

SIPPING  
Internet-Draft  
Expires: April 26, 2004

J. Rosenberg  
dynamicsoft  
H. Schulzrinne  
Columbia University  
O. Levin, Ed.  
Microsoft Corporation  
October 27, 2003

**A Session Initiation Protocol (SIP) Event Package for Conference  
State  
draft-ietf-sipping-conference-package-02**

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

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 April 26, 2004.

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.

Abstract

This document defines a conference event package for the Session Initiation Protocol (SIP) Events framework, along with a data format used in notifications for this package. The conference package allows users to subscribe to a conference URI. Notifications are sent about changes in the membership of this conference, the status of users' participation in the conference, and the sidebars in the conference.



## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">3</a>
<a href="#">2.</a>	Terminology . . . . .	<a href="#">4</a>
<a href="#">3.</a>	Conference Event Package . . . . .	<a href="#">5</a>
<a href="#">3.1</a>	Event Package Name . . . . .	<a href="#">5</a>
<a href="#">3.2</a>	SUBSCRIBE Bodies . . . . .	<a href="#">5</a>
<a href="#">3.3</a>	Subscription Duration . . . . .	<a href="#">5</a>
<a href="#">3.4</a>	NOTIFY Bodies . . . . .	<a href="#">5</a>
<a href="#">3.5</a>	Notifier Processing of SUBSCRIBE Requests . . . . .	<a href="#">6</a>
<a href="#">3.6</a>	Notifier Generation of NOTIFY Requests . . . . .	<a href="#">6</a>
<a href="#">3.7</a>	Subscriber Processing of NOTIFY Requests . . . . .	<a href="#">6</a>
<a href="#">3.8</a>	Handling of Forked Requests . . . . .	<a href="#">7</a>
<a href="#">3.9</a>	Rate of Notifications . . . . .	<a href="#">7</a>
<a href="#">3.10</a>	State Agents . . . . .	<a href="#">7</a>
<a href="#">4.</a>	Conference Data Format . . . . .	<a href="#">8</a>
<a href="#">4.1</a>	Conference Information . . . . .	<a href="#">8</a>
<a href="#">4.1.1</a>	User Element . . . . .	<a href="#">8</a>
<a href="#">4.1.1.1</a>	User Statuses . . . . .	<a href="#">9</a>
<a href="#">4.1.1.2</a>	Media Stream Information . . . . .	<a href="#">9</a>
<a href="#">4.1.2</a>	Sidebar . . . . .	<a href="#">10</a>
<a href="#">4.2</a>	Constructing Coherent State . . . . .	<a href="#">10</a>
<a href="#">4.3</a>	Schema . . . . .	<a href="#">11</a>
<a href="#">4.4</a>	Example . . . . .	<a href="#">13</a>
<a href="#">5.</a>	Security Considerations . . . . .	<a href="#">15</a>
<a href="#">6.</a>	IANA Considerations . . . . .	<a href="#">16</a>
<a href="#">6.1</a>	conference Event Package Registration . . . . .	<a href="#">16</a>
<a href="#">6.2</a>	application/conference-info+xml MIME Registration . . . . .	<a href="#">16</a>
<a href="#">6.3</a>	URN Sub-Namespace Registration for urn:ietf:params:xml:ns:conference-info . . . . .	<a href="#">17</a>
<a href="#">6.4</a>	XML Schema Registration . . . . .	<a href="#">18</a>
<a href="#">7.</a>	Acknowledgements . . . . .	<a href="#">19</a>
<a href="#">8.</a>	Changes since -01 . . . . .	<a href="#">20</a>
	Normative References . . . . .	<a href="#">21</a>
	Informative References . . . . .	<a href="#">22</a>
	Authors' Addresses . . . . .	<a href="#">22</a>
	Intellectual Property and Copyright Statements . . . . .	<a href="#">23</a>



## **1. Introduction**

The Session Initiation Protocol (SIP) [3] Events framework [2] defines general mechanisms for subscribing to, and receiving notifications of, events within SIP networks. It introduces the notion of a package, which is a specific "instantiation" of the events framework for a well-defined set of events. Here, we define an event package for SIP conferences. This package provides the conference notification service as outlined in the SIP conferencing framework [9]. As described there, subscriptions to a conference URI are routed to the focus that is handling the conference. It acts as the notifier, and provides clients with updates on conference state.

