Title	Real time communication and collaboration
Sub Title	実時間コミュニケーションとコラボレーション
Author	Sinchai Kamolphiwong Thossaporn Kamolphiwong Suthon Sae-Wong
Publisher	慶應義塾大学湘南藤沢学会
Publication year	2008
Jtitle	Keio SFC journal Vol.8, No.2 (2008.) ,p.7- 20
JaLC DOI	10.14991/003.00080002-0007
Abstract	In this paper, we first present some proposed extensions based on SIP (Session Initiation Protocol) conference scenarios. We then show how to use such extensions to enhance interactive distance learning (IDL) applications as an example. Our work describes some enhancements of conference scenarios based on SIP. The system architecture to support such scenarios as well as signal flows are presented. Some flow parameters are given, to show real deployment possibility. We have proposed two additional components to enhance conference features: Conference Manager Server (CMS) and Conference Repository (CR). We have deployed these scenarios for SIP based e-learning applications. We have shown some e-learning scenarios when interactive communications are needed, for example, class scheduled learning, class mate group finding. In addition, other rich features can be added, e.g. sharing on-line objects and documents in real-time, virtual interactive white-board, and multimedia recorder. We have proposed the implementation architecture, some main features are described. We discussed for some future challenging works: P2P based SIP may be adopted. Moreover, converging to UCC (Unified Communication and Collaboration), as well as working on multi-platform and multi-devices are concerned. 本稿では、まず、SIP (Session Initiation Protocol) の会議通話向けの拡張を提案した。次に、この拡張の利用例として、enhance interactive distance learning (IDL) アプリケーションにおける利用例を示した。本研究では、会議向けの拡張のため、Conference Manager Server (CMS) と Conference Repository (CR) の 2 つのコンポーネントの追加を提案し、E ラーニング用のアプリケーションへの適用を検討した。E ラーニングにおいては、受講者間の会話など、インタラクティブなコミュニケーションが必要な場合がある。また、教材のリアルタイム共有、黒板の共有、マルチメディア録音機能の利用も考えられる。本研究では、これらの機能の実装方針を提案した。
Notes	特集 アジアにおけるネットワークと遠隔教育 招待論文
Genre	Journal Article
URL	https://koara.lib.keio.ac.jp/xoonips/modules/xoonips/detail.php?koara_id=0402-0802-0100

◆招待論文◆

Real Time Communication and Collaboration

実時間コミュニケーションとコラボレーション

Sinchai Kamolphiwong
Associate Professor, Center for Network Research (CNR),
Department of Computer Engineering, Faculty of Engineering, Prince of Songkla University

シンチャイ カモルピワォン
ソンクラーナカリン大学工学部准教授

Thossaporn Kamolphiwong
Associate Professor, Center for Network Research (CNR),
Department of Computer Engineering, Faculty of Engineering, Prince of Songkla University

トサポン カモルピワォン
ソンクラーナカリン大学工学部准教授

Suthon Sae-Wong
Lecturer, Center for Network Research (CNR),
Department of Computer Engineering, Faculty of Engineering, Prince of Songkla University

ストン サエ・ワォン
ソンクラーナカリン大学工学部専任講師

In this paper, we first present some proposed extensions based on SIP (Session Initiation Protocol) conference scenarios. We then show how to use such extensions to enhance interactive distance learning (IDL) applications as an example. Our work describes some enhancements of conference scenarios based on SIP. The system architecture to support such scenarios as well as signal flows are presented. Some flow parameters are given, to show real deployment possibility. We have proposed two additional components to enhance conference features: Conference Manager Server (CMS) and Conference Repository (CR). We have deployed these scenarios for SIP based e-learning applications. We have shown some e-learning scenarios when interactive communications are needed, for example, class scheduled learning, class mate group finding. In addition, other rich features can be added, e.g. sharing on-line objects and documents in real-time, virtual interactive white-board, and multimedia recorder. We have proposed the implementation architecture, some main features are described. We discussed for some future challenging works: P2P based SIP may be adopted. Moreover, converging to UCC (Unified Communication and Collaboration), as well as working on multi-platform and multi-devices are concerned.

本稿では、まず、SIP (Session Initiation Protocol) の会議通話向けの拡張を提案した。次に、この拡張の利用例として、enhance interactive distance learning (IDL) アプリケーションにおける利用例を示した。本研究では、会議向けの拡張のため、Conference Manager Server (CMS) と Conference Repository (CR) の２つのコンポーネントの追加を提案し、E ラーニング用のアプリケーションへの適用を検討した。E ラーニングにおいては、受講者間の会話など、インタラクティブなコミュニケーションが必要な場合がある。また、教材のリアルタイム共有、黒板の共有、マルチメディア録音機能の利用も考えられる。本研究では、これらの機能の実装方針を提案した。

Keywords: SIP, Interactive communication, Distance Learning, IDL, VoIP

1 Introduction

Conference systems based on the Internet infrastructure are widely deployed, in both commercial and free services. The most widely used standards are H.323 [1] and SIP (Session Initiation Protocol) [2]. Both of them are developed to support for a group conference system. H.323 is highlighted to load balancing and management issue. In contrast, SIP is highlighted to light-weighted flexible and extensible issues. Recently, SIP is more widely used than H.323 [3] [4]. However, SIP may have some difficulties for legacy conference, e.g., a conference manager.

When we figure out the whole conference system, starting at the beginning state, we have found that it is hard to find a complete system in any literatures. In [12], some parts are left out, e.g., a conference mixer. A new protocol, XCCP, which is based on XML, was created but it is not suitable for real deployment when such service needs to be installed on all UAs (User Agent). SOAP (Simple Object Access Protocol) was proposed to be used as a conference policy control protocol [13] but it has some limitations, e.g., high overhead and transaction modification.

To make a conference system working on a real environment with SIP as the protocol, reusing the most existing protocols is necessary. This will benefit from: no interoperability issues, mature and proven (of existing) protocols, and gain all benefits what existing protocols have and provide. However, there is a number of alternative ways to re-use existing protocols. Some of them, for example, are "Non-SIP Related Protocols" - e.g. XCAP, LDAP, and MSCML. Those of Non-SIP Protocols are engineering development issues, depending on their purposes and deployed scenarios.

