

DISPATCH Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 13, 2014

M. Barnes
Polycom
C. Boulton
NS-Technologies
S. Loreto
Ericsson
July 12, 2013

Chatrooms within a Centralized Conferencing (XCON) System
draft-boulton-dispatch-session-chat-01

Abstract

The document "A Framework 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 provides an overview of the mechanisms defined in the centralized conferencing framework that can be used to support multi-user chat. In addition, the document describes additional functionality and requirements necessary to provide feature rich functionality.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on January 13, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
2.	Conventions and Terminology	3
3.	Overview	4
3.1.	Basic Protocol Operations	5
3.2.	Chat Session and Conferencing Identifiers	7
4.	Advanced Operations	8
5.	Additional Operations	8
5.1.	Nicknames	9
5.2.	Logging	10
5.3.	History	11
5.4.	Indicating Alternate Venue	11
5.5.	File Transfer	11
5.6.	Real Time Collaboration	12
6.	Security Considerations	12
7.	IANA Considerations	13
8.	Acknowledgments	13
9.	References	13
9.1.	Normative References	13
9.2.	Informative References	13
	Authors' Addresses	15

1. Introduction

A Centralized Conference as defined by the "A Framework for Centralized Conferencing" (XCON Framework) [[RFC5239](#)] 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 provides an overview of the mechanisms and associated framework elements involved when text is the media for the conference. This functionality is often referred to as a "multi-user chat" as it enables a participant to join a chatroom (e.g. hosted by the conference server) for the exchange of messages between multiple participants. The message can be plain text or can contain different format for more advanced functionality.

Several existing protocols support this multi-user chat functionality, such as Extensible Messaging and Presence Protocol (XMPP) [[RFC6120](#)], [[RFC6121](#)] and Internet Relay Chat (IRC) defined in [[RFC1459](#)] and its successors: [[RFC2810](#)], [[RFC2811](#)], [[RFC2812](#)], [[RFC2813](#)]. In addition, [[I-D.ietf-simple-chat](#)] provides multi-user chat functionality for a purely SIP signaling based solution option using Message Session Relay Protocol (MSRP) [[RFC4975](#)].

The focus of this document is to describe the interface and provide guidelines for the the support of existing multi-user chat functionality on a conferencing system based on the XCON framework using the Conference Control Manipulation Protocol (CCMP) independent of the specific media type used by the chat client.

The functionality described in this document is not intended to replace any of the existing chat protocols, nor is it specifying a new chat protocol. The motivation for this document is to allow clients that use the conferencing framework model for other media types (e.g. voice/video) to utilize the same conference control mechanisms and conferencing system to establish, update and delete a chatroom associated with a conference instance, independent of the chat protocol. This approach also allows the conferencing system to provide a natural interworking point for various chat protocols - the details of the interworking are outside the scope of this document.

2. Conventions and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)].

This document reuses the terminology defined in the Centralized Conferencing Framework [[RFC5239](#)] and related protocol document

[[RFC6503](#)].

Additional terminology used in this document:

Chat Client: a Conferencing Client as defined in [[RFC5239](#)] that participates in a "chatroom".

Chatroom: A virtual space that users figuratively enter in order to participate in real-time, text-based conferencing with other users.

Multi-user chat: The functionality that allows multiple users to exchange messages in the context of a room or channel, similar to Internet Relay Chat (IRC).

Private message: A message sent from one participant directly to another participant - i.e., not to the chatroom itself to all participants.

3. Overview

Figure 1 provides a general illustration of chat clients having a direct, 1:1 connection to the conferencing system. Participants can use the chat clients to join a room associated with a conference instance and send messages. The conferencing system receives the messages sent from a client participating in a conference instance and then distributes them to the other clients associated with the conference instance, that are also present in the chatroom.

