Instant Messaging Sessions within a Centralized Conferencing (XCON) System
draft-boulton-xcon-msrp-conferencing-05

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on January 10, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

The document "A Framework and Data Model for Centralized Conferencing" defines a centralized conference as both signaling and protocol agnostic. The primary examples within this framework focus on audio and video as the media types for the session. This document describes the mechanisms, in the context of this centralized conferencing framework, when using instant messaging sessions as the
media.

Table of Contents

1. Introduction ............................................. 3
2. Conventions and Terminology ............................. 3
3. Overview .................................................. 3
   3.1. Protocol Operations ................................. 5
   3.2. IM and Conferencing Identifiers .................. 5
4. Basic Operations ......................................... 7
   4.1. Creating a Chat Room ............................... 8
   4.2. Joining a Chat Room ............................... 10
   4.3. Deleteing a Chat Room ............................. 12
5. Advanced Operations .................................... 13
   5.1. Text Sidebar ....................................... 13
   5.2. Private Message .................................. 16
6. Security Considerations ................................. 19
7. Acknowledgements ........................................ 20
8. Informative References .................................. 20
Authors' Addresses ........................................... 21
Intellectual Property and Copyright Statements .......... 22
1. Introduction

A Centralized Conference as defined by the "A Framework for Centralized Conferencing" [1] is both signaling and protocol agnostic. The primary examples within the framework focus on audio and video as the media types for the session. This document describes the mechanisms and associated framework elements involved when instant messages are the media for the conference. This functionality is often referred to as a "chat room" or simply "chat" as it provides the text equivalent of a voice conversation involving multiple parties.

Several existing protocols support this chat functionality, such as Internet Relay Chat (IRC) [3] and Extensible Messaging and Presence Protocol (XMPP) [4]. In addition, [8] provides IM chat functionality for a purely SIP signaling based solution option using Message Session Relay Protocol MSRP [6]. The focus of this document is to describe the interface to a conferencing system based on the XCON framework, independent of the specific IM media type used by the client. The intent is not replace the existing IM protocols, but rather to allow clients using the different protocols, that also use the centralized conferencing framework model for voice and audio conferences, to make use of the same conference control mechanisms and conferencing system to establish, update and delete the chat room. While this approach also allows the conferencing system to provide a natural interworking point for various IM protocols, the details of the interworking are outside the scope of this document.

2. Conventions and Terminology

This document reuses the terminology defined in "A Framework for Centralized Conferencing" and the protocol operations defined in the Centralized Conferencing Protocol document [ref:TBD].

3. Overview

Figure 1 provides a general illustration of IM clients having a direct, 1:1 connection to the conferencing system. The conferencing system receives any IMs sent by the clients and then distributes them to the appropriate IM sessions.
Figure 1: Client Connection

The approach in this document is to have no impact on the existing IM protocols, while taking full advantage of the functionality provided by the centralized conferencing framework. The solution proposal in this document meets the requirements identified in the requirements section of the multiparty IM with MSRP document [8], with the exception of the requirements (REQ-5, REQ-6 and REQ-7) related to the concept of nicknames. A solution for nicknames is currently not within the scope of the centralized conferencing framework or associated protocol documents, although it is a useful and desirable concept. [Editor's Note: Should the generic requirements from the simple-chat draft be pulled into a separate document (as they had been in the past) or is it okay for this doc to reference those
requirements, which are for the most part quite generic?].

A basic solution for IM chat sessions, also meeting the Multiparty IM using MSRP requirements, is documented in [8]. It uses the concept of an "MSRP switch" as the centralized component, whose role is very similar to the MSRP Conferencing Server in this document. The solution in [8] doesn't explicitly take advantage of the centralized conferencing framework model, as it primarily intends to make use of the basic SIP conferencing framework to provide the basic chat functionality. However, that solution approach is compatible with the solution components described in this document, with no impact on that basic solution proposal. One of the advantages of applying the two solutions in concert would be a reuse of the centralized conferencing framework model for advanced features, such as sidebars and private conferences, and manipulation of the conference data.

### 3.1. Protocol Operations

An IM client wishing to join a conference uses standardized centralized conferencing mechanisms for creating and joining a conference, as identified in the centralized conferencing framework and related protocol documents.