In this paper, we propose enhancements of IP Conference scenarios based on Session Initiation Protocol (SIP). Our works have described some enhanced conference scenarios based on SIP, as well as system architecture to support such scenarios. A sample flow and some flow parameters, an implementation based on this framework are shown in a real practice. This paper also discusses some features for real-time interactive communications.

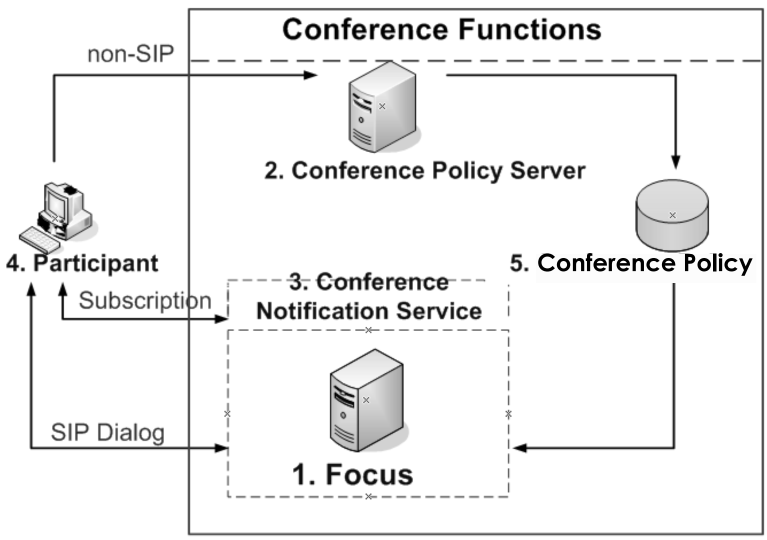


Figure 1 XCON Conference Framework

This paper is organized as follows: Section 2 presents some related work and standard reference models. System architecture design and signal flows are described in section 3. System performance discussion is made in Section 4. Section 5 presents the future works with some challenging issues. Conclusion is made in Section 6.

2 Related Works

XCON [7] and SIPPING [6] working groups have defined a conference framework for SIP protocol [5]. It consists of 5 important components as shown in Figure 1.

SIPPING framework uses SIP protocol as the signaling protocol to establish, tear down and modify conferences, while XCON framework does not depend on any particular signaling protocols. They focus on a standardized suite of protocols for tightly-coupled multimedia conferences. However, XCON framework is referred by SIPPING framework. Although XCON and SIPPING conferencing framework are widely adopted to implement a conference system using SIP, some parts of the framework are not clearly described, e.g., mechanism to configure or manage a conference policy.

Our work presents a SIP based conference system designed for a practical propose. We design the conference system based on RFC 4597 [6], which discusses basic and advance conference scenarios for SIP protocol, and RFC 4579 [7] which discusses a conference call control for SIP user agent. Our first proposal was IDLP (Interactive Distance Learning Protocol) that was presented in [16]. The extension scheme has been shown in [17][18] where more existing protocols are reused, e.g. LDAP (Lightweight Directory Access Protocol), MSCML (Media Server Control Markup Language), XCAP (Extensible Markup Language). Furthermore, how to deal with mobility feature e.g., mobile e-learning is presented in our previous work [19].

3 System Architecture Design

3.1 Proposed system architecture

In this section, we propose the system architecture for a SIP based real-time communication with collaboration capability. In our framework [17], we have proposed two main components as shown in Figure 2: Conference Manager Server (CMS) and Conference Repository (CR). Conference Server (CS) is equivalent to a Focus in XCON framework [7]. Focus is addressed to be the heart component of the system in XCON to support more complex media manipulations and enhanced conferencing features as well as the conference information (e.g. specific media mixing details, available floor controls). CMS is used for managing information and conference policy before starting a conference. CMS can initiate a conference by using SIP INVITE to a mixer, using SIP REFER for every pre-configured participant to join a conference. The signal flow of the conference manager is described in the next section. CMS does not try to replace any operations of SIP Focus. Since Focus is used when a conference session is created while CMS is used to manager a conference in both before and after conferences, even during a conference session. This means that CMS works with Focus to help and handle complicated conference scenarios, to make it simpler. Moreover, CMS handles more protocols, e.g. XCAP, LDAP, and SIP, due to its functions (while Focus understands only SIP signals). In addition, the Conference Repository which store information of participants is managed by CMS.

In our scenarios, a conference can be either open (public) or closed (private) conference. Conference Repository (CR), which stores conference information (e.g. conference URI, type of conference) is the place to find such conference information, for example a public conference can be found by anonymous users while a private conference can only be found by authenticated users. This mechanism can be done by

