Internet Engineering Task Force
INTERNET-DRAFT
draft-singh-sip-h323.ps

Singh/Schulzrinne
Columbia University

January 10, 2000
Expires: July 2000

Interworking Between SIP/SDP and H.323

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.”

To view the list Internet-Draft Shadow Directories, see http://www.ietf.org/shadow.html.

Copyright Notice

Abstract

This document describes the interworking between SIP and H.323, including the translation between H.245 and SDP. We list general requirements for such a translation and a solution which meets those requirements. We describe the call setup via message flows and pseudo code.

Contents

1 Introduction

It appears likely that both SIP [1], with SDP [2], and H.323 [3] will be used for internet multimedia signaling in the next few years. Both these protocols run over IP (Internet Protocol) and use RTP (Real time Transport protocol [4]) for transferring realtime audio/video data, reducing the task of interworking between these protocols to merely translating the signaling protocols and session descriptions.

We enumerate the requirements for a translation between H.323 and SIP/SDP and then propose a solution which meets those requirements. Issues related to a new enhanced version of SDP (Session Description Protocol [2]) is kept open while discussing the solution, so in future any change in SDP can be easily included in this document.

Section 2 describes the scope of this document. Section 3 lists the terminology used in the document. Section 4 gives the requirements for a simple translation between SIP/SDP and H.323. Section 5 describes simple call scenarios for call setup and address resolution. In section 6 we have described a mapping between H.323 and SIP addresses. Section 7 describes an algorithm to find a common subset of H.323 and SIP capabilities. Section 8 lists the protocol level requirements for the gateway. Pseudo-code for a simple translation is given in Appendix A.

2 Scope of This Document

This document describes interworking between H.323 Version 2.0 and SIP Version 2.0. However, since an H.323v2 terminal may or may not support FastConnect, solutions without using this feature are also detailed. Only a simple call scenario is presented. It does not cover conferencing or advanced call services like call forwarding, call transfer. This document also describes the translation between H.323 and SDP for session description.

Overlap sending of dialed digits is not supported. DataApplication (T.120), encryption, security and authentication are not covered in this document.

3 Terminology and Conventions

Gateway (GW): The H.323-SIP signaling gateway described in this document.

Endpoint: H.323 endpoint or SIP user agent.

Signaling: Generic name for protocols specified by Q.931 [5], H.245 [6] or SIP [1].

Data traffic: RTP/RTCP encapsulated data (multimedia) traffic.

Gatekeeper (GK): H.323 gatekeeper which can accept RRQ (registration request) and ARQ (admission request) messages belonging to the RAS protocol.

Registrar: SIP server which accepts REGISTER requests.

Cloud: Logical collection of entities using the same signaling protocol. In this document, we refer to the H.323 and SIP clouds. Note that we assume that both of these clouds use IP as their underlying network layer.

The H.323 [3] and SIP [1] specifications provide additional terms used here.

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 [7].

In message flow sequences, we label message flows as follows:

```
=========>  SIP message
~~~~~~~~~>  RAS message
--------->  Q.931 message
--+--+--->  H.245 message
```

4 Translation Requirements

Basic requirements for any SIP-H.323 signaling gateway are summarized below:

1. Protocol compliance:
   The gateway should use the components of H.323 and SIP. The gateway should handle all mandatory features of H.323 as well as SIP. Common call scenarios should be simple to implement.

2. User registration:
   The gateway should use the user registration in both the H.323 and SIP clouds to resolve the user name (alias or URL) to an IP address. In other words, it should provide a framework in which the user may dial any address without actually knowing whether it belongs to the H.323 or the SIP cloud.

3. Mapping between H.245 and SDP:

   The gateway should be able to map all the mandatory H.245 messages to apporopriate SDP messages and vice-versa, without the endpoint being aware that such conversion is taking place. Other optional features of H.245 and SDP should be mapped as much as possible to facilitate maximum interworking between the two clouds.

4. Direct RTP/RTCP traffic between the endpoints:
   Where possible, the gateway should route RTP/RTCP traffic directly between the endpoints involved in the conference without going through the gateway. This reduces the delay for media packets and helps building scaleable gateways.

5. Transparent support for audio/video algorithms:
   The gateway should provide transparent support for audio/video algorithms, i.e., the gateway should not restrict the capabilities of the endpoints in terms of audio/video algorithms supported.

6. Call sequence mapping:
   The gateway should map the message sequence between H.323 and SIP in such a way that every important decision (accept or reject a call, choose an algorithm for a logical channel, and so on) is taken by the endpoints involved in the conference and not by the gateway itself.

We assume throughout most of this document that the session description given by a SIP endpoint refers to both the transmit and the receive capabilities of the SIP endpoint. This may not be true in a particular application. If that is the case then the SIP endpoint is expected to give that information in SDP using recvonly or sendonly media attributes.

The analysis of SIP-H.323 interworking can be split into

- simple call setup;
- mapping addresses;
- finding a subset of capabilities described by H.245 and SDP;
- conferencing and call services;
- security and authentication.

The last two issues are not addressed in this document. Section 5 describes call setup and teardown; while Section 6 describes address mapping and section 7 the capabilities calculation.

5 Call Scenario

A simple gateway architecture is shown in Fig. 1. Note that an H.323 gatekeeper and/or a SIP server may be part of the gateway. The H.323 cloud is shown on the left hand side and the SIP cloud on the right hand side.

The following subsections describe and evaluate different call scenarios.

5.1 User Registration and Address Resolution

5.1.1 Gateway Contains SIP Proxy Server and Registrar

Fig. 2 shows a gateway that contains a SIP proxy and registrar.

When receiving a SIP REGISTER request, the GW generates an H.323 RAS RRQ request to its local GKs. The callSignalAddress of the RAS message contains the network address of the GW; the terminalType is set to "gateway" and the terminalAlias is derived from the SIP To SIP-Address, as described in Section 6.

Thus, any address resolution request coming from the H.323 cloud to a SIP address can be resolved by H.323 gatekeeper(s) using H.323 RAS requests. Any request coming from the SIP cloud to H.323 is forwarded to the H.323 gatekeeper(s) by the gateway. H.323 gatekeeper(s) resolve this address using RAS/H.323.

During initialization, the gateway registers its own alias address (e.g., gw1) with its local H.323 gatekeepers, so that anybody from the H.323 cloud can reach SIP endpoints by directly connecting to the alias address of the gateway and by providing a SIP address in the remote extension address of the SETUP message of H.323.

Fig. 3 shows the message flow sequences for successful initialization.

Address resolution from SIP to H.323 is shown in Fig. 4, while address resolution from H.323 to SIP is shown in Fig. 5.

This scheme assumes that the gateway is aware of the client part of the H.323 RAS protocol so that it can talk to the gatekeeper. Each SIP UA that registers with the registrar also appears in the gatekeeper's database.

```
                         Gateway
 +---------+         +------+------+      +-------+
 |  H.323  |---------|      |      |      | SIP   |
 | endpoint|         | H.323| SIP  |======| user  |
 +---------+         |Termi-| user |      | agent |
                     | nal  |agent |      +-------+
+----------+         |      |      |
| H.323    |---------|      |      |      +------------+
|gatekeeper|         |      |      |======| SIP Server |
+----------+         |      |      |      +------------+
                     +------+------+
```

Figure 1: SIP-H.323 gateway architecture

```
 SIP REGISTER  |---------|-------------|  RAS   |-------------|
   ===========>| SIP     |   gateway   |~~~~~~~>| Gatekeeper  |
               |registrar|             |        |             |
               |---------|-------------|        |-------------|
```

Figure 2: Gateway colocated with SIP server

```
H.323         GK1            GK2            GW          SIP UA
Terminal                                    gw1
|              |              |              |             |
|              |              |     RRQ      |             |
|              |              |<~~~~~~~~~~~~~|             |
|              |              |    (gw1)     |             |
|              |              |              |             |
|              |              |     RCF      |             |
|              |              |~~~~~~~~~~~~~>|             |
|     RRQ      |              |              |             |
|~~~~~~~~~~~~~>|              |              |             |
| (kns10@columbia.edu)        |              |             |
|              |              |              |             |
|     RCF      |              |              |             |
|<~~~~~~~~~~~~~|              |              |             |
|              |              |              |  REGISTER   |
|              |              |              |<============|
|              |              |              | To: hgs@cs.columbia.edu
|              |              |     RRQ      |             |
|              |              |<~~~~~~~~~~~~~|             |
|              |              |  (hgs@cs.columbia.edu)     |
|              |              |              |             |
|              |              |     RCF      |             |
|              |              |~~~~~~~~~~~~~>|             |
|              |              |              |   200 OK    |
|              |              |              |============>|
|              |              |              |             |
```

Figure 3: Gateway initialization, as described in Section 5.1.1