The information provided by this package is comprised of conference identifier, conference participants (optionally with their statuses and media types) and conference sidebars.



## **2. Terminology**

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC 2119](#) [[1](#)] and indicate requirement levels for compliant implementations.

### **3. Conference Event Package**

The conference event package allows a user to subscribe to a conference. In SIP, conferences are represented by URIs. These URIs route to a SIP user agent, called a focus, that is responsible for ensuring that all users in the conference can communicate with each other [9]. The focus has sufficient information about the state of the conference to inform subscribers about it.

This section provides the details for defining a SIP Events package, as specified by [2].

#### **3.1 Event Package Name**

The name of this event package is "conference". This package name is carried in the Event and Allow-Events header, as defined in [2].

#### **3.2 SUBSCRIBE Bodies**

A SUBSCRIBE for a dialog package MAY contain a body. This body defines a filter to apply to the subscription. Filter documents are not specified in this document, and at the time of writing, are expected to be the subject of future standardization activity.

A SUBSCRIBE for a dialog package MAY be sent without a body. This implies the default subscription filtering policy. The default policy is:

- o Notifications are generated every time there is any change in the state of the conference.
- o Notifications do not normally contain full state; rather, they only indicate the state that has changed. The exception is a NOTIFY sent in response to a SUBSCRIBE. These NOTIFYS contain the full state of the information requested by the subscriber.

#### **3.3 Subscription Duration**

The default expiration time for a subscription to a conference is one hour. Once the conference ends, all subscriptions to that particular conference are terminated, with a reason of "noresource" [2].

#### **3.4 NOTIFY Bodies**

As described in RFC 3265 [2], the NOTIFY message will contain bodies that describe the state of the subscribed resource. This body is in a format listed in the Accept header field of the SUBSCRIBE, or a





package-specific default if the Accept header field was omitted from the SUBSCRIBE.

In this event package, the body of the notification contains a conference information document. This document describes the state of a conference. All subscribers and notifiers **MUST** support the "application/conference-info+xml" data format described in [Section 4](#). The subscribe request **MAY** contain an Accept header field. If no such header field is present, it has a default value of "application/conference-info+xml". If the header field is present, it **MUST** include "application/conference-info+xml", and **MAY** include any other types capable of representing dialog state.

Of course, the notifications generated by the server **MUST** be in one of the formats specified in the Accept header field in the SUBSCRIBE request.

### [3.5](#) Notifier Processing of SUBSCRIBE Requests

The conference information contains very sensitive information. Therefore, all subscriptions **SHOULD** be authenticated and then authorized before approval. Authorization policy is at the discretion of the administrator, as always. However, a few recommendations can be made.

It is **RECOMMENDED** that all users in the conference be allowed to subscribe to the conference.

### [3.6](#) Notifier Generation of NOTIFY Requests

Notifications **SHOULD** be generated for the conference whenever there is a change in the state in any of the information delivered to the subscriber.

The changes generally occur when a new participant joins, a participant leaves, or a participant is put on-hold. Subject to a local focus policy, changes in media types and other optional media attributes **MAY** be reported by the focus. In addition, creation and deletion of sidebars together with their rosters **MAY** be reported by the focus, subject to its local policy.

### [3.7](#) Subscriber Processing of NOTIFY Requests

The SIP Events framework expects packages to specify how a subscriber processes NOTIFY requests in any package specific ways, and in particular, how it uses the NOTIFY requests to construct a coherent view of the state of the subscribed resource.



Typically, the NOTIFY for the conference package will only contain information about those users whose state in the conference has changed. To construct a coherent view of the total state of all users, a subscriber to the dialog package will need to combine NOTIFYS received over time.

Notifications within this package can convey partial information; that is, they can indicate information about a subset of the state associated with the subscription. This means that an explicit algorithm needs to be defined in order to construct coherent and consistent state. The details of this mechanism are specific to the particular document type. See [Section 4.2](#) for information on constructing coherent information from an application/conference-info+xml document.