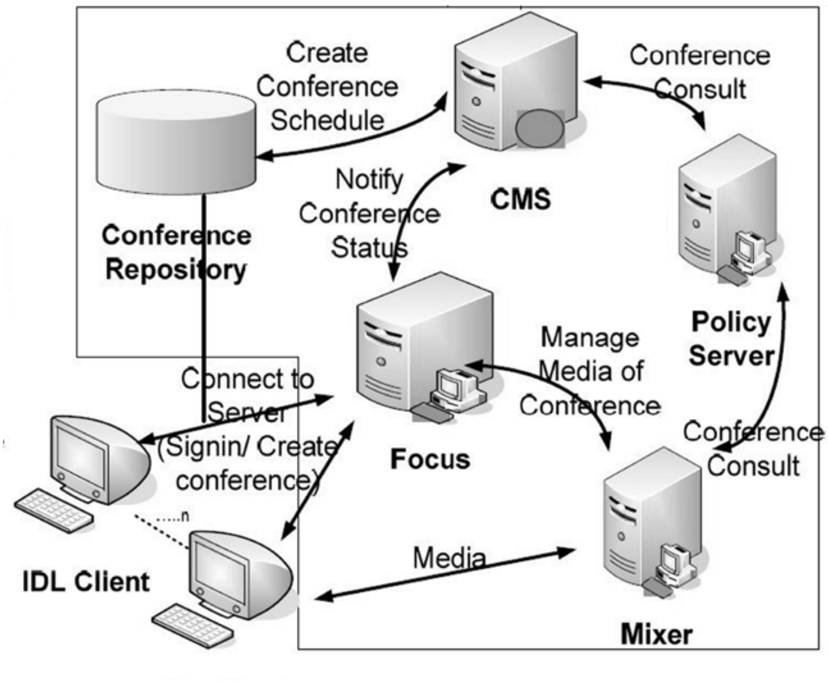


Figure 2 Proposed SIP based conference architecture

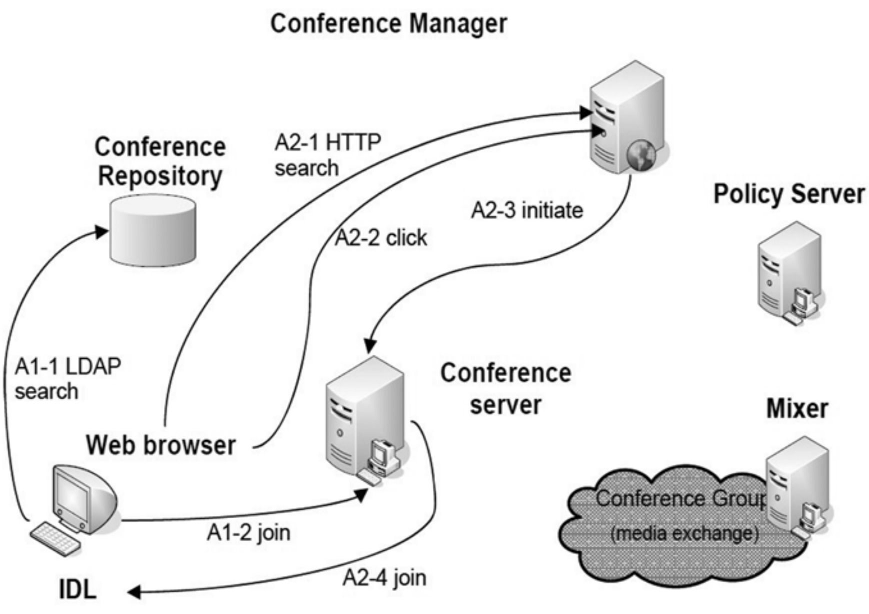


Figure 3 Group finding scenario

using conference descriptions and LDAP.

In this framework design [17], we have shown some details and signal flows on how to deal with extended conferences scenarios, for example, just conferencing (a few clicks to conference), group finding (a few click to create and reach the group), and conference scheduling. Based on the proposed architecture, we will show only one sample scenario, i.e., group finding, as shown in Figure 3. It shows how to search the conference information. This will allow users to find conferences of their interest without being asked to sign-in. There are two approaches as follows:

Approach 1: using LDAP

- Search entries in the conference repository. User uses LDAP to query conference information from the Conference Repository. The results are the conference URIs.
- Join the given the conference URI. User sends an INVITE request to join the conference based on the conference information (Conference URI).

Approach 2: using HTTP

- Use search function via web application.
- Click on the link to the Conference URI.
- CMS initiates an invite conference. CMS will send REFER request to Focus.
- The conference server consults the policy server whether the participant can join.
- Participant then can join the conference.

Figure 4 shows an example of a conference group finding using HTTP signaling flows. A User Agent (UA) queries a Conf-ID URI from the Conference Manager using HTTP request *F1*. Then the Conference Manager answers HTTP response *F2* with Conf-ID to the UA. The UA sends HTTP request