The request to send an IM to an IM media session is specific to the IM protocol (e.g., MSRP SEND). On issuing an request to send an IM to an IM media session that is a member of a conference instance, the IM will be replicated and forwarded, in the relevant context, to all other IM media sessions that are participants of the conference instance.

An IM client wishing to delete a chat room uses standardized mechanisms for deleting a conference. Non-signaling specific mechanisms are defined in the Centralized Conferencing Framework [1] and related protocol document [TBD], with protocol specific mechanisms defined in other documents such as for SIP in the SIPPING Conference Framework [5].

### 3.2. IM and Conferencing Identifiers

As mentioned in the overview, an IM client connecting to a conferencing system has a 1:1 relationship with the IM signaling entity, each having a unique protocol specific session ID. When referring to IM session ID's the document is making reference to the locally (at conferencing system) generated session ID used for IM session signaling identification. In the case of MSRP, this session ID is inserted into the local path SDP attribute. An important concept in this proposal is the creation and management of IM sessions. It is important that each IM session created, as
identified by a unique IM session ID, is explicitly tied to an associated conference, represented by the conference identifier (as defined in the Centralized Conferencing Framework [1]). This provides the relevant association between IM session and an XCON Conference. A generic example representation is illustrated by the rows contained in Figure 2.

<table>
<thead>
<tr>
<th>Conference Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM Session ID=8asjdhk</td>
</tr>
<tr>
<td>IM Session ID=38iuhsds</td>
</tr>
<tr>
<td>IM Session ID=djiowid</td>
</tr>
<tr>
<td>IM Session ID=389hewu</td>
</tr>
</tbody>
</table>

Figure 2: Simple Session Association

The Centralized Conferencing Framework[1] introduces the concept of a conference user identifier defined in [2]. When a user joins a conference instance through the signaling protocol, it is allocated an appropriate conference user identifier either through authentication or system allocation. The conference user identifier MUST be used in conjunction with an IM session identifier to internally represent a participant in a conference instance. Figure 2 is then expanded to look like Figure 3. Again a row in the table representing a single entry.

<table>
<thead>
<tr>
<th>Conference Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM Session ID=8asjdhk</td>
</tr>
<tr>
<td>IM Session ID=38iuhsds</td>
</tr>
<tr>
<td>IM Session ID=djiowid</td>
</tr>
<tr>
<td>IM Session ID=389hewu</td>
</tr>
</tbody>
</table>

Figure 3: Advanced Session Association

A more complex session association is necessary due to potential for a user to have multiple IM sessions in a single conference instance, such as multi-lingual conference support. In an example with SIP and MSRP, the conference representation in Figure 3 allows for such functionality when separate SIP dialogs represent MSRP sessions.
This process becomes complex when multiple SDP MSRP media sessions (m=) are defined in a single payload. This internal representation now needs expanding to enable a conferencing system to explicitly associate a media session (m=). This involves including the media label, as defined in [9], to maintain the internal conference association. An example is illustrated in Figure 4.

<table>
<thead>
<tr>
<th>Conference Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSRP Session ID=8asjdhk</td>
</tr>
<tr>
<td>MSRP Session ID=38iuuhds</td>
</tr>
<tr>
<td>MSRP Session ID=838unaH</td>
</tr>
<tr>
<td>MSRP Session ID=djiowid</td>
</tr>
<tr>
<td>MSRP Session ID=389hewu</td>
</tr>
<tr>
<td>MSRP Session ID=Ko03jdk</td>
</tr>
</tbody>
</table>

Figure 4: Advanced Session Association + Media Label

In Figure 4 conference user identifiers '0283hHu' and 'pakdj7H' appear twice. The combination of multiple conference user identifiers and a unique MSRP session ID enables the conference system to clearly identify a specific MSRP instance. The representation also includes the media label, as defined in [9], for identification purposes. This added property, which is extracted from the SDP media line, enables clear identification when multi SDP media (m=) lines appear in the same SDP payload.

A client MUST include the media label attribute defined in [9] when including multiple MSRP sessions in the same SDP payload. Even in the simplest conferencing system, where users are allowed to enter anonymously, the internal representation described in this section should be observed. In this case, the conferencing system would still internally create a conference user identifier for participant reference purposes.