```
H.323 Terminal GK1            GK2            GW          SIP UA
128.59.16.1                                        columbia.edu
The H.323 Terminal is initialized:

|              |              |              |             |
|     RRQ      |              |              |             |
|~~~~~~~~~~~~~>|              |              |             |
| kns10@columbia.edu          |              |             |
|              |              |              |             |
|     RCF      |              |              |             |
|<~~~~~~~~~~~~~|              |              |             |

Call:

|              |              |              |   INVITE    |
|              |              |              |<============|
|              |              |              | To: kns10@columbia.edu
|              |              |              |             |
|              |              |              | 100 Trying  |
|              |              |              |============>|
|              |              |     ARQ      |             |
|              |              |<~~~~~~~~~~~~~|             |
|              |              |  kns10@columbia.edu        |
|              |     LRQ      |              |             |
|              |<~~~~~~~~~~~~~|              |             |
|              |  kns10@columbia.edu         |             |
|              |              |              |             |
|              |     LCF      |              |             |
|              |~~~~~~~~~~~~~>|              |             |
|              | 128.59.16.1  |              |             |
|              |              |              |             |
|              |              |     ACF      |             |
|              |              |~~~~~~~~~~~~~>|             |
|              |              | 128.59.16.1  |             |
```

Figure 4: Address translation from SIP to H.323, as described in Section 5.1.1

```
SIP UA        GW             GK2            GK1     H.323 Terminal
         cs.columbia.edu
           128.59.16.2
|              |              |              |             |
|  REGISTER    |              |              |             |
|=============>|              |              |             |
| To: hgs@cs.columbia.edu     |              |             |
|              |              |              |             |
|              |     RRQ      |              |             |
|              |~~~~~~~~~~~~~>|              |             |
|              | hgs@cs.columbia.edu         |             |
|              | at 128.59.16.2              |             |
|              |              |              |             |
|              |     RCF      |              |             |
|              |<~~~~~~~~~~~~~|              |             |
|     OK       |              |              |             |
|<=============|              |              |             |
|              |              |              |             |

The SIP UA has registered its address during initialization.

|              |              |              |     ARQ     |
|              |              |              |<~~~~~~~~~~~~|
|              |              |              | hgs@cs.columbia.edu
|              |              |              |             |
|              |              |     LRQ      |             |
|              |              |<~~~~~~~~~~~~~|             |
|              |              hgs@cs.columbia.edu          |
|              |              |              |             |
|              |              |     LCF      |             |
|              |              |~~~~~~~~~~~~~>|             |
|              |              |128.59.16.2   |             |
|              |              |              |             |
|              |              |              |     ACF     |
|              |              |              |~~~~~~~~~~~~>|
|              |              |              | 128.59.16.2 |
```

Figure 5: Address translation from H.323 to SIP, as described in Section 5.1.1

5.1.2 Gateway Contains an H.323 Gatekeeper

In an alternative architecture, shown in Fig. 6, the gateway contains an H.323 gatekeeper in addition to a SIP UA. Address resolution from SIP to H.323 is shown in Fig. 7. while address resolution from H.323 to SIP is shown in Fig. 8.

5.1.3 Gateway Does Not Contain Gatekeeper or Registrar

Instead of having the GW contain a GK or registrar, it may be preferable to have the GW resolve addresses when call setup requests arrive. Thus, the GK does not store any address mappings of H.323 or SIP endpoints. When a call arrives at the gateway from SIP cloud, the gateway sends a RAS ARQ request to the H.323 cloud. If the address cannot be resolved or if the RAS request times out, it sends an appropriate response to the SIP endpoint. Similarly, calls from the H.323 cloud are translated into SIP requests and sent to a proxy or end system.

This approach works well if calls are identified by URLs indicating the signaling scheme, i.e., if an H.323 request is directed to a SIP URL or vice versa. In that case, it is sufficient if the GK or proxy is pre-configured with the address of the GW.

If the destination address does not indicate the signaling protocol, a SIP proxy server has to forward all incoming requests to a local gateway, just in case the destination happens to be reachable via H.323.

In this architecture, the gateway MUST implement the RAS LRQ (location request) and LCF (location confirmation) messages. When a call is initiated by an H.323 entity, its gatekeeper will send an LRQ request to other gatekeepers at the well-known GK multicast address. The gateway captures the LRQ message and can use one of two approaches to find out if a SIP end point is available at that address. In the first approach, the GW sends a REGISTER request without Contact information to the domain identified in the request (see Section 6). If the registrar has information about the endpoint, it returns this information in the Contact headers of the response. The GW then translates this information and responds to the H.323 cloud with a LCF (location confirmation) message. If the registrar returns a negative indication, the gateway responds with a LRJ (location reject) message or remains silent. (Note that it is permitted that a terminal responds to LRQ messages, so that a gatekeeper is not needed as a part of the gateway application.) This approach is equivalent to SIP third-party registration and will not work if the registrar requires authentication. The second approach uses SIP OPTIONS messages, but is otherwise identical.

5.1.4 Direct Connection

If a gateway receives a Q.931 SETUP message, the gateway tries to parse the Q.931 destinationAddress. If the destinationAddress is not of the gateway itself and if it is able to resolve it to a SIP address, then the procedure described in section 5.2 is used to establish the call. (Note that the user registration steps are not involved in this scenario.) Otherwise, if the destination address is that of the gateway and a remote extension address is present in the SETUP message of Q.931, then the gateway should use the remote extension address to determine the SIP address. The gateway MAY also be configured to forward all requests to a pre-defined SIP proxy.

5.2 Call Establishment

A call requires three crucial pieces of information, namely the logical destination address, the media transport address and the media description.

```
   RRQ      +--------------------+ REGISTER +-------------+
~~~~~~~~~~~>|   H.323    |SIP UA |=========>| SIP         |
            | gatekeeper |       |          | registrar   |
            +--------------------+          +-------------+
                  GW
```

Figure 6: Gateway colocated with a gatekeeper

```
H.323 Terminal    GW/GK            SIP server            SIP UA
                   gw1             columbia.edu
|                   |                   |                  |
|        RRQ        |                   |                  |
|~~~~~~~~~~~~~~~~~~>|                   |                  |
| kns10@columbia.edu|                   |                  |
|                   |                   |                  |
|                   |     REGISTER      |                  |
|                   |==================>|                  |
|                   | To: kns10@columbia.edu               |
|                   | Contact: kns10@gw1                   |
|                   |                   |                  |
|                   |      200 OK       |                  |
|                   |<==================|                  |
|        RCF        |                   |                  |
|<~~~~~~~~~~~~~~~~~~|                   |                  |
|                   |                   |                  |

The H.323 terminal has registered its alias address.

|                   |                   |     INVITE       |
|                   |                   |<=================|
|                   |                   | To: kns10@columbia.edu
|                   |                   |                  |
|                   |                   |   302 Moved      |
|                   |                   |=================>|
|                   |                   | Contact: kns10@gw1
|                   |                   |                  |
|                   |        INVITE kns10@gw1              |
|                   |<=====================================|
|                   |        To: kns10@columbia.edu        |
|                   |                   |                  |
|                   |             100 Trying               |
|                   |=====================================>|
|                   |                   |                  |
```

Figure 7: Address translation from SIP to H.323 when GW contains an H.323 GK

```
SIP UA        SIP server              GW/GK      H.323 terminal
x.cs.columbia.edu                  128.59.16.1
|                   |                   |                  |
|     REGISTER      |                   |                  |
|==================>|                   |                  |
| To: hgs@cs.columbia.edu               |                  |
| Contact: hgs@x.cs.columbia.edu        |                  |
|                   |                   |                  |
|      200 OK       |                   |                  |
|<==================|                   |                  |
|                   |                   |                  |

SIP user agent has registered its address with the server.

|                   |                   |       ARQ        |
|                   |                   |<~~~~~~~~~~~~~~~~~|
|                   |                   | hgs@cs.columbia.edu
|                   |      OPTIONS      |                  |
|                   |<==================|                  |
|                   | To: hgs@cs.columbia.edu              |
| OPTIONS hgs@x.cs.columbia.edu         |                  |
|<==================|                   |                  |
| To: hgs@cs.columbia.edu               |                  |
|                   |                   |                  |
|      200 OK       |                   |                  |
|==================>|                   |                  |
|                   |      200 OK       |                  |
|                   |==================>|                  |
|                   |                   |       ACF        |
|                   |                   |~~~~~~~~~~~~~~~~~>|
|                   |                   | 128.59.16.1      |
|                   |                   |                  |
```

Figure 8: Address translation from H.323 to SIP when GW contains an H.323 GK

Logical Destination address (A): This is the SIP address in To header or the destination alias address in the Q.931 SETUP message.