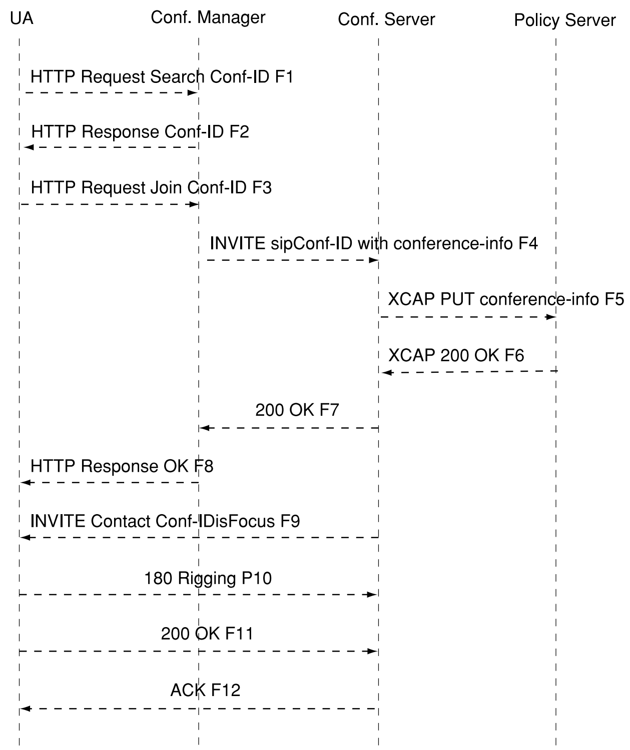


Figure 4 Group finding signal flow using HTTP

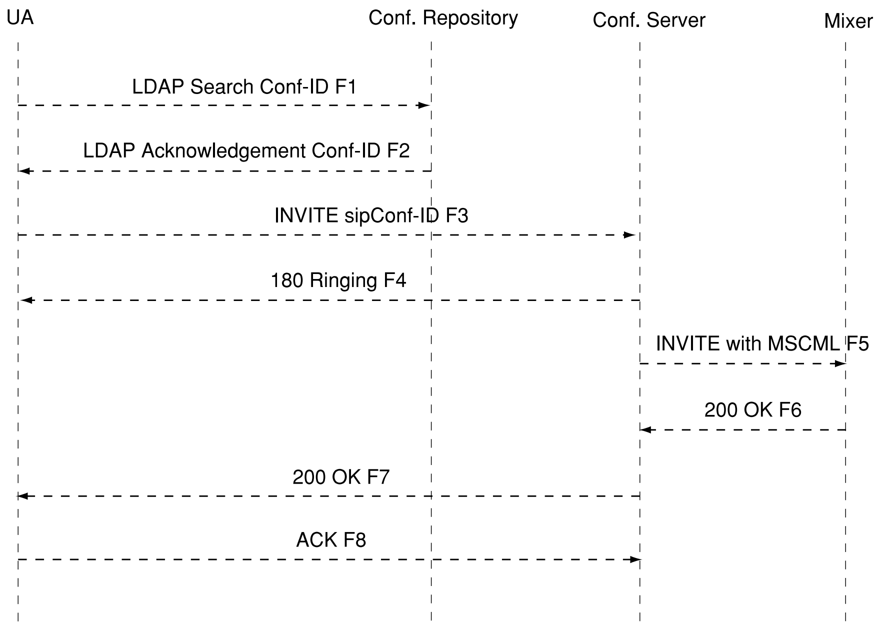


Figure 5 Group finding signal flow using LDAP

F3 to the Conference Manager. When the Conference Manager receives HTTP request *F3*, it sends INVITE request *F4* containing a conference information to the Conference Server for establishing conference. The Conference Server uses XCAP PUT to store conference information in the Policy Server. After that the Conference Server sends INVITE request containing Contact header field with parameter is Focus to the UA. After that system is ready for the conferencing.

Figure 5 shows an example of group finding using LDAP signal flow. A UA queries a Conf-ID URI from the Conference Repository using LDAP F1. The Conference Repository answers with LDAP Acknowledge containing Conf-ID URI. The UA sends INVITE method F3 to Conference Server for initialing the conference. The Conference Server replies with 180 (Ringing) response. Then the Conference Server requests the Mixer using INVITE request F5 containing MSCML data for specific Mixer. The Mixer sends 200 (OK) to response. The Conference Server is ready for conferencing.

3.2 System development

Based on our design framework, we developed the prototype software, named IDL (Interactive Distance Learning). Some important features are: conference facility, interactive white board, on-line chat, file transfer and sharing, real-time presentation, and multimedia recorder. Figure 6 shows our development architecture [20]. They are:

- Conference Manager is the heart component to manage a conference, for example, to create a conference, to maintain a contact list.
- Conference UI is a main part of showing each conference information and GUI. Our idea is the Conference UI is independent from the Conference Manager, to allow GUI layout for skin changing.

- Component is a software module for each function under Component Manager. There are 2 types of the component:
  - Component is the module to handle commands and events.
  - Visual component is the same as the component with the Conference UI interaction.
- Service is the common service elements of IDL, for example, doing a file transfer. We developed the service layer where these components can be shared and re-used by higher service components.

When interactive communication is concerned, providing services to serve this requirement is a challenge. The interactive white board (similar to a virtual white-bard with real-time interaction between participants) is such the tool needed, appeared on IDL's presentation layer (Figure 6). All participants can share the white board as a single entity. All documents, e.g., PowerPoint, PDF, can be brought to the white-board as an object for presentation. Any actions, such as drawing lines or objects, can be shared among all participants in real time. A policy may be used for sharing objects, for example, only object owner can edit such object. To make objects independent, the objects are drawn on another layer, on top of the presentation layer, so called "transparency". One presentation page may have several transparencies, so called "3D-transparency". Figure 7 shows a sample GUI of the interactive white-board.

In our implementation, the video conference can select its source from multiple sources. The video is considered as a media stream, and all video sources that have the same behavior should be able to use this session. For example, conference participants

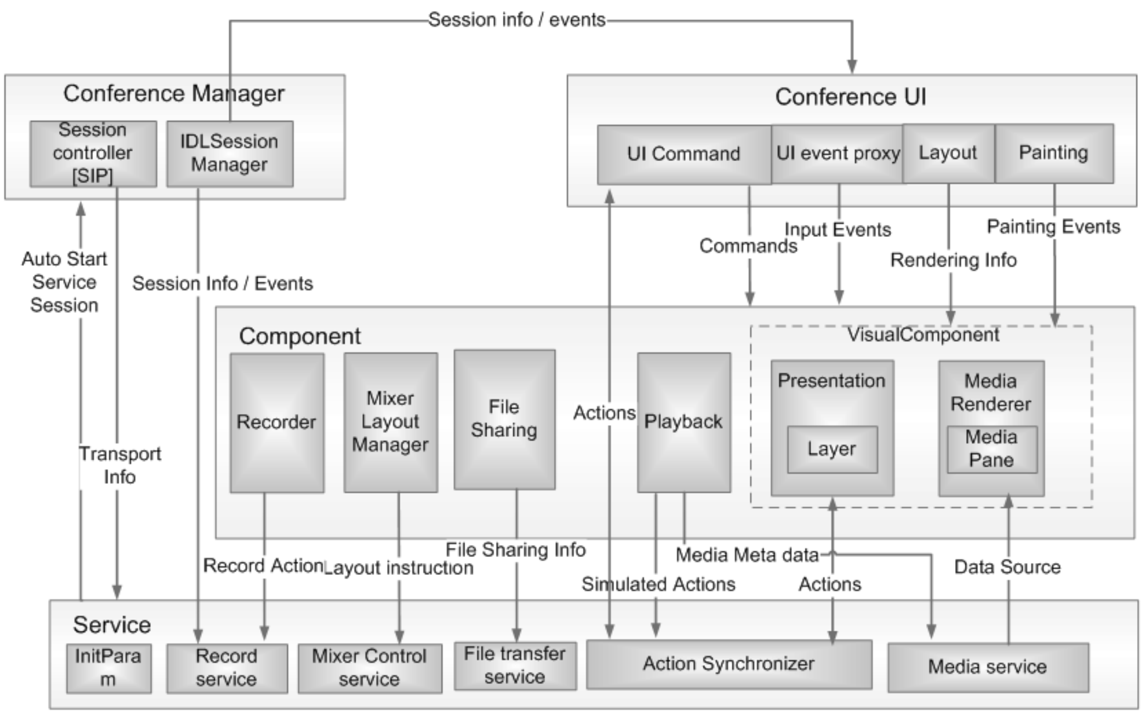


Figure 6 Interactive Distance Learning (IDL) Architecture

should be able to switch between USB camera, IP camera, and any video files, such as MPEG or WAV files. IDL treats all of them as the same media source. Some commercial software allow to share a working desktop, and this can also be treated as the same media source. However, in order to reduce bandwidth usage, a special tool is needed (considering that it is a slow moving screen).