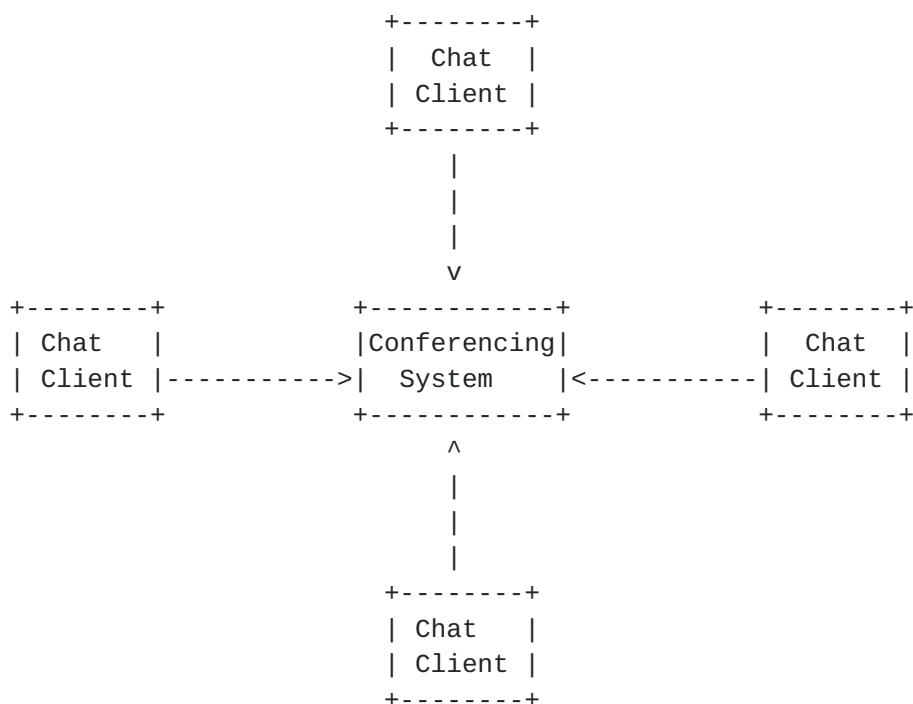


Figure 1: Client Connection

The approach in this document is to have no impact on the existing chat protocols, while taking full advantage of the functionality provided by the centralized conferencing framework.

A basic solution for MSRP based IM chat sessions is documented in [\[I-D.ietf-simple-chat\]](#). It uses the concept of an "MSRP switch" as the centralized component, whose role is very similar to the Conferencing Server in this document. The solution in [\[I-D.ietf-simple-chat\]](#) 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. The MSRP based IM chat solution 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 to take advantage of the centralized conferencing framework model for advanced features, such as sidebars and private conferences, and manipulation of the conference data.

XMPP assumes a decentralized client-server architecture similar to the one shown in Figure 1, wherein a client utilizing XMPP accesses a server and servers can also communicate with each other over TCP connections, similar to the email network. The XMPP server can provide as additional functionality the multi-user conferencing services [\[XEP-0045\]](#). The XMPP multi-user conferencing service is also compatible with the solution components described in this document with no impact on the basic solution proposal. Indeed, the centralized conferencing framework model is perfectly able to manage the XMPP strong room control model, including the ability to kick and ban users, to name room moderators and administrators, to require membership or passwords in order to join the room. However it is worth noting that the centralized conferencing framework does not encompass the communication between servers, as XMPP does. Thus, for the solution proposal in this document, the XMPP client SHOULD only have a direct connection with the server hosting the chatroom instance, and federations between servers SHOULD NOT be allowed.

[3.1.](#) Basic Protocol Operations

The multi-user chat protocol operations, such as create, join and delete can be performed using both non-signaling specific mechanisms or protocol specific mechanisms, if defined. Non-signaling specific mechanisms are defined in the Centralized Conferencing Framework [\[RFC5239\]](#) and related Conference Control Manipulation Protocol (CCMP) document [\[RFC6503\]](#). This document provides the details for the non-signaling specific mechanisms using CCMP with detailed examples

provided in [[RFC6504](#)]. Protocol specific mechanisms are defined in other documents such as for SIP in the SIPING Conference Framework [[RFC4353](#)] and for XMPP in Multi-User Chat [[XEP-0045](#)].