Media Description (M): In SIP, M is the list of supported payload types as given by SDP media description (“m=”) line. In H.245, M is given by the Terminal Capability Set.

Media Transport Address (T): The media transport address indicates the IP address and port number at which RTP/RTCP packets can be received. This information is available in the “c=” and the “m=” lines of SDP and the Open Logical Channel message of H.245.

The difficulty in translating between SIP and H.323 arises because A, M, and T are all contained in the SIP INVITE message, while H.323 may spread this information among several messages.

5.2.1 Call Establishment with H.323v2 Fast Connect

With H.323v2 FastConnect, the protocol translation is simplified because there is a one-to-one mapping between H.323 and SIP call establishment messages. Both the H.323 SETUP message with FastConnect and the SIP INVITE request have all three components (A, M and T). Call scenarios are shown in Fig. 9 and 10.

5.2.2 Call Establishment without H.323v2 FastConnect

Since H.323v2 terminals do not have to support the FastConnect feature, it is likely that the gateway receives incoming calls from the H.323 cloud without Fast Connect PDUs.

When the call is initiated by a SIP UA all the call information (A, M and T) is present in the SIP INVITE message and can be used to format H.323 messages. But when the call in initiated by an H.323 terminal, A, M and T are present in different messages. In a H.323 call without FastConnect, A is found in the Q.931 SETUP message, the TerminalCapabilitySet of H.245/H.323 contains M and T is present in the H.245 OpenLogicalChannel message. There are different ways in which these can be combined to form a SIP INVITE message. Two possible approaches are discussed below (in section 5.2.3 and 5.2.4).

5.2.3 Call from H.323 cloud to SIP cloud with H.245 TerminalCapabilitySet (TCS) Mapped to SDP

A first approach has the gateway send a SIP INVITE request when it receives a Q.931 SETUP message. The SDP body of the INVITE request contains a default session sescription. The default session description MUST be either empty or contain media description (m=) lines indicating the minimal capabilities of any H.323 terminal handled by the gateway. Currently, these minimal capabilities include only PCMU audio. If the session description is not empty, the gateway has two choices:

1. The gateway controls an RTP translator that can forward RTP packets between two different IP addresses. The SDP “c=” line indicates the address of the translator, with the port indicated in the “m=” line.
2. The “c=” line indicates a zero address and the “m=” line a zero port.

When the GW receives a 200 (OK) response for the INVITE request from the SIP cloud, the GW transmits a Q.931 CONNECT message to the H.323 endpoint. The GW initiates the H.245 capability with the TCS (Terminal Capability Set) sent to the H.323 endpoint. On receipt of the TCS from the H.323 end

```
SIP UA                        GW                     H.323 terminal
128.59.19.194                                       128.59.21.152
|                             |                            |
|           INVITE            |                            |
|============================>|                            |
|  To: kns10@columbia.edu     |                            |
|  c=IN IP4 128.59.19.194     |                            |
|  m=audio 8000 RTP/AVP 0     |                            |
|                             |           SETUP            |
|                             |--------------------------->|
|                             | fastStart={g711Ulaw,Tx},   |
|                             |  {g711Ulaw,Rx,128.59.19.194:8000}
|                             |                            |
|                             |          Connect           |
|                             |<---------------------------|
|                             | fastStart=                 |
|                             | {g711Ulaw,Tx,128.59.21.152:10000},
|                             | {g711Ulaw,Rx}              |
|           200 OK            |                            |
|<============================|                            |
|  c=IN IP4 128.59.21.152     |                            |
|  m=audio 10000 RTP/AVP 0    |                            |
|                             |                            |
|             ACK             |                            |
|============================>|                            |
|                             |                            |
```

Figure 9: Call setup from SIP UA to H.323 terminal with FastConnect

```
H.323 terminal                GW                         SIP UA
128.59.21.152                                         128.59.19.194
|                             |                            |
|            SETUP            |                            |
|---------------------------->|                            |
| destination:hgs@cs.columbia.edu                          |
| fastStart={g711Ulaw,Tx},    |                            |
|  {g711Ulaw,Rx,128.59.21.152:10000}                       |
|                             |                            |
|                             |          INVITE            |
|                             |===========================>|
|                             | To:hgs@cs.columbia.edu     |
|                             | c=IN IP4 128.59.21.152     |
|                             | m=audio 10000 RTP/AVP 0    |
|                             |                            |
|                             |          200 OK            |
|                             |<===========================|
|                             | c=IN IP4 128.59.19.194     |
|                             | m=audio 8000 RTP/AVP 0     |
|           CONNECT           |                            |
|<----------------------------|                            |
|  fastStart={g711Ulaw,Tx,128.59.19.194:8000},             |
|            {g711Ulaw,Rx}    |                            |
|                             |            ACK             |
|                             |===========================>|
|                             |                            |
```

Figure 10: Call setup from H.323 terminal to SIP UA with FastConnect

point, which has a list of media supported by the H.323 endpoint, a SIP ACK message is formed with an updated session description reflecting the TCS. However, T is still unknown at this point, so that the SDP “m=” and “c=” lines remain as described above.

When the GW receives an H.245 Open Logical Channel (OLC) message, the GW acknowledges it with session information derived from the session description received from the SIP UA in the 200 (OK) response. When the first RTP packet of any media is received by the gateway from the SIP cloud, the GW knows what payload type is used by the SIP UA for that media type and it can send OLC to the H.323 cloud. RTP packets received until OLC Ack is received are ignored or blocked.

The problem with this approach is that RTP packets from the SIP UA cannot directly go to the H.323 terminal, but are instead routed through the RTP translator, violating requirement 4 in Section 4. This problem can be solved by having the GW send a re-INVITE to the SIP endpoint after the logical channels have been opened. This new INVITE message indicates media transport addresses (T) of the H.323 endpoint and not that of the translator.

A second problem is caused by the different interpretation of dynamic payload type switching in H.323 and SIP. When the TCS is mapped to SDP, the “m=” line is likely to list more than one payload type. This indicates to the SIP-controlled media agent that it may switch dynamically between all the payload types listed, without any H.323 or SIP signaling. However, in H.323, switching payload types requires Open Logical Channel signaling. This problem can be solved by restricting the SDP sent to the SIP endpoint to contain only one payload type per media description line. It is not clear how this payload type should be chosen or how the SIP endpoint can then switch payload types.

A third problem is that mapping a generic TCS to SDP requires enhancing SDP or SIP so that it can indicate different capability descriptors of H.245. For example, we could use SIP multipart message bodies, with each body part containing the SDP mapped from a single capability descriptor.

(Section 7 describes how to calculate a common subset of H.245 and SDP capabilities.) To solve this problem, the gateway could send a SIP OPTIONS request to the SIP UA and use that to calculate the common subset of capabilities.