IDL can create a conference in two ways, as shown in Figure 9, using a central mixer (for voice and video), and mixerless. When creating a conference using a central mixer, media streams (e.g.,

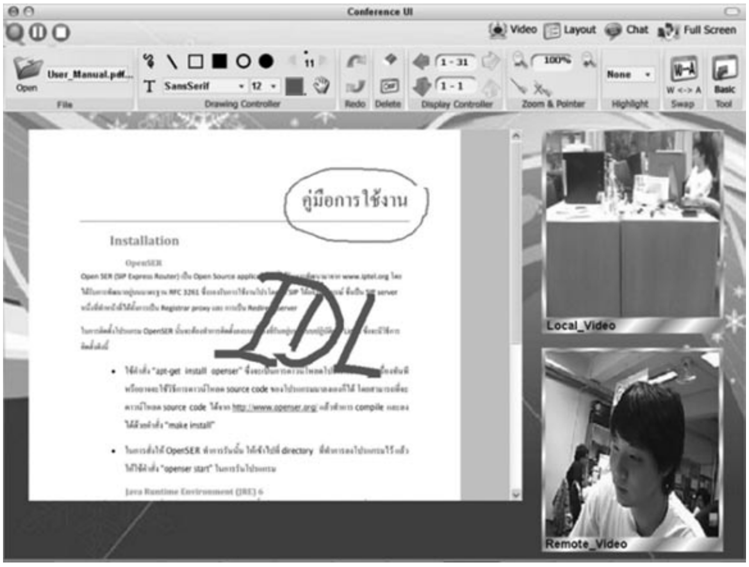
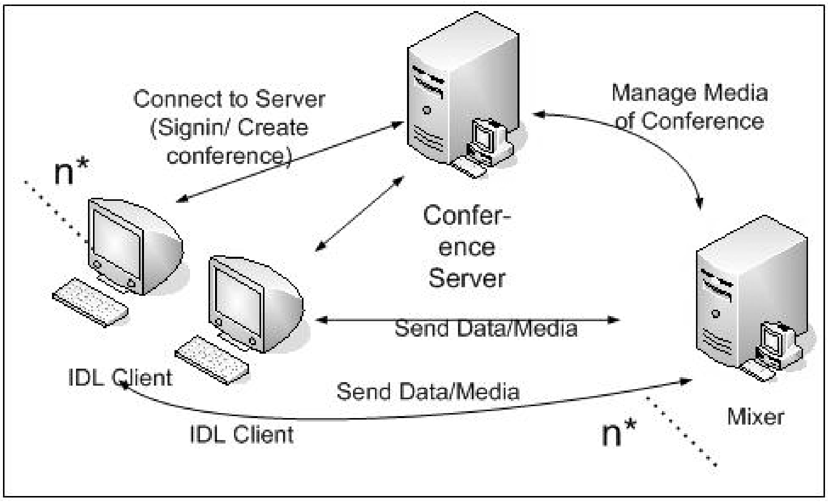


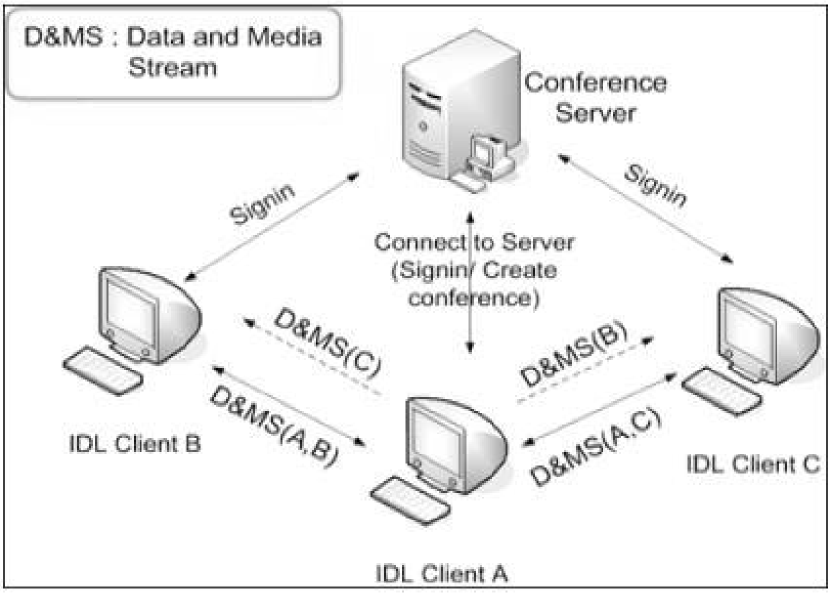
Figure 7 A sample GUI of interactive white-board

Figure 8 Input sources can be re-directed to other multimedia source. The sample picture shows the input source is from a movie.

voice and video) from all participants are sent to the mixer. Mixed stream is sent back to all participants via multicast protocol. This method will reduce the consumed bandwidth. However scalability is a big issue due to a performance of a mixer, and it is a single point of failure. Alternatively, IDL can create a conference without using a mixer. In this way, each node sends and receives media streams to and from other participants directly. The main benefit to this scenario is there is no need for voice and video mixers; no single of failure. However, this scenario consumes more bandwidth than the former. It is suitable for a small group of conference. We note that this model can be extended to Peer to Peer conference in the future.



(a)



(b)

Figure 9 Two conference modes can be made: (a) using a central mixer, (b) mixerless

Another proposed component to enhance our features is the Multimedia Recorder and Player (M-RP). The recorder is considered as a SIP client, to reduce the system complexity. We use standard SIP signals for this design purpose. This will gain the benefit that the recorder can be any clients of conference participant. This means that there can be many recorders, and there is no need to specify to any particular clients. In this case, the source information, such as PowerPoint files, PDF files, and video streams, are stored together with drawing objects drawn by conference participations on the interactive whiteboard, as well as synchronized actions (events/actions created by conference participations, and handled by the action synchronizer). The system does not record all media sources into a single video stream. The most advantageous feature of our prototype is it allows us to edit, insert, and delete such recorded stream any time, for example when playing back we are able to move all objects freely, to re-produce a new presentation. The proposed components are shown in Figure 10.