4. Basic Operations

This section provides details of the realization of the Multi-party IM (chat) within the context of the centralized conferencing framework. A brief discussion and diagrams are provided for creating, joining and deleting a chat based conference. The objective of this section is to further illustrate the model, mechanisms and protocols presented in the previous sections and also
serves to validate that the model, mechanisms and protocols are sufficient to support IM chat.

The scenarios provide a high level primitive view of the necessary operations and general logic flow. The details shown in the scenarios are for illustrative purposes only and don't necessarily reflect the actual structure of the conference control protocol messages nor the detailed data, including states, which are defined in separate documents. It should be noted that not all entities impacted by the request are shown in the diagram (e.g., Focus), but rather the emphasis is on the new entities introduced by this centralized conferencing framework. [Editor's Note: Ideally, this section will be updated with details once the protocol is agreed.]

4.1. Creating a Chat Room

There are different ways to create a conference. A participant can create a conference using call signaling means only, such as SIP and detailed in [10]. For a conferencing client to have more flexibility in defining the characterisitics and capabilities of a chat based conference, a conferencing client would implement a conference control protocol client. By using a conference control protocol, the client can determine the capabilities of a conferencing system and its various resources.

Figure 5 provides an example of one client "Alice" determining the conference blueprints available to support various types of chat rooms for a particular conferencing system and creating a chat based conference using the desired blueprint.
Figure 5: Client Creation of Chat room

Upon receipt of the Conference Control Protocol request for blueprints associated with chat rooms, the conferencing system would first authenticate "Alice" (and allocate a conference user...
identifier, if necessary) and then ensure that "Alice" has the appropriate authority based on system policies to receive any chat room based blueprints supported by that system. Any blueprints that "Alice" is authorized to use are returned in a response, along with the conference user ID.

Upon receipt of the Conference Control Protocol response containing the blueprints, "Alice" determines which blueprint to use for the conference to be created. "Alice" creates a conference object based on the blueprint (i.e., clones) and modifies applicable fields, such as membership list, topic details, and start time. "Alice" then sends a request to the conferencing system to create a conference reservation based upon the updated blueprint.

Upon receipt of the Conference Control Protocol request to "create" a conference based upon the blueprint in the request, the conferencing system ensures that the blueprint received is a valid blueprint (i.e. the values of the various field are within range). The conferencing system determines the appropriate read/write access of any users to be added to a conference based on this blueprint (using membership, roles, etc.). The conferencing system uses the received blueprint to clone a conference reservation. The conferencing system also reserves or allocates a conference ID to be used for any subsequent protocol requests from any of the members of the conference. The conferencing system maintains the mapping between this conference ID and the conference object ID associated with the reservation through the conference instance.

Upon receipt of the conference control protocol response to reserve the conference, "Alice" now creates an active chat room using that reservation. "Alice" provides the conference information, including the necessary conference ID, to desired participants to allow them to join the chat room. "Alice" may also add other users to the chat room. When the first participant, including "Alice", requests to be added to the conference, an active conference and focus are created. The focus is associated with the conference ID received in the request.

4.2. Joining a Chat Room

A participant can join and leave the conference using call signaling means only, such as SIP. However, in order to perform richer conference control a user client can implement a conference control protocol client. By using a conference control protocol, the client can affect its own state and the state of other participants, depending upon policies, which may indirectly affect the state of any of the conference participants.
In the example in section Section 4.1, "Alice" has reserved a chat room. "Alice" has also already joined the conference and made the chat room active. "Alice" can either add additional participants to the chat room or provide the conference information, including the necessary conference ID, to desired participants and allow them to request to join themselves. Any participants that have the authority to manipulate the conference would receive the conference object identifier of the active conference object in the response to their request to join.

Figure 6 provides an example of "Bob" joining the chat room using the conference ID provided by "Alice" (e.g., in an IM).

```
+--------------------------------+   | Conferencing System |
| "Bob"                           |   +-------------------++---+|
| ++-------++                   |   |                   |
| |                    |     |                   |
| | CCP Request        |     |                   |
| | <-------------------|     |                   |
| | Client              |     |                   |
| | -------------------+--->|Conference| | Active |
| | | Conference Object ID, | |Control | -->| Chat |
| | | Change, Member="Bob" | |Server | | Conference |
| | | <-------------------+| |                   |
| | | "Alice"            |   +-------------------++---+|
| | ++-------++       |   |                   |
| | | NOTIFY <"Bob"="added"> |   |                   |
| | | <------------------- |   |                   |
| | | Client              |   |                   |
| | | -------------------+--->|Notification|<~~~|     |
| | |                     |   |                   |
| | | ++-------++ | "Bob" |   |                   |
| | | | NOTIFY <"Bob"="added"> |   |                   |
| | | |                     |   |                   |
| | +-------------------++---+|
```