5.2.4 Call from H.323 Cloud to SIP Cloud Mapping H.245 Open Logical Channel (OLC) to SDP

In the second approach, on receipt of a Q.931 SETUP message, the gateway sends a SIP INVITE request as in Section 5.2.3. The gateway performs the H.323 capability exchange with the H.323 cloud without involving the SIP UA. The gateway then calculates the subset of capabilities from the H.323 TCS and the SDP contained in the 200 (OK) response to the INVITE. The GW then sends an H.245 OpenLogicalChannel message for each of the media present in this subset. The OpenLogicalChannelAck message received from H.323 terminal will have the media transport addresses (T) of the H.323 terminal. On receipt of OpenLogicalChannelAck for all the OpenLogicalChannel messages, the GW sends a SIP ACK message with the new transport addresses. This call scenario is shown in figures 11 and 12.

Dynamic switching of H.245 Mode or Logical Channels is accomplished using SIP re-INVITE. For example, if video logical channel is opened from H.323 to GW after initial call setup procedure (i.e., Logical Channels for audio are already opened), then the gateway sends a re-INVITE message to the SIP side with new SDP describing the video capability also. When the gateway receives 200 response from the SIP side, it sends OpenLogicalChannelAck to H.323 side with the media transport address as received in SDP in the response. The gateway will also initiate OpenLogicalChannel procedure for the video channel in GW to H.323 direction.

If the media transport address of SIP UA changes during a call for a particular logical channel, (e.g.,

```
H.323                        GW                           UA
128.59.21.152             128.59.19.216           128.59.19.194
|                             |                            |
|           Setup             |                            |
|---------------------------->|                            |
| (hgs@cs.columbia.edu)       |    INVITE w/default SDP    |
| (no fastStart)              |===========================>|
|                             | c=IN IP4 128.59.19.216     |
|                             | m=audio 0 RTP/AVP 0        |
|                             |                            |
|                             |        200 OK              |
|          Connect            |<===========================|
|<----------------------------| c=IN IP4 128.59.19.194     |
|                             | m=audio 8000 RTP/AVP 8     |
|            TCS              |                            |
|<--+--+--+--+--+--+--+--+--+-|                            |
|  {g711Alaw for tx and rx}   |                            |
|                             |                            |
|          TCSAck             |                            |
|--+--+--+--+--+--+--+--+--+->|                            |
|                             |                            |
|            TCS              |                            |
|--+--+--+--+--+--+--+--+--+->|                            |
|  {g711Alaw and g711Ulaw}    |                            |
|                             |                            |
|           TCSAck            |                            |
|<--+--+--+--+--+--+--+--+--+-|                            |
|                             |                            |
|            OLC              |                            |
|<--+--+--+--+--+--+--+--+--+-|                            |
|  {mode=g711Alaw}            |                            |
|                             |                            |
|           OLCAck            |                            |
|--+--+--+--+--+--+--+--+--+->|      ACK with updated SDP  |
|  {Rx=128.59.21.152:10000}   |===========================>|
|                             |  c=IN IP4 128.59.21.152    |
|                             |  m=audio 10000 RTP/AVP 8   |
|            OLC              |                            |
|--+--+--+--+--+--+--+--+--+->|                            |
|  {mode=g711Alaw}            |                            |
|                             |                            |
|           OLCAck            |                            |
|<--+--+--+--+--+--+--+--+--+-|                            |
|  {Rx=128.59.19.194:8000}    |                            |
```

Figure 11: Call from H.323 to SIP with Conversion between OLC and SDP

```
SIP UA                       GW                   H.323 terminal
128.59.19.194            128.59.19.216            128.59.21.152
|                             |                            |
|           INVITE            |                            |
|============================>|          Setup             |
|  (To:kns10@columbia.edu)    |--------------------------->|
|  (c=IN IP4 128.59.19.194)   | (destination:kns10@cs.columbia.edu)
|  (m=audio 8000 RTP/AVP 0)   | (fastStart={g711Ulaw,Tx}   |
|                             |  {g711Ulaw,Rx,128.59.19.194:8000})
|                             |                            |
|                             |          Connect           |
|                             |<---------------------------|
|                             |    (fastStart absent)      |
|                             |                            |
|                             |           TCS              |
|                             |--+--+--+--+--+--+--+--+--->|
|                             |    {g711Ulaw Tx and Rx}    |
|                             |                            |
|                             |          TCSAck            |
|                             |<--+--+--+--+--+--+--+--+---|
|                             |                            |
|                             |           TCS              |
|                             |<--+--+--+--+--+--+--+--+---|
|                             |    {g711Alaw and g711Ulaw} |
|                             |                            |
|                             |          TCSAck            |
|                             |--+--+--+--+--+--+--+--+--->|
|                             |                            |
|                             |           OLC              |
|                             |--+--+--+--+--+--+--+--+--->|
|                             |     {mode=g711Ulaw}        |
|                             |                            |
|                             |          OLCAck            |
|           200 OK            |<--+--+--+--+--+--+--+--+---|
|<============================|  {Rx=128.59.21.152:10000}  |
|  c=IN IP4 128.59.21.152     |                            |
|  m=audio 10000 RTP/AVP 0    |                            |
|                             |                            |
|             ACK             |                            |
|============================>|            OLC             |
|                             |<--+--+--+--+--+--+--+--+---|
|                             |      {mode=g711Ulaw}       |
|                             |                            |
|                             |           OLCAck           |
|                             |--+--+--+--+--+--+--+--+--->|
|                             |   {Rx=128.59.19.194:8000}  |
```

Figure 12: Call from SIP to H.323 with Conversion between OLC and SDP

as a result of re-INVITE initiated by the SIP side) then the gateway sends RequestChannelClose H.245 message to the H.323 terminal for the logical channel. H.323 terminal will close the logical channel and will re-open it using OpenLogicalChannel. The changed media transport address of SIP UA can then be returned to H.323 terminal in OpenLogicalChannelAck message.

In this approach, RTP packets can be sent directly between the two endpoints. However, the SIP UA is restricted to algorithms chosen by the gateway. Since these algorithms are derived from the subset of H.323 and SIP capabilities, communications should still be possible.

A small problem with this message flow sequence is that ACK timeout on the SIP side and OLC timeouts on H.323 side may not match. This may result in lots of retransmission in SIP network. To avoid this, the gateway may choose to send an ACK immediately upon receipt of the 200 (OK) response from the SIP UA and then re-INVITE with an updated SDP after all OpenLogicalChannelAcks have been received from the H.323 endpoint.

A third approach would accept the H.323 SETUP message before forwarding it to SIP endpoint. However, this approach violates some of the requirements listed before and are not deemed appropriate by the authors.

We prefer the mapping of SDP to and from OpenLogicalChannel (section 5.2.4) for the following reasons:

- Mapping OLC is simpler than mapping TerminalCapabilitySet to SDP, which requires modifications to SIP or SDP.
- It avoids the introduction of a temporary RTP translator.

6 Address Conversion between H.323 and SIP

A SIP address can be either a SIP URL or any URI. This document only describes the translation of the SIP (“sip:”), telephone (“tel:”) and H.323 (“h323:”) URL schemes.

The BNF of a SIP address is given below for reference:

```
SIP-Address      =  (name-addr | addr-spec)
name-addr        =  [display-name] ”<” addr-spec ”>”
addr-spec        =  SIP-URL
SIP-URL          =  ”sip:” [ userinfo ”@” ] hostport url-parameters
                    [headers]
userinfo         =  user [ ”:” password ]
hostport         =  host [ ”:” port ]
host             =  hostname | IPv4address
url-parameters   =  *(”;” url-parameter)
url-parameter    =  user-param | . . .
```

In the url-parameter, only the user-param parameter is relevant. The user name may be a telephone number.

H.323 addresses are typically sequences of Alias Addresses (see H.225.0 [8]). The ASN.1 description of an H.323 Alias Address is:

```
H323-Alias-Address ::= CHOICE
```

```
{
  e164    IA5String (SIZE(1..128)) (FROM("0123456789\#*,")),
  h323-ID BMPString (SIZE (1..256)),
  ...,
  url-ID  IA5String ( SIZE(1 .. 512)),-- URL Style address
  transport-ID TransportAddress,  -- IPv4, IPv6, IPX etc.,...
  email-ID IA5String (SIZE(1..512)),
                      -- rfc822 compliant email address
  partyNumber PartyNumber
}
```

The PartyNumber parameter is not described in this document and is left for further study. Telephone numbers can be conveyed via e164 field of H323-Alias-Address or called/calling party number fields of Q.931 message.