The privilege to create a chatroom associated with a conference instance can be restricted to certain users or can be reserved to an administrator of the conference. The room creation can be performed using non-signaling mechanism or protocol specific mechanism if defined. In the case of CCMP, a confRequest message with a "create" operation is sent by the chat client.

A participant can query the conferencing system to discover the list of the chat rooms associated with a hosted conference instance. In the case of CCMP, a blueprintsRequest message for the chatrooms supported by a conferencing system or a confsRequest message for the active chatrooms can be sent by the chat client.

In order to participate in the discussions held in a multi-user chat room, a participant MUST first enter the room. A chat client wishing to enter a chatroom associated with a conference instance MAY use a non-signaling or protocol specific mechanism if defined. In the case of CCMP, a participant MUST join a conference instance using the mechanisms which are described in [[RFC6503](#)] - e.g., userRequest message with a "create" operation to be added to a conference instance.

The request to send a message is specific to the chat protocol (e.g., MSRP SEND). Upon receipt of a request to send a message, the conferencing system replicates and forwards the message to all other chat clients that are participants of the chat room. Depending upon policy, a conferencing system MAY ignore or reject messages, in which case they are not distributed to the other chat room participants.

A participant MAY send a "private message" to a selected participant or a group of participant. This privilege SHOULD be allowed for all participants unless local policy dictates otherwise.

A chat client wishing to exit a chat room uses a non-signaling mechanism or protocol specific mechanism, if defined. If the chat client is the last to exit, the conferencing system can be responsible for deleting the room or the originator/owner/moderator. The privilege to delete a chatroom associated with a conference instance can be restricted to certain users or can be reserved to an administrator of the conference. The room deletion can be performed using non-signaling mechanism or protocol specific mechanism if defined. In the case of CCMP, the client MUST send a CCMP confRequest message with an operation of "delete".

3.2. Chat Session and Conferencing Identifiers

As highlighted in the overview section, a chat client connecting to a conferencing system has a 1:1 relationship with the chat signaling entity, each having a unique protocol specific chat session identifier (chat-ID). When referring to chat-IDs the document is making reference to the locally (at conferencing system) generated chat-ID used for session signaling identification. An important concept in this proposal is the creation and management of Group Chat sessions. It is important that each chat session created, as identified by a unique chat-ID, is explicitly tied to an associated conference. The Centralized Conferencing Framework [RFC5239] introduces the concept of a conference user identifier which is defined in [RFC6501]. When a user joins a conference instance through the signaling protocol, the user is allocated an appropriate conference user identifier either through authentication or system allocation. The conference user identifier is represented by the 'entity' attribute of a <user> element in the <users> element in the conference information. The association of the chat-IDs is accomplished by including each of the chat-IDs in the conference information in the 'entity' attribute of an <endpoint> element in the <user> element. The conference information as a whole is uniquely identified within the conferencing system by an XCON-URI, thus providing the relevant association between a chat session and a centralized conference. Figure 2 shows the logical representation of the chat-IDs with the conf-userIDs, with each row in the table representing a single entry.

+-----+			
	XCON-URI: 711331@example.com		
+-----+			
	Chat-ID=8asjdhk	xcon-userid: 839ULjj	
	Chat-ID=38iuhds	xcon-userid: 0283hHu	
	Chat-ID=djiowid	xcon-userid: ncH37Hs	
	Chat-ID=389hewu	xcon-userid: pakdjH	
+-----+			

Figure 2: Session Association

A more complex session association is necessary due to potential for a user to have multiple group chats in a single conference instance, such as multi-lingual conference support. In an example with SIP and MSRP, the conference representation in Figure 2 allows for such functionality when separate SIP dialogs represent MSRP sessions.