Figure 6: Joining a chat room

Upon receipt of the Conference Control Protocol request to "add" a party ("Bob") in the specific conference as identified by the conference object ID, the conferencing system must determine whether "Bob" is already a user of this conferencing system or whether he is a new user. If "Bob" is a new user for this conferencing system, a Conference User Identifier is created for Bob. The conferencing system must also ensure that "Bob" has the appropriate authority based on the policies associated with that specific conference object to perform the operation.
Once "Bob" has been successfully added to the chat room, per updates to the state, and depending upon the policies, other participants (including "Bob") may be notified of the addition of "Bob" to the conference via the Conference Notification Service.

### 4.3. Deleteing a Chat Room

Depending upon the conferencing system policies and policies specific to the chat room, the creator of the chat would typically be the participant authorized to delete the chat room.

In the example in section **Section 4.1**, "Alice" has created a chat room and provided the conference information, including the necessary conference ID, to desired participants and allow them to request to join themselves. "Bob" and others are participants in the chat. Figure 6 provides an example of "Alice" later deleting this same chat room.

![Figure 6: Deleting a chat room](image)

Upon receipt of the Conference Control Protocol request to "delete" the specific chat room as identified by the conference object ID, the conferencing system must determine whether "Alice" has the authority to delete this conference. Since "Alice" is the creator of the conference, the "delete" operation is performed, with the appropriate signalling sent to the participants.

![Figure 7: Deleting a chat room](image)
One step in the deletion of the chat room may include notifying the participants (including "Bob") that they have been removed via the Conference Notification Service.

5. Advanced Operations

This section provides details of the realization of advanced chat features, such as sidebars and private messages, within the context of the centralized conferencing framework. As with Section 4, the objective of this section is to further illustrate the model, mechanisms and protocols presented in the previous sections and also serves to validate that the model, mechanisms and protocols are sufficient to support advanced IM chat features.

5.1. Text Sidebar

The Multi-party IM using MSRP document [8] identifies the requirement (REQ-GEN-12) to set up a sidebar conference with one or more participants of the conference.

The concept of a 'sidebar' in an XCON compliant conference system is fully described in the Sidebar section and related subsections within the Conferencing Scenarios Realization section of the centralized conferencing framework document [1]. The creation, manipulation and deletion of sidebars for MSRP based sessions follows the same principles.

A conference object representing a sidebar is created by cloning the parent associated with the existing conference and updating any information specific to the sidebar. A sidebar conference object is implicitly linked to the parent conference object (i.e. it is not an independent object) and is associated with the parent conference object identifier. A conferencing system manages and enforces the parent and appropriate localized restrictions on the sidebar conference object (e.g., no members from outside the parent conference instance can join, sidebar conference can not exist if parent conference is terminated, etc.).

Figure 8 provides an example of one client "Alice" involved in active chat conference with "Bob" and "Carol". "Alice" wants to create a sidebar to have a side discussion with "Bob" while still receiving the session based messaging associated with the main chat conference. Whether the text is interleaved with the main chat or whether a separate window is created for the sidebar is implementation specific. "Alice" initiates the sidebar by sending a request to the conferencing system to create a conference chat reservation based upon the active chat conference object. "Alice" and "Bob" would
remain on the roster of the main conference, such that other participants could be aware of their participation in the main conference, while the text sidebar conference is occurring.
Figure 8: Client Creation of a Sidebar Conference

Upon receipt of the Conference Control Protocol request to "reserve" a new sidebar chat conference, based upon the active chat conference received in the request, the conferencing system uses the received active chat conference to clone a conference chat reservation for the sidebar. As discussed previously, the sidebar reservation is NOT independent of the active conference (i.e., parent). The conferencing system also reserves or allocates a conference ID to be used for any subsequent protocol requests from any of the members of the conference. The conferencing system maintains the mapping between this conference ID and the conference object ID associated with the sidebar reservation through the conference instance.