6.1 Converting SIP Addresses to H.323 Addresses

6.1.1 h323-ID

The SIP-Address is stored as is in the h323-ID of the Alias Address. If the SIP-Address contains more than 256 characters, only the addr-spec part is copied. If the addr-spec exceeds 256 characters, the gateway generates a SIP response of 414 (Address Too Long). Each BMP character in h323-ID stores the corresponding text character in the SIP Address. (BMP stands for basic multilingual plane i.e., Basic ISO/IEC 10646-1 (unicode) character set)

The h323-ID MUST always be generated so that a terminal running version 1.0 of H.323 (which supports only e164 and h323-ID, but does not support transport-ID, url-ID or email-ID) can still decode the address.

6.1.2 e164

If the SIP-Address's user is a telephone-subscriber, user-param is set to phone and the user part does not contain a "w", it is converted to the e164 field of Alias-Address. The e164 field only allows characters from the set "0123456789#*,". Thus, any leading "+" is removed from the SIP telephone-subscriber part, as are any visual separators "-" and ".". The pause "p" is replaced with ",".

6.1.3 url-ID

The SIP-URL part of the SIP address is copied verbatim to the url-ID parameter. If the SIP URL exceeds 512 bytes in size, the GW generates the SIP status 414 (Address too long).

6.1.4 email-ID

The user and host parts are used to generate an email identifier, as in "*user@host*", which is stored in the email-ID field of AliasAddress. If the size exceeds 512 characters, the GW generates the SIP status 414 (Address Too Long).

6.1.5 transport-ID

If the host part of the SIP-URL is indicated as a dotted quad, it is translated into a transport-ID. If a port parameter is present in the SIP address, the number is used. Otherwise, the port number depends on the context. For example, for the destination address of H.323 SETUP messages, it is set to 1720, otherwise it is set to 0.

> Although a numeric IP address requires no further address resolution, it is worth noting that other fields (e164, url-ID, h323-ID) are also needed. If the destination is a VoIP gateway, for example, then an Internet telephony gateway destination is mapped from the e164 field or the called party number.

6.1.6 Examples

- The SIP Address “sip:j.doe@big.com” is converted to an H.323 Address sequence with three elements: { h323-ID=”sip:j.doe@big.com”, url-ID=”sip:j.doe@big.com”, email-ID=“j.doe@big.com” }

- The SIP Address “sip:+1-212-555-1212:1234@gateway.com; user=phone” is converted to the H.323 Address: { e164=”12125551212”, h323-ID=”sip:+1-212-555-1212:1234@gateway.com”, url-ID =”sip:+1-212-555-1212:1234@gateway.com”, email-ID=“+1-212-555-1212:1234@big.com”}

- The SIP Address ”sip:alice@10.1.2.3” is converted to H.323 Address: { h323-ID=”sip:alice@10.1.2.3”, url-ID=”sip:alice@10.1.2.3”, tranport-ID= IPAddress 10.1.2.3:1720, email-ID=“alice@10.1.2.3” }

- The SIP Address “A. Bell ¡sip:a.g.bell@bell-tel.com¿” is converted to H.323 Address: { h323-ID=”A. Bell ¡sip:a.g.bell@bell-tel.com¿”, url-ID=”sip:a.g.bell@bell-tel.com”, email-ID=“A. Bell ¡a.g.bell@bell-tel.com¿” }

6.2 Converting H.323 Addresses to SIP Addresses

In H.323, addresses are typically a sequence of Alias Addresses (referred to as H.323 addresses in this document). Since it is not possible to convert all the addresses to a single SIP Address, the gateway will have to drop some of the addresses. However, a gateway MAY try more than one converted addresses either sequentially or in parallel.

The conversion is done in the following order. If the conversion succeeds in one step, the conversion concludes and the remaining steps are ignored.

If a url-ID is present and it is a SIP-URL, then it is used as is in the SIP Address.

If an h323-ID is present and it can be parsed as a valid SIP-Address, it is used. This is needed when talking to an H.323 terminal running version 1.0.

If the transport-ID is present and it does not identify the gateway, then it forms the hostport portion of the SIP URL and the user portion is constructed using h323-ID or e164.

If the email-ID is present, then it is used in the SIP-URI. The email-ID is prefixed by the scheme name “sip:”.

If all these efforts fail, then the gateway MAY attempt to construct a legal SIP Address using the information available. For example h323-ID may become the display-name, e164 may become the user and host may be some default domain name.

If the gateway is configured to route all calls to a default proxy, then it will forward whatever SIP addresses it can form (from the H.323 Alias Address) to the proxy. This may be needed when the gateway implementation is split into two (physically separate) parts, namely an H.323 terminal and a SIP user agent. The H.323 terminal receives the call, maps the H.323 address to the SIP address and forwards the request to the SIP proxy server.

7 Calculating a Common Subset of Capabilities

The *capability set* of a terminal or a user agent refers to the set of algorithms for audio, video and data that it can support. It also conveys information about constraints in the selection of algorithms it may have. For example, due to limited bandwidth, a terminal may indicate that it can use either G.711 without video or G.723.1 with H.261 video.

The *operating mode* of a call refers to the selected algorithms which are used for the actual transfer of media. To determine the operating mode for a call it is often necessary to find out the intersection of the capabilities of the endpoints in the conference. This section presents a way to calculate this intersection of the capability sets described by H.245 Terminal Capability Set (TCS) and that by SDP.

A *maximal intersection* of two capability sets is a capability set which is a subset of both the capability sets and no other superset of the maximal intersection is a subset of those capability sets. It can be proven that if M is an operating mode for capability set $C1$ as well as for capability set $C2$, then M will be an operating mode for maximal intersection of $C1$ and $C2$. Thus, we fulfill requirement 5 described in Section 4.

H.245 defines *Terminal Capabilities* as a list of capability descriptors, ordered in decreasing preference. Any one of the capability descriptors can be used for selecting operating modes. Each capability descriptor includes a simultaneous capability set. Each element in the simultaneous capability set is an alternative capability set. Each element in the alternative capability set represents an algorithm. Each algorithm has a payload type and can be fully described by the payload type, a profile and some optional attributes.

Convention:

```
{ } capability descriptor or simultaneous capability set
[ ] alternative capability set
```

Example: Let a1, a2, a3, a4, a5 be audio algorithms and v1, v2, v3 be video algorithms. $C1$ represents a capability set with two capability descriptors:

```
C1 = { [a1, a2, a3] [v1, v2] }
     { [a1, a4, a5] [v1] }
```

Operating modes could be (a1, v1), (a1, v2), (a4, v1), (a5), etc. Note that (a4, v2) is not an operating mode since a4 and v2 are drawn from different capability descriptors.

Let C2 be another capability set.

```
C2 = { [a1, a4, a2] [v1, v2, v3] }
     { [a1, a2, a5] [v1, v3] }
```

The maximal intersection of C1 and C2 is

```
C = { [a1, a2] [v1, v2] }
    { [a1, a4] [v1] }
    { [a1, a5] [v1] }
```

Note that there are other capability sets which are intersections of C1 and C2 (e.g., {[a1,a2][v2]}), but they are subsets of C and hence can be derived from C.

7.0.1 Algorithm for Finding Maximal Intersection of Capability Sets

An algorithm to find the maximal intersection of any two capability sets $C1$ and $C2$ is given below:

1. Set the result C to the empty set.

2. For each pair of capability descriptors ($d1$, $d2$), where $d1$ is from $C1$ and $d2$ is from $C2$, derive the permutations of alternative sets, $s1$ and $s2$.

   For each such permutation, where $s1$ is from $d1$ and $s2$ is from $d2$, intersect $s1$ and $s2$ (written as s=$s1$ˆ$s2$) and add s to C.

3. Remove duplicate entries from C.

Example: Using the example with $C1$ and $C2$ given above, the outer loop runs for four iterations, since $C1$ and $C2$ both have two descriptors.