4 System Performance Discussion

In communication system, to connect both end parties, there are three main steps: (1) connection establishment, (2) media flow, and (3) connection tear down, as shown in Figure 10. After a connection is established, media flows (e.g. voice and video) are exchanged. SIP has been involved on step 1 and 3 which describes a procedure of creating a connection, terminal capability exchanging, and finally, how to

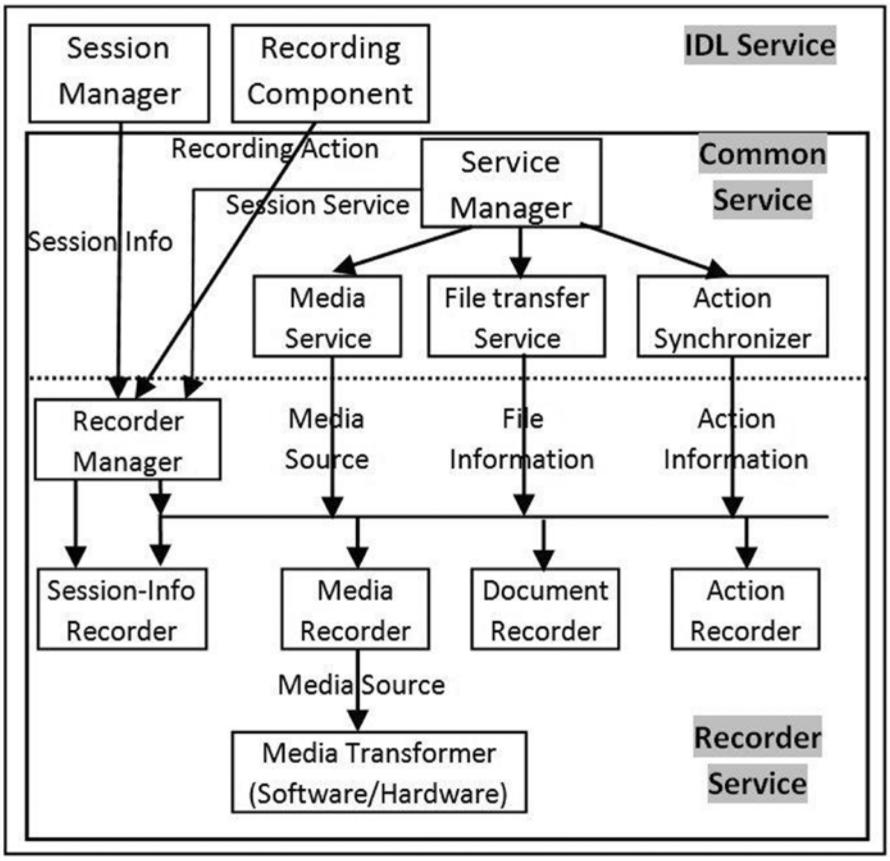


Figure 10　Multimedia Recorder (MR) components

terminal a connection. Some mobility issues are also concerned by the connection establishment, e.g. performance issues during hand-over [29]. We noted in [29], hand-over may be done separately between MIPv6 and SIP applications. In most cases, hand-over by SIP applications directly perform faster but create more complexity. Less expensive overhead, however, for a hand-over mechanism for a mobility node regardless of application capability has been proposed [30].

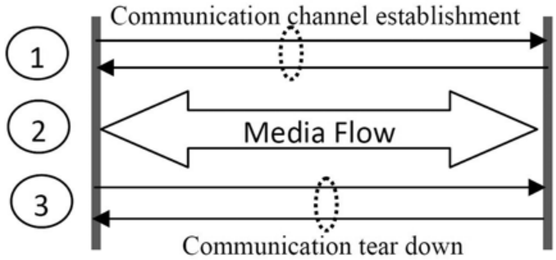


Figure 11 Three main steps of creating a communication: channel establishment, media flow, and tear down

SIP has been heavily investigated and widely deployed. There are research on the performance of SIP server [22], security [23], 3G and beyond [24][25]. It seems that SIP performs well in most cases in terms of light overhead, simplicity, less processing power, a large number of call handling, real time communication signal handling. We may conclude that SIP is a good signaling protocol to support a wide variety of communications. It can create rich features of multimedia communications.

Regarding to step 2 in Figure 11, the media flow, all media stream are exchanged in this step. When dealing with interactive communications, the main performance issues are: low delay, packet lost toleration, and less bandwidth consume. For low delay, it means that low end-to-end time delay is preferred to allow fast actions between all parties. Packet lost toleration means conversation is still maintained in some good quality event bandwidth is fluctuated. A number of voice and

