-lists+xml
32 Content-Disposition: recipient-list
33
34 <?xml versino=" 1.0" encoding=" UTF-8" ?>
35 <resource-list xmlns=" urn:ietf:params:xml:ns:resource-lists"
36   xmlns:cp=" urn:ietf:params:xml:ns:copyControl" >
37   <list>
38     <entry uri=" sip:bill@210.240.207.10" cp.copyControl=" to" />
39     <entry uri=" sip:joe@210.240.207.20" cp.copyControl=" to" />
40     <entry uri=" sip:ted@210.240.207.30" cp.copyControl=" to" />
41   </list>
42 </resource-list>
43 --boundary1--
  
```

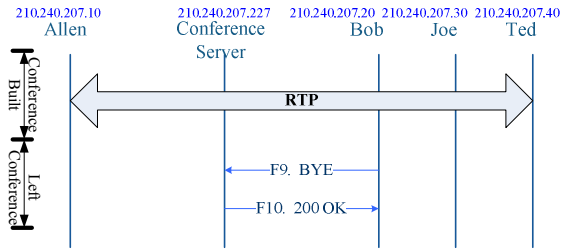
圖八： INVITE request sent by the conference server

- 當 URI-list 上的成員都邀請至會議之後，即可開始進行會議，展開 Real Time Protocol 傳輸，參與的成員會將各自的 Audio、Video 傳給 Conference Server，而 Conference Server 會將混合後的 Audio、Video 再回傳給參與的成員。

4.2 會議結束

會議進行中時，要離開會議是使用各自離開的方式，流程如圖九所示，Bob 離開時會傳送 F9 BYE 的訊息，其內容如圖十所示，從第 5 行可得知 Bob 離開時會

告知 Conference Server，當 Conference Server 收到訊息確認後，會回傳 F10 200 OK 給 Bob，告知 Bob 已經成功離開此會議。



圖九：The Sequence of Left Conference

```

1 BYE sip:confB4@210.240.207.227 SIP/2.0
2 Via: SIP/2.0/TCP 210.240.207.20:5060;branch=z9hG4bK30126
3 Max-Forwards: 70
4 From: <sip:bill@210.240.207.20>;tag=1441
5 To: <sip:confB4@210.240.207.227>;tag=as28ae3c7c
6 Call-ID: lj9FpLxk3ux
7 CSeq: 21 BYE
8 Contact: <sip:bill@210.240.207.20:5060>
9 Content-Length: 0

```

圖十：BYE request sent by Bob

結論

本論文主要是在探討多媒體會議的初始，使用者可以使用含有 URI-list 的 INVITE 訊息，快速的將會議建立起來，並且邀請成員參與會議，將會議的建立簡單化，並將邀請成員流程精簡化。在本論文以 SIP 為基礎多媒體會議系統之設計與實作，使 Conference Server 能夠支援 URI-list 並且主動邀請成員至會議中；而 Client 端能夠發送 URI-list 的 INVITE 訊息，具有 Conferencing Call 能力，藉此實作出目前在會議所缺乏的服務功能。透過本論文的設計與實作，探討 VoIP Conferencing，相信在未來對於 VoIP Conferencing 能夠更為普及，並廣為應用。

參考文獻

- [1] H.323v4: Packet-based multimedia communications systems, ITU-T Std, Nov 2000.
- [2] Camarillo, G. and Johnston, A. "Conference Establishment Using Request-Contained Lists in the Session Initiation Protocol (SIP)," draft-ietf-sip-uri-list-conferencing-01, January 26, 2007.
- [3] Levin, O. and Even, R. "High-Level Requirements for Tightly Coupled SIP Conferencing," RFC 4245, Nov 2005.
- [4] Rosenberg, J. "A Framework for Conferencing with the Session Initiation Protocol (SIP)," RFC 4353, Feb 2006.
- [5] Johnston, A. and Levin, O. "Session Initiation Protocol (SIP) Call Control-Conferencing for User Agents," RFC 4579, Aug 2006.
- [6] Koskelainen, P. and Ott, J. and Schulzrinne, H. and Wu, X. "Requirements for Floor Control Protocols," RFC 4376, Feb 2006.
- [7] Camarillo, G. and Ott, J. and Drage, K. "The Binary Floor Control Protocol (BFCP)," RFC 4582, Nov 2006.
- [8] Camarillo, G. "Session Description Protocol (SDP) Format for Binary Floor Control Protocol (BFCP) Streams," RFC 4583, Nov 2006.
- [9] Johnston, A. and Levin, O. "Session Initiation Protocol (SIP) Call Control-Conferencing for User Agents," BCP 110, RFC 4579,

August 2006.

- [10] Rosenberg, J. "Extensible Markup Language (XML) Formats for Representing Resource Lists," draft-ietf-simple-xcap-list-usage-05, February 2005.
- [11] Garcia-Martin, M. and Camarillo, G. "Extensible Markup Language (XML) Format Extension for Representing Copy Control Attributes in Resource List," draft-ietf-sipping-capacity-attribute-03, December 2006.
- [12] Asterisk 官方網站, <http://www.asterisk.org>
- [13] CONFIANCE 官方網站, <http://confiance.sourceforge.net>
- [14] SimpleOPAL 官方網站, <http://www.voxgratia.org>