1.
   ```
   d1 = {[a1,a2,a3][v1,v2]},
   d2 = {[a1,a4,a2][v1,v2,v3]}
   ```

   Inner loop runs for 2 iterations:

   ```
      1) {[a1,a2,a3]ˆ[a1,a4,a2],[v1,v2]ˆ[v1,v2,v3]}
                  = {[a1,a2][v1,v2]}
      2) {[a1,a2,a3]ˆ[v1,v2,v3],[v1,v2]ˆ[a1,a4,a2]}
                  = {[][]}  /* Empty set */
   ```

2.
   ```
   d1 = {[a1,a4,a5][v1]},
   d2 = {[a1,a4,a2][v1,v2,v3]}
     1) {[a1,a4,a5]ˆ[a1,a4,a2], [v1] ˆ[v1,v2,v3]}
                  = {[a1,a4][v1]}
     2) {[a1,a4,a5]ˆ[v1,v2,v3],[v1]ˆ[a1,a4,a2]}
                  = {[][]}  /* Empty set */
   ```

3.
   ```
   d1 = {[a1,a2,a3][v1,v2]},
   d2 = {[a1,a2,a5][v1,v3]}
     1) {[a1,a2,a3]ˆ[a1,a2,a5],[v1,v2]ˆ[v1,v3]}
                  = {[a1,a2][v1]}
     2) {[a1,a2,a3]ˆ[v1,v3],[v1,v2]ˆ[a1,a2,a5]}
                  = {[][]}  /* Empty set */
   ```

4.
   ```
   d1 = {[a1,a4,a5][v1]},
   d2 = {[a1,a2,a5][v1,v3]}
     1) {[a1,a4,a5]ˆ[a1,a2,a5],[v1]ˆ[v1,v3]}
                  = {[a1,a5][v1]}
     2) {[a1,a4,a5]ˆ[v1,v3],[v1]ˆ[a1,a2,a5]}
                  = {[][]}  /* Empty set */
   ```

After these iterations the intersection set becomes

```
{ [a1,a2] [v1,v2] } { }
{ [a1,a2] [v1] } { }
{ [a1,a4] [v1] } { }
{ [a1,a5] [v1] } { }
```

After removing duplicates, the maximal intersection is

```
{ [a1,a2] [v1,v2] }
{ [a1,a4] [v1] }
{ [a1,a5] [v1] }
```

Since H.323 does not require that all algorithms listed within a single alternative capability have the same media type, we need the inner loop to find out all the possible combinations.

For example, if C1 = {[a1,a2,a3] [a1,a4,v2,v1]} and C2 = {[a1,a4,v2] [v1,v2,v3]}, then the above algorithm correctly finds the intersection as {[a1] [v1,v2]} {[a1,a4,v2]}

8 Implementation Requirements

This section lists the messages which MUST be supported by the signaling gateway. It also highlights the typical values for parameters for the messages.

8.1 H.323 (H.225.0 and H.245)

All the messages which are mandatory in the Q.931 portion of H.225.0 and H.245 MUST be supported. RAS is optional; if used, all messages that are mandatory in RAS MUST be supported. Parameter values (if not specified in this document) MUST be derived from H.225.0 version 2.0 and H.245 version 4.0 for Q.931 and H.245 messages, respectively. This assures that requirement 1 in Section 4 is fulfilled.

8.1.1 Handling of Q.931 Messages

The gateway SHOULD support the Q.931 messages listed in Table 1. An entry of "not applicable" in the table means that it is not visible to the SIP endpoint and is only local to the gateway's H.323 stack.

The gateway MUST NOT close the Q.931 call connection after the call is established. However, if the call is routed through a gatekeeper and the gatekeeper closes the Q.931 call connection, the gateway MUST comply with H.323 and MUST NOT assume that the call is closed as long as H.245 channel is open. If the Q.931 TCP connection is closed without closing the call connection, then the gateway SHOULD try reopening the TCP connection, as specified by H.323. In case of failure such as TCP connection refused or TCP connection timeout, the gateway SHOULD close the call on the SIP side also by sending a BYE.

Q.931-specific information elements, other than user-user information element (UUIE), do not affect the operation of this gateway, however they are required for interoperation with other H.323 entities. The specific fields of UUIE used in translating to SIP message are given in Appendix A.

Bearer Capability: Information transfer capability (octet 3, bits 0–5): Unless some other restrictions apply (e.g., the gateway is connected to a bandwidth-restricted ISDN network), the parameter

Message	GW sends to H.323	H.323 sends to GW
Alerting	Supported	Supported
Call proceeding	Supported	Supported
Connect	Supported	Supported
Progress	Not applicable	Not applicable
Setup	Supported	Supported
Setup Ack	Not applicable	Not applicable
Release Complete	Supported	Supported
User Information	Not applicable	Not applicable
Information	Not applicable	Not applicable
Notify	Not applicable	Not applicable
Status	Not applicable	Not applicable
Status Inquiry	Not applicable	Not applicable
Facility	Not applicable	Not applicable

Table 1: Support for Q.931 messages

A “Not applicable” entry in the table means that it is not visible to the SIP endpoint and is only local to the gateway’s H.323 stack.

SHOULD be set to “unrestricted digital information” or “restricted digital information” on outgoing side. If the gateway knows that the call is going to be voice only, it may choose to set it as “speech” or “3.1 kHz Audio”. The GW ignores this field on incoming requests.

Information Transfer Rate and Rate multiplier: If bandwidth information is available from the gatekeeper or some external means (e.g., from bandwidth information in SDP message), then information transfer rate and rate multiplier may be set to values reflecting the bandwidth, else they should be set to some high value as appropriate. This way the bandwidth is not limited to 64 kb/s or 128 kb/s. On the incoming side these values SHOULD be ignored. Note that in Q.931 message the only possible values are multiples of 64 kb/s.

Layer 1 protocol (octet 5, bits 1–5): For outgoing Q.931 messages, the parameter is set to H.221 (’00101’), indicating an H.323 video phone call, unless the gateway knows that the call is going to be voice only (e.g., if this is hardcoded in the gateway). In that case, it may encode the parameter as G.711 A-law or mu-law to indicate this.

For incoming Q.931 messages, the GW ignores this field.

Calling or Called party number: For outgoing Q.931 messages, the GW translates the SIP request-URI into an e164 number, as described in Section 6. The calling/called party subaddress is not included in Q.931 messages originating from the GW.

For incoming Q.931 messages, the gateway relies on user-user information element for addresses (e.g., sourceAddress and destinationAddress fields of UUIE) and ignores the Q.931 parameter. However, if the calling/called party number is present and e164-ID is not present in the H.323 Alias Address then the calling/called party number is used instead of e164-ID while translating address in section 6.

H.323 specifies that the called and calling party Subaddress fields are needed for some circuit switched call scenarios and they SHOULD NOT be used for packet based network side only calls.

SIP	status	releaseCompleteReason
400	Bad Request	undefinedReason
401	Authentication Required	noPermission
402	Payment Required	undefinedReason
403	Forbidden	noPermission
404	Not Found	unreachableDestination
406	Not Acceptable	undefinedReason
407	Proxy Authentication Required	noPermission
409	Conflict	undefinedReason
410	Gone	undefinedReason
413	Request Entity Too Large	undefinedReason
414	Request-URI Too Large	badFormatAddress
415	Unsupported Media Type	undefinedReason
420	Bad Extension	badFormatAddress
480	Temporarily not available	unreachableDestination
483	Too Many Hops	undefinedReason
484	Address Incomplete	badFormatAddress
485	Ambiguous	badFormatAddress
486	Busy Here	destinationRejection
600	Busy Everywhere	destinationRejection
603	Decline	destinationRejection
604	Does not exist anywhere	unreachableDestination

Table 2: Mapping between SIP status codes and reason fields

Display: For incoming Q.931 messages, the GW MAY copy the Display IE to the display parameter of the SIP To header field.

Similarly, for outgoing Q.931 messages, the Display parameter MAY be copied from the display parameter of the SIP To field.

Cause: For incoming Q.931 messages, the Q.931 Cause information element and/or the UUIE reason field are mapped to the appropriate SIP status response code, as described in Table 2. H.225.0 [8] specifies that either the Cause information element or the releaseCompleteReason MUST be present. It also gives a mapping between the Cause information element and the releaseCompleteReason. Table 2 gives the mapping between releaseCompleteReason and the appropriate SIP status response.

Similarly, for outgoing Q.931 messages, the Q.931 Cause information element and the UUIE reason field are derived according to Table 2.