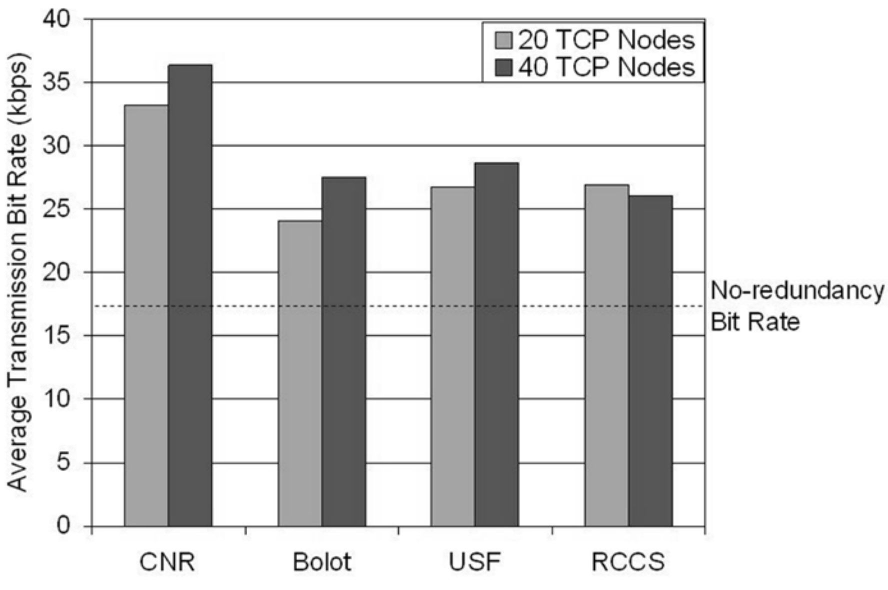


Figure 12 MPEG 4 throughput change when using media flow control mechanisms

video codec has been proposed, well studied and deployed. The performance does not only depend on voice and video codec performance but also a flow control mechanism [26][28]. Figure 12 [27] shows MPEG 4 throughput change when using media flow control mechanisms, where CNR is our proposed scheme. The performance also depends on a network transport mechanisms, e.g. RTP (Real Time Transport Protocol), STCP (Stream Transport Control Protocol) [31], DCCP (Datagram Congestion Control Protocol) [32]. The situation is more harder to handle when it is a multi-point communications especially when a call party is a mobile node.

In conclusion, this paper addresses a singling capability design to create a rich features of communications.

5 Future Works

The following topics are expected to carry out in our on- going works:

- P2P based SIP Conference

  P2P (Peer-to-Peer) based SIP VoIP is an alternative approach proposed recently. Most of today implementations are using the traditional SIP conference which is a centralized approach, while P2P is a de-centralized system. There is a number of literatures proposing P2P based SIP. In our on-going work, we will describe the architecture designed to construct P2P network collaborating with the existing traditional SIP for conferencing system [19]. Comparing to others, our proposed scheme is more attractive in terms of scalability (e.g. based on pure P2P), interoperability with traditional SIP, as well as interactive communication approach for all SIP clients (simultaneously interactive communications). However, media flow design is still a big issue when real-time communication is concerned. Moreover, security in P2P is still an issue.

- Convergence of communication services

  We are extending this work to UCC (Unified Communication and Collaboration). UCC is the integration of all kinds of communications into one single service. We consider IDL as a particular application of UCC. Based on our recently work, more features and services will be added, for example, we will allow a regular SIP phone system to call in, and it may be described as passive or active participants. Voice mail box as well as a part of CRM (Customer Relationship Management) will be added.

Table 1 Comparisons of P2P SIP-based features

Reference	Registration & Lookup	Call Management	Conference Management	Realtime Media	Non-realtime Media
[31]	SIP and P2P-SIP simultaneously	P2P-SIP	N/A	—	P2P-SIP
[34]	P2P-SIP	P2P-SIP	—	—	—
[35]	SIP and P2P-SIP simultaneously	P2P-SIP	—	—	—
[36]	P2P-SIP	P2P-SIP	—	—	—
[37]	P2P	SIP	—	—	—
[38]	N/A	N/A	N/A	ALM	—
[39]	P2P-SIP	P2P-SIP	—	—	—
Proposed	SIP	SIP	P2P-SIP	ALM	P2P-SIP

• Working on multiplatform and multi-devices for ubiquitous application needs to work on seamless access in any-where and any-time. There are challenging issues need to be solved especially when dealing with multimedia communications, e.g., bandwidth limitation and fluctuation, bandwidth mis-match, device capability (e.g. display, computer power, energy). For example, voice and video transcoding is needed if the bandwidth in some links is too small. Several techniques of overlay multicast may help for this problem.

6 Conclusion

In this paper, we have presented some extensions of SIP based conference scenarios. System architect and signal flows have been described. We have proposed two additional components to enhance conference features: Conference Manager Server (CMS) and Conference Repository (CR). Based on the proposed framework, in our implementation, we have demonstrated some services. Recently, we are deploying these scenarios for SIP based e-learning applications, called IDL: Interactive Distance Learning. We believe that the proposed scenarios will make SIP in more effectively use. We noted in the system performance analysis where this work concerns on signaling flow, how to create a rich feature of communications. Some future works are discussed to extend this development. For example, deploying SIP application over P2P overlay network, extending to UCC (Unified Communication and Collaboration), and working on multi-platform and devices.

Acknowledgement

This work was partly supported by

• Multimedia Architecture and Applications Based on SIP Project (MySIP), STIC-Asia Program 2006.

• Faculty of Engineering, Prince of Songkla University, Hat-Yai, Thailand.

• UniNet, Commission of Higher Education, Ministry of Education, Thailand.

References

[1] Vineet Kumar, Markku Korpi and Senthil Sengoodan, “IP Telephony with H.323”, Wiley, 2001.

[2] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, “SIP: Session Initiation Protocol”, RFC 3261, Internet Engineering Task Force, June 2002.

[3] K. Katrinis, G. Parissidis, and B. Plattner (Switzerland), “A Comparison of Framework for Multimedia Conferencing: SIP and H.323”.

[4] Glasmann J., W. Kellerer, and H. Muller, “Service Architectures in H.323 and SIP: A Comparison,” In *IEEE Communications Society Surveys and Tutorials,* 2003.

[5] J. Rosenberg, “A Framework for Conferencing with the Session Initiation Protocol (SIP),” RFC 4353, Internet Engineering Task Force, February 2006.

[6] Session Initiation Proposal Investigation (sipping), http://www.ietf.org/html.charters/sipping-charter.html

[7] Centralized Conferencing (xcon), http://www.ietf.org/html.charters/xcon-charter.html

[8] Even & Ismail, “Conferencing Scenarios,” RFC 4597, Internet Engineering Task Force, July 2006.

[9] A. Johnston, O. Levin, “Session Initiation Protocol (SIP) Call Control - Conferencing for User Agents,” RFC 4579, Internet Engineering Task Force, August 2006.

[10] K. Zeilenga, “Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map,” RFC4510, Internet Engineering Task Force, June 2006.

[11] J. Van Dyke, E. Burger, “Media Server Control Markup Language (MSCML) and Protocol”, RFC 4722, Internet Engineering Task Force, November 2006.

[12] J. Rosenberg, “The Extensible Markup Language (XML) Configuration Access Protocol”, Internet draft <draft-ietf-simple-xcap-12>, October 13, 2006.