This process becomes complex in the case that multiple SDP MSRP media sessions (m=) are defined in a single payload. This internal representation needs expanding to enable a conferencing system to explicitly associate a media session (m=). This involves including the media label, as defined in [\[RFC4574\]](#), to maintain the internal conference association. An example is illustrated in Figure 3.

```
+-----+
|                XCON-URI: 711331@example.com                |
+-----+
| Chat-ID=8asjdhk | xcon-userid: 839ULjj | Label=iede3 |
| Chat-ID=38iuhds | xcon-userid: 0283hHu  | Label=8heus  |
| Chat-ID=838unaH | xcon-userid: 0283hHu  | Label=3cnu7  |
| Chat-ID=djiowid | xcon-userid: ncH37Hs  | Label=jd38J  |
| Chat-ID=389hewu | xcon-userid: pakdj7H  | Label=U83hd  |
| Chat-ID=Ko03jdk | xcon-userid: pakdj7H  | Label=ehy3h  |
+-----+
```

Figure 3: Advanced Session Association + Media Label

In Figure 3, conference user identifiers '0283hHu' and 'pakdj7H' appear twice. The combination of multiple conference user identifiers and a unique chat-ID enables the conference system to clearly identify a specific Group Chat instance. 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. Advanced Operations

Advanced chat features, such as sidebars and private messages can also be supported within the context of the centralized conferencing framework using CCMP. Additional protocol details for these advanced features are provided in [\[RFC6504\]](#).

5. Additional Operations

This section discusses additional operations or features required to provide chat room functionality. Most of the operations are not explicitly defined in the centralized conferencing framework. While

most of the features and operations are achievable using the XCON data model [[RFC6501](#)] and data maintained by a conferencing system per the XCON framework, some advanced features require extensions to the XCON data model and may be optimized with more discrete CCMP messages.

5.1. Nicknames

Nicknames allow a user to define a text string that uniquely identifies the user within a particular chatroom without necessarily reflecting any protocol specific identity (e.g., SIP URI, Conference User Identifier, etc.). It is also important to note that the functionality to provide nicknames is not limited to users involved in chatrooms, thus it should be a general feature of the conferencing system.

Within a conferencing system, all nicknames MUST map to a conference user identifier. The nicknames are unique only to the specific conferencing system. To ensure uniqueness of nicknames, any new 'nickname' created MUST be compared with nicknames already in use or reserved following the rules defined in Preparation and Comparison of Nicknames [[I-D.ietf-precis-nickname](#)].

There may be multiple nicknames associated with a single conference user identifier (e.g., a user that has different nicknames for different chat rooms and/or voice/video conferences). In order to support nicknames, a 'nickname' attribute is defined in the XCON data model within the <user> element. A 'nickname' can be assigned to the conference user when an XCON-USERID is assigned to the user. The conferencing client MAY include a preferred nickname in the CCMP userRequest with a "create" operation.

The conferencing system allocates a conference user identifier and a nickname using system specific mechanisms, which can also include authentication. The conferencing system MUST associate the assigned nickname with the specific conference user identifier that has been allocated by updating the conference information. Another option would be to define a new CCMP message to just manipulate the 'nickname' element, but that is not necessary.

As described [Section 3.2](#), the XCON-userid identifier is used in conjunction with a chat-ID to internally represent a participant in a conference instance. This association is created when a conferencing client requests to create or join a specific chatroom. The nickname allocated for the specific conferencing user identifier MUST also be associated with the chat session ID. Figure 4 provides an example of the association between the chat session identifier, the conference user identifier and conference nickname for a specific Group Chat

represented by the conference identifier.

+-----+ XCON-URI: 711331@example.com +-----+			
Chat-ID=8asjdhk	xcon-userid: 839ULjj	nickname=Alice	
Chat-ID=38iuhds	xcon-userid: 0283hHu	nickname=Bob	
Chat-ID=838unaH	xcon-userid: 0283hHu	nickname=CliffyBob	
Chat-ID=djiowid	xcon-userid: ncH37Hs	nickname=Dude	
Chat-ID=389hewu	xcon-userid: pakdjjH	nickname=Elliot	
Chat-ID=Ko03jdk	xcon-userid: pakdj7H	nickname=Fluffy	
+-----+			

Figure 4: Nickname Associations for a Group Chat

Depending upon the conferencing system, the conference system either allocates the preferred nickname for that user or allocates a different nickname. The nickname **MUST** be included in the CCMP userResponse message.