### **[3.8](#) Handling of Forked Requests**

By their nature, the conferences supported by this package are centralized. Therefore, SUBSCRIBE requests for a conference should not generally fork. Users of this package MUST NOT install more than a single subscription as a result of a single SUBSCRIBE request.

### **[3.9](#) Rate of Notifications**

For reasons of congestion control, it is important that the rate of notifications not become excessive. As a result, it is RECOMMENDED that the server not generate notifications for a single subscriber at a rate faster than once every 5 seconds.

### **[3.10](#) State Agents**

Conference state is ideally maintained in the element in which the conference resides. Therefore, the elements that maintain the conference are the ones best suited to handle subscriptions to it. Therefore, the usage of state agents is NOT RECOMMENDED for this package.



## **4. Conference Data Format**

Conference information is an XML document that **MUST** be well-formed and **SHOULD** be valid. Dialog information documents **MUST** be based on XML 1.0 and **MUST** be encoded using UTF-8. This specification makes use of XML namespaces for identifying dialog information documents and document fragments. The namespace URI for elements defined by this specification is a URN [4], using the namespace identifier 'ietf' defined by [5] and extended by [6]. This URN is:

```
urn:ietf:params:xml:ns:conference-info
```

A conference information document begins with the root element tag "conference-info".

### **4.1 Conference Information**

Conference information begins with the top level element "conference-info". This element has three mandatory attributes:

version: This attribute allows the recipient of conference information documents to properly order them. Versions start at 0 and increment by one for each new document sent to a subscriber. Versions are scoped within a subscription. Versions **MUST** be represented using a 32 bit integer.

state: This attribute indicates whether the document contains the full conference information, or whether it contains only the information that has changed since the previous document (partial).

entity: This attribute contains the conference URI that identifies the conference being described in the document.

The "conference-info" element has zero or more "user" sub-elements which contain information on the users in the conference. This is followed by zero or more "sidebar" sub-elements which contain information on the sidebars in the conference.

#### **4.1.1 User Element**

The user element has one mandatory attribute, "uri" that indicates the URI for the user in the conference. This is a logical identifier, not a machine specific one (i.e., it's taken from the authenticated identity of the participant). The optional attribute "display-name" contains a display name for the user. The standard "xml:lang" language attribute can also be present to indicate the language of



the display-name.

#### **4.1.1.1 User Statuses**

Two optional status elements are defined: activity-status and history status. "Activity-status" provides information about user's current level of participation in the conference. "History-status", if present, provides information about the way the user joined or left the conference. Additional optional indication "is-on-dial-out-list" completes the information about the user membership in the conference.

The following statuses are defined for the activity-status element:

connected: The user is a participant in the conference. Depending on the media policies, he/she can send and receive media to and from other participants.

disconnected: The user is not a participant in the conference and no active dialog exists between the user and the focus.

on-hold: Active SIP dialog exists between a user and a focus, but user is "on-hold" for this conference. As an example, the user has asked to join the conference using SIP, but his/her participation is pending based on moderator approval. In the meantime he/she is hearing music-on-hold or some other kind of related content.

The following statuses are defined for the history-status element:

dialed-in: The user dialed into the conference, i.e. sent INVITE to the focus, which resulted in successful dialog establishment.

dialed-out: The focus has brought the user into the conference by sending a successful INVITE to the user.

departed: The user sent a BYE, thus leaving the conference.

booted: The user was sent a BYE by the focus, booting him/her out of the conference.

failed: The server tried to bring the user into the conference, but its attempt to contact the specific user resulted in a non-200 class final response.

#### **4.1.1.2 Media Stream Information**

Each user has zero or more "media-stream" sub-elements.





Each "media-stream" element indicates the media stream that the user is currently connected to. Here, "connected to" implies that a user has a media line in their SDP [10]. With this definition, a user is connected to a media stream even if they are not sending any media.

The "media-stream" element has a mandatory "media-type" attribute which identifies the media type (e.g. audio, video, message and application) and MUST have one of the values registered for "media" of SDP [10].