Upon receipt of the conference control protocol response to reserve the conference, "Alice" can now create an active chat conference using that reservation or create additional reservations based upon the existing reservations. In this example, "Alice" wants only "Bob" to be involved in the sidebar, thus she manipulates the membership. "Alice" also only wants the text from the original conference, but wants the text within the sidebar to be restricted to the participants in the sidebar. "Alice" sends a conference control protocol request to update the information in the reservation and to create an active conference.

Upon receipt of the conference control protocol request to update the reservation and to create an active chat conference for the sidebar, as identified by the conference object ID, the conferencing system ensures that "Alice" has the appropriate authority based on the policies associated with that specific conference object to perform the operation. The conferencing system must also validate the updated information in the reservation, ensuring that a member like "Bob" is already a user of this conferencing system.

Depending upon the policies, the initiator of the request (i.e., "Alice") and the participants in the sidebar (i.e., "Bob") may be notified of his addition to the sidebar via the conference notification service.

5.2. Private Message

The case of private messages can be handled as a sidebar with just two participants, identical to the example in section Section 5.1. The other context, referred to as whisper, in this document refers to situations involving one time media targeted to specific user(s). An example of a whisper would be a text message injected only to the conference chair or to a new participant joining a conference.
Figure 9 provides an example of one user "Alice" whose chairing a fixed length conference with "Bob" and "Carol". The configuration is such that only the chair is providing a warning when there is only 10 minutes left in the conference. At that time, "Alice" is moved into a sidebar created by the conferencing system and only "Alice" receives that text message announcing the 10 minute warning.
"Alice"

+----------+
| NOTIFY <"Alice"=added, |
| Client |
| activeChatSideConfObjID, |
| confID > |

---Text message delivered to "Alice"---

"Alice"

+----------+
| NOTIFY <"Alice"=removed, |
| Client |
| activeChatSideConfObjID, |
| confID > |
When the conferencing system determines that there is only 10 minutes left in the conference which "Alice" is chairing, rather than creating a reservation as was done for the sidebar in Section 5.1, the conferencing system directly creates an active chat sidebar conference, based on the active chat conference associated with "Alice". As discussed previously, the sidebar conference is NOT independent of the active conference (i.e., parent). The conferencing system also allocates a conference ID to be used for any subsequent manipulations of the sidebar chat conference. The conferencing system maintains the mapping between this conference ID and the conference object ID associated with the active sidebar conference through the conference instance.

Immediately upon creation of the active chat sidebar conference, the text announcement is provided to "Alice". Depending upon the policies, Alice may be notified of her addition to the sidebar via the conference notification service. "Alice" continues to receive the text messages from the main conference.

Upon delivery of the text announcement, "Alice" is removed from the sidebar and the sidebar conference is deleted. Depending upon the policies, "Alice" may be notified of her removal from the sidebar via the conference notification service.

6. Security Considerations

As discussed in the XCON Framework, there are a wide variety of potential attacks related to conferencing, due to the natural involvement of multiple endpoints and the many, often user-invoked, capabilities provided by the conferencing system. Examples of attacks in the context of MSRP conferencing would include the following: an endpoint attempting to receive the messages for conferences in which it is not authorized to participate, an endpoint attempting to disconnect other users, and theft of service, by an endpoint, in attempting to create conferences it is not allowed to create.

Since this document describes the use of existing protocols (e.g. MSRP, Conference Control Protocol, SIP, etc.), it also re-uses the security solutions for those protocols and the associated authorization mechanisms. Since this solution makes use of the XCON framework, it makes use of the policy associated with the conference object to ensure that only authorized entities are able to manipulate the data to access the capabilities. This solution also makes use of the privacy and security of the identity of a user in the conference,
as discussed in the XCON Framework.

7. Acknowledgements

The authors appreciate the input and comments from Miguel Garcia-Martin and Dave Morgan.

8. Informative References


Authors' Addresses

Chris Boulton
Ubiquity Software Corporation
Building 3
Wern Fawr Lane
St Mellons
Cardiff, South Wales  CF3 5EA

Email: cboulton@ubiquitysoftware.com

Mary Barnes
Nortel
2201 Lakeside Blvd
Richardson, TX

Email: mary.barnes@nortel.com