In the future, if a more generic nickname mechanism is available, rather than provide nicknames that are specific to the conferencing system, a conferencing system may interface with a nickname registry, for example, in order to allocate a new nickname for a specific conferencing client. This change in how a conferencing system allocates nicknames should not impact the CCMP protocol interface to support nicknames.

5.2. Logging

A common chat feature involves logging the history of a chat room. This provides a record of a chat room that can be used when a user first joins a chat room as discussed in [Section 5.3](#). It can also be used to provide a complete capture of a specific chat room session. When a participant enters a room in which the discussions are logged, the conferencing system **MUST** warn the participant that the discussions are logged.

The centralized conferencing framework does not fully describe the role of recording or logging of active conferences. However, this functionality can be realized with the manipulation of the appropriate elements in the data model using the general conference control protocol operations. One approach for implementing this function would be to have it be based on specific manipulation of the

conference by a user with the appropriate permissions (i.e., confRequest message with an "update" operation to start and stop recording). Another mechanism for implementing this function would be to have a specific user as part of the conference to perform this function, and having the media proxied to a logging device. In the case of systems that support the Media Control architecture [RFC5567] and SIP Control Framework [RFC6230] along with the specific Mixer control package [RFC6505], the addition of a user to proxy the media for recording is described in section 6.2.3 in [I-D.ietf-mediactrl-call-flows]

5.3. History

A common chat feature allows users to view the past history of chat rooms. This operation is common when a user first joins a chat room that is underway. A user is often offered the option to review a specific number of past messages.

Conferencing systems that maintain the history associated with specific chat rooms through logging, as described in [Section 5.2](#), should provide a mechanism, using the conference identifier, to access the specific information requested by a user based on a specific timestamp. The user request for the information and the rendering of the information is specific to the user's session based messaging protocol and may not be supported by all the messaging protocols.

5.4. Indicating Alternate Venue

Another chat room feature provides the details of an alternate chat room venue for previously active chat rooms that have been closed, with a related topic. While not detailed in the centralized conferencing framework, this functionality can be accomplished by creating the new chat room as a child or sibling of the previous chat room and providing the Active chat conference object identifier to any valid users that attempt to join a previous chat room. The information about the new chat room can also be provided at the end of a chat room that is being de-activated at the end of the session.

5.5. File Transfer

The ability to send files to a selected participant or group of participants is another common functionality, supported by messaging protocols. This functionality also enables the exchange of information (e.g. name, size, and date) about the file to be transferred and usually provides a mechanism to show an image thumbnail for files such as photos. This capability could be reflected in the conference data (in the conference instance) and

requires at least an extension to the "available-media" element. The thumbnail rendering of the image is outside the scope of the data model and is specific to the client application. Additional functionality to support this capability requires further study.

5.6. Real Time Collaboration

The messaging protocols can be used, and are being used, in applications quite different from a simple exchange of text messages between two participants in the context of a chatroom. Real-time collaboration tools (e.g. Whiteboarding, screen-share, co-browse or document-share) are some of these applications.

The Conferencing Systems are usually bound to Real-time collaboration tools to increase the productivity of distributed teams. In terms of correlating this functionality with CCMP, the mechanisms for manipulating the conference are the same in terms of updating the data associated with the conference with the additional attributes to reflect the multiple sources of media for the chatroom. This capability could be reflected in the conference data (in the conference instance) with an extension to the "available-media" element. Some current systems using SIP embed the attributes in the media stream. Overall, supporting this functionality requires further study, in particular with regards to the RTCWeb initiative as described in documents such as [\[I-D.ietf-rtcweb-use-cases-and-requirements\]](#)

6. Security Considerations

As discussed in the Centralized Conferencing 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 associated with chatrooms includes 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 (i.e., MSRP/SIP, CCMP, XMPP, etc.), it depends on the security solutions for those protocols and the associated authorization mechanisms. This solution is based on the Centralized Conferencing framework and 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 Centralized Conferencing Framework.