The "media-stream" element has also an optional "proto" sub-element, which MUST have the value registered for "proto" of SDP [10]).

An optional "ssrc" sub-element, if present, carries the value of SSRC (RTP/RTCP [8]) as generated by the user for the stream it sends.

When an RTP mixer generates a CSRC list according to RTP/RTCP [8], it inserts a list of the SSRC identifiers of the sources that contributed to the generation of a particular packet into the RTP header of that packet. "An example application is audio conferencing where a mixer indicates all the talkers whose speech was combined to produce the outgoing packet, allowing the receiver to indicate the current talker, even though all the audio packets contain the same SSRC identifier (that of the mixer)."

#### **4.1.2 Sidebar**

The sidebar element has one attribute - "entity" that indicates the URI which identifies the sidebar. A sidebar has zero or more users that are of type "user-type" as the users of the main conference are.

### **4.2 Constructing Coherent State**

The conference information subscriber maintains a table for the list of users in the conference. The table contains a row for each user. Each row is indexed by a URI, present in the "uri" attribute of the "user" element. The contents of each row contain the state of that user as conveyed in the document.

The table is associated with a version number. The version number MUST be initialized with the value of the "version" attribute from the "conference-info" element in the first document received. Each time a new document is received, the value of the local version number, and the "version" attribute in the new document, are compared. If the value in the new document is one higher than the local version number, the local version number is increased by one, and the document is processed. If the value in the document is more than one higher than the local version number, the local version



number is set to the value in the new document, the document is processed, and the subscriber SHOULD generate a refresh request to trigger a full state notification. If the value in the document is less than the local version, the document is discarded without processing.

The processing of the conference information document depends on whether it contains full or partial state. If it contains full state, indicated by the value of the "state" attribute in the "conference-info" element, the contents of the table is flushed. It is repopulated from the document. A new row in the user table is created for each "user" element. If the document contains partial state, as indicated by the value of the "state" attribute in the "conference-info" element, the document is used to update the table. For each "user" element in the document, the subscriber checks to see whether a row exists for that user in the user table. This check is done by comparing the URI in the "uri" attribute of the "user" element with the URI associated with the row. If the user doesn't exist in the table, a row is added, and its state is set to the information from that "user" element. If the user does exist, its state is updated to be the information from that "user" element. If a row is updated or created, such that its state is now disconnected, booted, failed or departed, that entry MAY be removed from the table at any time.

### [4.3](#) Schema

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-info"
xmlns:tns="urn:ietf:params:xml:ns:conference-info" xmlns:xs="http://www.w3.org/
2001/XMLSchema" xmlns="urn:ietf:params:xml:ns:conference-info"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <!--
    This import brings in the XML language attribute xml:lang
  -->
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
schemaLocation="http://www.w3.org/2001/03/xml.xsd" />

  <xs:element name="conference-info">

    <xs:complexType>
      <xs:sequence>
        <xs:element name="user" type="user-type" minOccurs="0"
maxOccurs="unbounded" />
        <xs:element name="sidebar" type="sidebar-type" minOccurs="0"
```

```
maxOccurs="unbounded" />  
  <xs:any processContents="lax" minOccurs="0" maxOccurs="unbounded" />  
</xs:sequence>
```

```
<xs:attribute name="version" type="xs:nonNegativeInteger" use="required" />
>
<xs:attribute name="state" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="full" />
      <xs:enumeration value="partial" />
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>

<xs:attribute name="entity" type="xs:anyURI" use="required" />

</xs:complexType>
</xs:element>

<xs:complexType name="user-type">
  <xs:sequence>
    <xs:element name="activity-status" type="tns:activity-status-type"
minOccurs="0" />
    <xs:element name="history-status" type="tns:history-status-type"
minOccurs="0" />
    <xs:element name="is-on-dial-out-list" type="xs:boolean" minOccurs="0" />
    <xs:element name="media-stream" type="tns:media-stream-type" minOccurs="0"
maxOccurs="unbounded" />
    <xs:any processContents="lax" minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>