User-User-Information-Element: Below, we detail the fields in UUIE which are relevant to H.323-SIP conversion. Other fields are interpreted as defined by H.225.0.

sourceInfo/destinationInfo: In all messages from the GW, this field SHOULD be set to indicate that this endpoint is a gateway. However, the sequence of supported protocols in “GatewayInfo” may be empty.

H.245SecurityMode, tokens, cryptoTokens: These fields are interpreted as in H.323. Note that since H.245 is terminated at the gateway, this kind of security information is not relevant to the SIP cloud.

fastStart: FastStart PDUs contain the OpenLogicalChannel (OLC) messages. The gateway converts incoming OLC messages to a SDP message body. One SDP media description line ("m=") is generated for each distinct session-ID. All logical channels with same session-ID appear as payload types in a single SDP media description line. When converting SIP to H.323, the SDP message is converted to a list of OpenLogicalChannel messages, one per payload type. H.323 endpoint will select atmost one OLC per session-ID. This selected OLC is returned by the H.323 endpoint in the fastStart field of Q.931 Connect message. When converting H.323 to SIP, each OLC in fastStart corresponds to a payload type of SDP. All the OLC messages with same session-ID form a single media description ("m=") line.

The parameters for the Q.931 SETUP message are listed below.

sourceAddress: Converted to/from SIP header From field as described in section 6.

destinationAddress: Converted to/from SIP header To field as described in section 6.

destCallSignalAddress: If the To SIP header field contains a numeric host identifier then destCallSignalAddress is set to the IPv4 address represented by the numeric identifier.

conferenceGoal: Set to "create" in outgoing Q.931 messages. (Additional values may be supported in future versions of this specification that support conferencing.)

remoteExtensionAddress: Not present in outgoing Q.931 messages. For incoming Q.931 messages, this parameter is combined with the DestinationAddress parameter to generate the SIP To header field and the request-URI.

mediaWaitForConnect: Set to "false" in outgoing Q.931 messages. Ignored in incoming Q.931 messages, as media transmission is transparent to the gateway.

canOverlapSend: Set to "false" in outgoing Q.931 messages and ignored in incoming Q.931 messages since this version of the specification does not support overlap sending.

Use of the Q.932 facility message for call redirection is for further study.

8.1.2 Handling H.245 Messages

Table 3 details how a gateway handles H.245 messages. An entry of "not applicable" means that the message does not affect the behavior within the SIP cloud.

The remainder of this subsection lists the possible values of some of the fields of H.245 messages. Refer to H.245 version 4.0 for description and details of the ASN.1 structures for H.245.

MasterSlaveDetermination: The terminalType parameter is set to indicate that this terminal is a gateway. H.323 specifies a set of numerical values of terminalType for different types of terminals. For example, a gateway without a multipoint controller (MC) has a terminalType of 60; A gateway with a MC and no multipoint processor (MP) has a terminalType value of 80. Other values of terminalType

Message	REQUIRED or Not applicable
MasterSlaveDetermination/Ack/Rej/Rel	Not Applicable
TerminalCapSet/Ack/Reject/Release	REQUIRED
Send TerminalCapabilitySet	Not Applicable
OpenLogicalChannel/Ack/Reject	REQUIRED
OpenLogicalChannelConfirm	Not Applicable
CloseLogicalChannel/Ack	REQUIRED
RequestChannelClose	OPTIONAL
RequestMode/Ack/Rej/Rel	RECOMMENDED
RoundTripDelayReq/Res	Not applicable
MaintenanceLoopReq/Ack/Reject	Not supported
MaintenanceLoopOffCmd	Not supported
CommunicationModeReq/Res/Cmd	For further study
ConferenceReq/Res/Cmd/Indic	For further study
EndSessionCommand	REQUIRED
FlowControlCommand	For further study
Encryption Command	For further study
Jitter Indication	For further study
User Input	OPTIONAL
H2250MaxSkewIndic	For further study
MClocationIndication	For further study
FunctionNotUnderstood	Not Applicable
FunctionNotSupported	Not Applicable
vendorIdentifier	Not Applicable
MiscCommand/Indication	For further study

Table 3: Support for H.245 messages.
An entry of “not applicable” means that it is not visible to the SIP endpoint and is only local to the gateway’s H.323 stack.

are not relevant to this gateway in the case where media traffic is transparent. See H.323 [3] for other possible values of terminalType.

TerminalCapabilitySet: **multiplexCapability::h2250Capability:** maximumAudioDelayJitter should be set to max possible value as specified by H.323. MultipointCapabilities should reflect minimum capability of Centralized Control/ Audio/ Video/ Data. Other conferencing capabilities are for further study. RTCP videoControlCapability should be set to false because anyway H.245 indications have to be used for this purpose. MediaPacketizationCapability should contain the information about the dynamic payload types used by SIP endpoint. Transport Capability should be absent. redundancyEncodingCapability should be absent. This is not supported in this version. logicalChannelSwitchingCapability may be supported by the gateway’s H.323 stack. This makes mapping SIP re-INVITE easier. t120DynamicPortCapability is set to false because T120 data is not supported in this version.

CapabilityTableEntry and

CapabilityDescriptor are mapped from the session description given by SDP. A single capability descriptor is used in H.245. All the payload types on a single media description line (m=) are combined to form an alternative capability set in H.245. All such media description lines are combined to form a simultaneous capability set (or a capability descriptor). Mapping multiple SDP received in multipart body of SIP to multiple capability descriptor is for further study.

Capability: H233Encryption is not applicable.

H235Security is not applicable.

DataApplication capability is not supported in this version of the specification.

ConferenceCapability is for further study and is not supported in this version of the specification.

UserInputCapability may be supported by the gateway. This is used to convey DTMF digits. Use of the SIP INFO method is being considered for this purpose.

maxPendingReplacementFor is not applicable.

Audio and Video: A capability in H.323 represents a payload type. Refer to

`http://www.iana.org/assignments/media-types/media-types`

for a list of MIME types and

http://www.iana.org/assignments/rtp-parameters

for a list of static RTP payload types. Use of static RTP payload types in SDP is discouraged. The gateway should maintain a list of all currently available payload types and media formats and the corresponding RFC numbers. (An intelligent gateway MAY periodically download and parse these HTML pages to update its database).

The predefined audio and video capabilities are mapped to appropriate media format and RTP payload type. This mapping is given in this document for ease of reference. This mapping should be used by the gateway to convert the H.323 capability to an SDP media description. When converting from H.323 to SDP, the gateway SHOULD use dynamic payload type. When converting from SDP to H.323, the gateway SHOULD NOT use dynamic payload types because many current implementations do not support these. However, the gateway MUST be able to receive dynamic payload types, in both H2250Capability.MediaPacketizationCapabilty.RTPPayloadType and in H2250LogicalChannelParameters.MediaPacketization. When dynamic RTP payload type are used, H225LogicalChannelParameters.dynamicRTPPayloadType MUST match the payload type description given in mediaPacketization.

AudioCapability: A subset of IANA-registered formats and H.323-supported capabilities are listed in Table 4.

Note that H.323 only supports a clock rate of 8000 Hz; other values cannot be mapped to H.323. TBD: A fmtp SDP attribute for silence suppression should be defined if silence suppression is on. TBD: Another possible fmtp attribute could be the list of annexes which are supported. This is useful in translating g729AnnexB, g729AnnexAwAnnexB, g7231AnnexC and so on to SDP.

VideoCapability: The mapping of video encodings is shown in Table 5. The Video MPI (Mean Picture Interval) is mapped to the SDP attribute “framerate” as follows:

mpi = 30 / framerate

H.323	IANA	payload type	clock/channels	RFC
g711Alaw64k	PCMA	8	8000/1	RFC1890
g711Ulaw64k	PCMU	0	8000/1	RFC1890
g711Alaw56k	N/A			
g711Ulaw56k	N/A			
g722-64k	G722	9	8000/1	RFC1890
g722-56k	N/A			
g722-48k	N/A			
g7231	G723	4	8000/1	None
g728	G728	15	8000/1	RFC1890
g729	G729?	Dynamic/18?	8000/1	-
g729AnnexA	?	Dynamic	8000/1	?
g729wAnnexB	?			
g729AwB	?			
g7231AnnexC	?			
gsmFullRate	GSM	3	8000/1	RFC1890
gsmHalfRate	GSM-HR	Dynamic	8000/1	-
gsmEnhFullRate	GSM-EFR	Dynamic	8000/1	-

Table 4: Audio capability mapping

H.323	IANA	Payloadtype	clock	RFC
h261VideoCap	H261	31	90000	RFC2032
h262VideoCap	?			
h263VideoCap	H263/H263+?	34	90000	RFC2190/2429?