[13] J. Rosenberg,“ A Session Initiation Protocol (SIP) Event Package for Registrations”, RFC 3680, Internet Engineering Task Force, March 2004.

[14] Eunsook Kim, Jong-Hwa Yi, Shin-Gak Kang, and Kishik Park, “Performance Evaluation of an XML-Based Conference Control Protocol for Centralized VoIP Conference,” *ETRI Journal*, vol.26, no.2, Apr. 2004, pp.181-184.

[15] Koskelainen, P., Schulzrinne, H. and X. Wu, “A sip-based conference control framework”, Nossdav 2002, Miami Beach, May 2002.

[16] S. Sae-Wong, T. Kamolphiwong, S. Kamolphiwong, and N. Wittayasirikul, “Interactive Distance Learning based on SIP”, *IJCIM (International Journal of the Computer, the Internet and Management)*, Special issue, the 3rd International Conference on e-learning for Knowledge-based Society, Vol 14, No. SP1, Aug 2006, pp.33.1-33.6.

[17] N. Wittayasirikul,P. Boonlert, S. Kamolphiwong, S. Sea-Wong, and T. Kamolphiwong, “Enhancements of IP Conference Scenarios”, In *Proceedings of ACM Mobility Conference*, September 10-11, 2007, Singapore, pp 359-365.

[18] P. Tangsermsook, S. Kamolphiwong, and S. Sae-Wong, “Reliable UDP Multicast in E-learning Applications”, In *Proceedings of PECC-2007*, Phuket, Thailand, May 10-11, 2007.

[19] Wasin, S. Sea-Wong, T. Kamolphiwong, and S. Kamolphiwong, “Mobile E-Learning: Interactive Distance Learning Framework for Mobile Devices”, In *Proceedings of IECC-2007*, Phuket, Thailand. May 10-11, 2007.

[20] Centre for Network Research Wiki, https://www.cnr.psu.ac.th/wiki/index.php/Main_Page, (in Thai), July 2008

[21] K. Tirasoontorn, S. Kamolphiwong, S. Sae-Wong, “Distributed P2P-SIP Conference Construction”, to appear on *Proceedings of ACM Mobility Conference 2008*, Taiwan, September 10-12, 2008.

[22] Performance Results for OpenSER and SIP Express Router, http://www.transnexus.com/White%20Papers/OpenSER-SER_Comparison.htm

[23] Eun-Chul Cha; Hyoung-Kee Choi; Sung-Jae Cho; “Evaluation of Security Protocols for the Session Initiation Protocol”, In *Proceedings of 16th Computer Communications and Networks*, 2007 (ICCCN 2007).

[24] Maria Isabel Pous1, Dirk Pesch, and Gerry Foster, “SIP-based Applications in UMTS: A Performance Analysis”, The Fifth European Wireless Conference Mobile and Wireless Systems beyond 3G, February 24-27, 2004, Barcelona, Spain.

[25] Dirk Pescha, Corresponding Author Contact Information, E-mail The Corresponding Author, Maria Isabel Pousa and Gerry Fosterb, “Performance evaluation of SIP-based multimedia services in UMTS”, *Computer Networks*, Volume 49, Issue 3, 19 October 2005, pp. 385-403.

[26] S. Chuaywong, S. Kamolphiwong, T. Kamolphiwong, “Adaptive Quality Control for Real-time MPEG-4 Video Communications,” In *Proceedings of the 5th International Symposium on Communications and Information Technology (ISCIT 2005)*, October. 2005, Beijing, China, pp.299-303.

[27] S. Chuaywong, S. Kamolphiwong, T. Kamolphiwong, R. Elz and S. Sae-Wong, “Adaptive quality control for multimedia communications,” *Songklanakarin J. Sci. Technology*, V 30 (1), Jan - Feb 2008, pp.71-84.

[28] S. Chuaywong, S. Kamolphiwong, T. Kamolphiwong, “Performance Enhancement of FEC Mechanism for IP Telephony”, In *Proceedings of ECTI-CON 2005*, Vol.2, May. 2005, Pattaya, Thailand, pp.697-700.

[29] J. Tepsoontorn, S. Kamolphiwong, T. Kamolphiwong, and R. Elz, “Enhanced SIP Mobility System with MIPv6,” In *Proceedings of the 3rd International Symposium on Communications and Information Technology (ISCIT2003)*, Thailand, September 2003, pp.937-942.

[30] R. Elz, and S. Kamolphiwong, “Toward Universal Mobile IP”, In *Proceedings of IPv6 workshop, Mobility Conference 2008*, Taiwan, September 12, 2008.

[31] R. Stewart, at. el, RFC 2960, Stream Control Transmission Protocol, http://www.ietf.org/rfc/rfc2960.txt

[32] Eddie Kohler. Mark Handley, Sally Floyd, “Designing DCCP: Congestion Control without Reliability”, SIGCOMM’06, September 11-15, 2006, Pisa, Italy.

[33] Kundan Singh and Henning Schulzrinne, “Peer-to-peer Internet Telephony using SIP”, NOSSDAV Skamania, Washington, June 2005.

[34] D. A. Bryan, B. B. Lowekamp, and C. Jenning, “SOSIMPLE: A Serverless Standard-based, P2P SIP Communication System”, In *Technical report*, AAA-IDA 2005.

[35] Fan Pu, P2P architecture for IP telephony using SIP. Mobile Communities Seminar on Internetworking, Spring 2006. http://www.tml.tkk.fi/Publications/C/21/Pu_ready.pdf

[36] Hongyan Ma, Baomin Xu, Huaxu Wan, Chunyan Li, A hierarchical P2P architecture for SIP communication. NGMAST '07.

[37] Kundan Singh and Henning Schulzrinne, “Using an External DHT as a SIP Location Service”, In *Columbia University Technical Report*, CUCS-007-06, New York, NY, Feb 2006.

[38] A. Rios, A. J. Gonzáles, A. Oller, J. López, and J. Alcober, “P2P Multipoint Conference System using SIP”, HET-NETs '08.

[39] Mo Yijun, Wang Fei, Huang Benxiong, Luo Fu, “Real-time Multimedia Services Using Hybrid P2P SIP”, ISCIT '06.