  <xs:attribute name="uri" type="xs:anyURI" use="required" />
  <xs:attribute name="display-name" type="xs:string" use="optional" />
  <xs:attribute ref="xml:lang" use="optional" />

</xs:complexType>

<xs:complexType name="sidebar-type">
  <xs:sequence>
    <xs:element name="user" type="user-type" minOccurs="0"
maxOccurs="unbounded" />
    <xs:any processContents="lax" minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>

  <xs:attribute name="entity" type="xs:anyURI" use="required" />
</xs:complexType>

<xs:complexType name="media-stream-type">
  <xs:sequence>
    <xs:element name="proto" type="xs:string" minOccurs="0" />
```

```
<xs:element name="ssrc" type="xs:nonNegativeInteger" minOccurs="0" />  
<xs:any processContents="lax" minOccurs="0" maxOccurs="unbounded" />  
</xs:sequence>
```

```
<xs:attribute name="media" type="xs:string" use="required" />
</xs:complexType>

<xs:simpleType name="activity-status-type">
  <xs:restriction base="xs:string">
    <xs:enumeration value="connected" />
    <xs:enumeration value="disconnected" />
    <xs:enumeration value="on-hold" />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="history-status-type">
  <xs:restriction base="xs:string">
    <xs:enumeration value="dialed-in" />
    <xs:enumeration value="dialed-out" />
    <xs:enumeration value="departed" />
    <xs:enumeration value="booted" />
    <xs:enumeration value="failed" />
  </xs:restriction>
</xs:simpleType>

</xs:schema>
```

#### [4.4](#) Example

The following is an example conference information document:





```
<conference-info version="0" state="full" entity="sip:conf233@example.com">
  <user uri="sip:bob@example.com" display-name="Bob Jones">
    <activity-status>connected</activity-status>
    <history-status>dialed-in</history-status>
    <media-stream media-type="audio"
      <proto> RTP/AVP </proto>
      <ssrc> 583398 <ssrc>
    </media-stream>
  </user>

  <user uri="sip:barbara@example.com" display-name="Barbara Jones">
    <activity-status>on-hold</activity-status>
  </user>

  <user uri="sip:bill@example.com" display-name="Bill Minelli">
    <activity-status>on-hold</activity-status>
  </user>

  <sidebar entity="sip:conf233.1@example.com">
    <user>uri="sip:barbara@example.com"</user>
    <user>uri="sip:bill@example.com"</user>
  </sidebar>

</conference-info>
```

This document describes a conference with three users, two of which are in a sidebar conversation.



## **5. Security Considerations**

Subscriptions to conference state can reveal very sensitive information. For this reason, the document recommends authentication and authorization, and provides guidelines on sensible authorization policies.

Since the data in notifications is sensitive as well, end-to-end SIP encryption mechanisms using S/MIME SHOULD be used to protect it.

## **6. IANA Considerations**

This document registers a SIP event package, a new MIME type, application/conference-info+xml, a new XML namespace, and a new XML schema.

### **6.1 conference Event Package Registration**

This specification registers an event package, based on the registration procedures defined in [RFC 3265](#) [2]. The following is the information required for such a registration:

Package Name: conference

Package or Template-Package: This is a package.

Published Document: RFC XXXX (Note to RFC Editor: Please fill in XXXX with the RFC number of this specification).

Person to Contact: Jonathan Rosenberg, jdrosen@jdrosen.net.

### **6.2 application/conference-info+xml MIME Registration**

MIME media type name: application

MIME subtype name: conference-info+xml

Mandatory parameters: none

Optional parameters: Same as charset parameter application/xml as specified in [RFC 3023](#) [7].

Encoding considerations: Same as encoding considerations of application/xml as specified in [RFC 3023](#) [7].

Security considerations: See [Section 10 of RFC 3023](#) [7] and [Section 5](#) of this specification.

Interoperability considerations: none.

Published specification: This document.

Applications which use this media type: This document type has been used to support SIP conferencing applications.



## Additional Information:

Magic Number: None

File Extension: .cif or .xml

Macintosh file type code: "TEXT"

Personal and email address for further information: Jonathan Rosenberg, <jdrosen@jdrosen.net>

Intended usage: COMMON

Author/Change controller: The IETF.

### **6.3 URN Sub-Namespace Registration for urn:ietf:params:xml:ns:conference-info**

This section registers a new XML namespace, as per the guidelines in [6].

URI: The URI for this namespace is  
urn:ietf:params:xml:ns:conference-info.

Registrant Contact: IETF, SIPING working group, <sipping@ietf.org>, Jonathan Rosenberg <jdrosen@jdrosen.net>.

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
    "http://www.w3.org/TR/xhtml1-basic/xhtml1-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=iso-8859-1"/>
  <title>Conference Information Namespace</title>
</head>
<body>
  <h1>Namespace for Conference Information</h1>
  <h2>urn:ietf:params:xml:ns:conference-info</h2>
  <p>See <a href="[[[URL of published RFC]]]">RFCXXXX</a>.</p>
</body>
</html>
END
```