7. IANA Considerations

This document requires no IANA registrations.

8. Acknowledgments

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

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC5239] Barnes, M., Boulton, C., and O. Levin, "A Framework for Centralized Conferencing", [RFC 5239](#), June 2008.
- [RFC6501] Novo, O., Camarillo, G., Morgan, D., and J. Urpalainen, "Conference Information Data Model for Centralized Conferencing (XCON)", [RFC 6501](#), March 2012.
- [RFC6503] Barnes, M., Boulton, C., Romano, S., and H. Schulzrinne, "Centralized Conferencing Manipulation Protocol", [RFC 6503](#), March 2012.
- [I-D.ietf-precis-nickname] Saint-Andre, P., "Preparation and Comparison of Nicknames", [draft-ietf-precis-nickname-06](#) (work in progress), July 2013.

9.2. Informative References

- [RFC6504] Barnes, M., Miniero, L., Presta, R., and SP. Romano, "Centralized Conferencing Manipulation Protocol (CCMP) Call Flow Examples", [RFC 6504](#), March 2012.
- [RFC1459] Oikarinen, J. and D. Reed, "Internet Relay Chat Protocol", [RFC 1459](#), May 1993.
- [RFC2810] Kalt, C., "Internet Relay Chat: Architecture", [RFC 2810](#), April 2000.

- [RFC2811] Kalt, C., "Internet Relay Chat: Channel Management", [RFC 2811](#), April 2000.
- [RFC2812] Kalt, C., "Internet Relay Chat: Client Protocol", [RFC 2812](#), April 2000.
- [RFC2813] Kalt, C., "Internet Relay Chat: Server Protocol", [RFC 2813](#), April 2000.
- [RFC6120] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core", [RFC 6120](#), March 2011.
- [RFC6121] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence", [RFC 6121](#), March 2011.
- [RFC4353] Rosenberg, J., "A Framework for Conferencing with the Session Initiation Protocol (SIP)", [RFC 4353](#), February 2006.
- [RFC4975] Campbell, B., Mahy, R., and C. Jennings, "The Message Session Relay Protocol (MSRP)", [RFC 4975](#), September 2007.
- [I-D.ietf-simple-chat]
Niemi, A., Garcia, M., and G. Sandbakken, "Multi-party Chat Using the Message Session Relay Protocol (MSRP)", [draft-ietf-simple-chat-18](#) (work in progress), January 2013.
- [RFC4574] Levin, O. and G. Camarillo, "The Session Description Protocol (SDP) Label Attribute", [RFC 4574](#), August 2006.
- [RFC5567] Melanchuk, T., "An Architectural Framework for Media Server Control", [RFC 5567](#), June 2009.
- [RFC6505] McGlashan, S., Melanchuk, T., and C. Boulton, "A Mixer Control Package for the Media Control Channel Framework", [RFC 6505](#), March 2012.
- [RFC6230] Boulton, C., Melanchuk, T., and S. McGlashan, "Media Control Channel Framework", [RFC 6230](#), May 2011.
- [I-D.ietf-mediactrl-call-flows]
Amirante, A., Castaldi, T., Miniero, L., and S. Romano, "Media Control Channel Framework (CFW) Call Flow Examples", [draft-ietf-mediactrl-call-flows-13](#) (work in progress), May 2013.

[I-D.ietf-rtcweb-use-cases-and-requirements]

Holmberg, C., Hakansson, S., and G. Eriksson, "Web Real-Time Communication Use-cases and Requirements", [draft-ietf-rtcweb-use-cases-and-requirements-11](#) (work in progress), June 2013.

[XEP-0045]

Saint-Andre, P., "Multi-User Chat", XSF XEP 0045, July 2008.

Authors' Addresses

Mary Barnes
Polycom

Email: mary.ietf.barnes@gmail.com

Chris Boulton
NS-Technologies

Email: chris@ns-technologies.com

Salvatore Loreto
Ericsson
Hirsalantie 11
Jorvas 02420, Finland

Email: salvatore.loreto@ericsson.com