#### **[6.4](#) XML Schema Registration**

This specification registers a schema, as per the guidelines in in [\[6\]](#).

URI: please assign.

Registrant Contact: IETF, SIPING Working Group  
(sipping@ietf.org), Jonathan Rosenberg (jdrosen@jdrosen.net).

XML: The XML can be found as the sole content of [Section 4.3](#).

## **7. Acknowledgements**

The authors would like to thank Dan Petrie and Sean Olson for their comments.

## **8. Changes since -01**

- o Package parameters are removed. Decision about performing "recursive" membership algorithm is perceived as a focus local policy.
- o General information (i.e. pointers to additional available services) is removed. The defined XML schema can be extended in future to include those when XCON work matures.
- o Dialog information is removed. It can be obtained by direct subscription to a dialog package of a participant.
- o Media stream information is aligned with SDP definitions (media and proto) and SSRC attribute is added.
- o Participant's status is expressed using two optional statuses: "activity" and "history". Optional "is-on-a-dial-out-list" indication is added.
- o Normative references to XCON work are removed.
- o Optional sidebar rosters are added.



## Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [2] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", [RFC 3265](#), June 2002.
- [3] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and E. Schooler, "SIP: Session Initiation Protocol", [RFC 3261](#), June 2002.
- [4] Moats, R., "URN Syntax", [RFC 2141](#), May 1997.
- [5] Moats, R., "A URN Namespace for IETF Documents", [RFC 2648](#), August 1999.
- [6] Mealling, M., "The IETF XML Registry", [draft-mealling-iana-xmlns-registry-05](#) (work in progress), June 2003.
- [7] Murata, M., St. Laurent, S. and D. Kohn, "XML Media Types", [RFC 3023](#), January 2001.
- [8] Schulzrinne, H., Casner, S., Frederick, R. and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", [RFC 3550](#), July 2003.



## Informative References

- [9] Rosenberg, J., "A Framework for Conferencing with the Session Initiation Protocol", [draft-ietf-sipping-conferencing-framework-00](#) (work in progress), May 2003.
- [10] Handley, M. and V. Jacobson, "SDP: Session Description Protocol", [RFC 2327](#), April 1998.

## Authors' Addresses

Jonathan Rosenberg  
dynamicsoft  
600 Lanidex Plaza  
Parsippany, NJ 07054  
US

Phone: +1 973 952-5000  
EMail: [jdrosen@dynamicsoft.com](mailto:jdrosen@dynamicsoft.com)  
URI: <http://www.jdrosen.net>

Henning Schulzrinne  
Columbia University  
M/S 0401  
1214 Amsterdam Ave.  
New York, NY 10027  
US

EMail: [schulzrinne@cs.columbia.edu](mailto:schulzrinne@cs.columbia.edu)  
URI: <http://www.cs.columbia.edu/~hgs>

Orit Levin (editor)  
Microsoft Corporation  
One Microsoft Way  
Redmond, WA 98052  
USA

EMail: [oritl@microsoft.com](mailto:oritl@microsoft.com)





## Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in [BCP-11](#). Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

## Full Copyright Statement

Copyright (C) The Internet Society (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assignees.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION



HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF  
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

#### Acknowledgment

Funding for the RFC Editor function is currently provided by the  
Internet Society.