Table 5: Video capability mapping.

It is assumed that 29.97 Hz is rounded to 30 Hz when calculating the framerate. So MPI of 1 become framerate 30.0, similarly MPI of 2 becomes framerate 15. However, the gateway shall do proper rounding error correction on the incoming side. So framerate of 29.97 should also map to MPI of 1. Note that in SDP any possible value for framerate is allowed, but in H.323 only multiples of 1/29.97 are allowed. The gateway should convert the framerate to the next lower value allowed in H.323. For example, a framerate of 12.3 frames per second in SDP is converted to an MPI value of 3 which is equivalent to 10 frames per second.

DataApplicationCapability: Not supported in this version of the specification.

Use of RSVP (Resource reservation protocol) to handle QoS (Quality of service) is for further study.

A Detailed Description of Gateway Behavior

This section describes how messages are processed by a SIP–H.323 signaling gateway. The discussion is split into two subsections, with SIP-originated requests discussed in Section A.1 and H.323-originated requests in Section A.2. Only fields relevant to the conversion are presented here. Other parameters are specific to either H.323 or SIP and can be generated by the respective protocol engine in the gateway without conversion.

The gateway maintains, apart from other call-state information, the capability sets and operating mode for each call. Capability sets are maintained for each H.323 and SIP endpoint, both receive and transmit directions. Operating mode contains the modes in each direction (SIP to H.323 and H.323 to SIP).

A.1 SIP-originated Requests

A.1.1 Gateway Receives REGISTER

The GW sends a RAS RRQ message to the H.323 GK, where the callSignalAddress is the address of the GW, the terminalType is set to "gateway" and the terminalAlias is mapped from the To header of the REGISTER request.

The GW stores the SIP Contact header field. A "200 OK" SIP status response is sent after receiving a RAS RCF message.

A.1.2 Gateway Receives INVITE for a New Call

The GW MAY respond with a 100 (Trying) response to the SIP entity that sent the INVITE request. It stores the SDP information as the terminal's SIP capability and convert the capability to H.245 format.

If the gateway is registered with a gatekeeper, send a RAS ARQ message to the gatekeeper, where the destinationInfo and destCallSignalAddress is derived from the To SIP header, the srcInfo is derived from the From SIP header field and srcCallSignalAddress is the call signaling address of the gateway itself. The gatekeeper assigns an endpointIdentifier during registration. That value of endpointIdentifier is used in the endpointIdentifier field of the ARQ message.

Next, the GW should receive either a RAS ACF or ARJ message. If an ACF message is received, establish an Q.931 channel as described below. If an ARJ message is received, the behavior depends on the reason parameter:

CalledPartyNotRegistered: The gateway responds with 404 (Not Found).

callerNotRegistered: The gateway MAY register, with a RAS RRQ message, the SIP address with the gatekeeper and then retransmit the RAS request, with the endpointIdentifier returned in RCF. Alternatively, it MAY send a 400 (Caller not registered) response to the SIP entity.

incompleteAddress: Send 484 (Address Incomplete) response to SIP entity.

Other reasons: Send 400 (H.323 translation failure) response to SIP entity.

If the GW times out waiting for an ARQ response, it sends a SIP 504 (Gateway time-out) response.

If the gateway is not registered with a gatekeeper and it is able to resolve the SIP address to a H.323 address or if the gateway is registered and has received an ACF for the registration request from the gatekeeper, the GW sends a Q.931 SETUP message to the H.323 entity, where the sourceAddress is derived from the SIP From header, the destinationAddress is derived from the SIP To header or from the RAS ACF response, destCallSignalAddress is derived from the RAS ACF response or from the To SIP header. The remoteExtensionAddress is copied from RAS ACF if present or extracted from To SIP header if possible. sourceCallSignalAddress is the call signaling transport address of the gateway. fastStart PDUs are mapped from the session description in the INVITE message body.

Each SDP payload type entry is converted to an OLC message. All the payload types on the SDP same media description line have the same session id in the OLC messages. This identifies them as belonging to the same group and the receiving H.323 entity will select one of these. (TBD: needs more description)

If the gateway receives a Q.931 CallProceeding message, send a 100 (Trying) response to the SIP entity, if not already sent. If fastStart PDUs are present, store them.

If the gateway receives a Q.931 Alerting message, send a 180 (Alerting) response to the SIP entity, indicating that the final destination is ringing. If fastStart PDUs are present, store them.

If the gateway receives a Q.931 Connect message, the behavior depends on whether a FastStart indication is present.

If a FastStart indication is present, the GW maps the received OLCs to the SDP payload types contained in the original INVITE request. Format a new SDP packet with more constrained media description and correct media transport address of the H.323 entity. Now each media description line will contain a single payload type, depending on which OLC PDUs are present. The operating mode and H.323 capability set are set to this reduced set of payloads.

The SDP message is sent in a 200 (OK) response. The GW then waits for the ACK request from the SIP entity. If the gateway times out, it declares the call closed and terminates the H.323 call. Once an ACK has been received, the gateway may proceed with other H.245 signaling (CESE, RTDSE and so on).

If the H.323 entity does not support FastStart, the gateway proceeds with H.245 signaling as described below. First, it sends a TCS to the the H.323 entity and uses the stored SIP capability set to generate the H.245 capabilities.

If the gateway receives an H.245 TCS message, it updates the H.323 capability set and calculates maximal intersection of H.323 and SIP capability sets (call this C). Derive a suitable operating mode from C (say, M). For each element in M (for the data from the SIP UA to the H.323 terminal), send an H.245 OLC message to the H.323 entity. Use the transport address of the SIP capability set, derived from the SDP received in the original INVITE message.

If the gateway receives an OLC message and the logical channel is present in the operating mode from the H.323 terminal to the SIP UA, the gateway sends an OLCAck to the H.323 terminal. The OLCAck contains the transport address from the SIP capability set, again derived from the SDP in the INVITE message body. If the logical channel is not present in that operating mode, the gateway sends an OLCReject.

Once the gateway has received an OLCAck or OLCRej for all outstanding OLC requests, it updates the operating mode and sends a 200 (OK) . response to the SIP entity. The session description in that response is formed using the new operating mode and the transport addresses received in the H.245 OLCAcks.

The gateway should wait for the ACK request from the SIP entity. If the gateway times out, it should close the H.323 call. This concludes the description of the non-FastStart handling.

If, at any time, the gateway receives a Q.931 ReleaseComplete message, a H.323 call could not be established. The GW sends a 400 (Client Failure) with reason phrase “H.323 call failed”.

If the Q.931 SETUP times out, the GW sends a 504 (Gateway time-out) response.

If the SIP address is not resolved to an H.323 address, send a 501 (Not Implemented) response to SIP entity.

A.1.3 Gateway Receives INVITE for Existing Call

- Update the SIP capability set.

- Recalculate the operating mode, minimizing changes. An H.245 Mode Request message is sent if the operating mode has changed. If the Mode Request fails, either close the media channel or the call.

A.1.4 Gateway Receives BYE Request

The GW sends an H.245 Endsession to the H.323 entity. Upon receipt of a response or on timeout, the GW sends a Q.931 ReleaseComplete to H.323 entity. If the call was admitted by a GK, send a RAS DRQ (Disengage Request) message to the GK.

A.1.5 Gateway Receives OPTIONS Request

TBD: how do we querry H.323 capabilities without establishing the call?

A.2 H.323-Originated Requests

A.2.1 Gateway Receives RAS GRQ

The GW sends a RAS GCF (Gatekeeper Confirm) response to GRQ (Gatekeeper Request) only if the GW also contains a gatekeeper implementation (see Section 5.1.2).

A.2.2 Gateway Receives RAS RRQ

This is possible only if the GW also contains a gatekeeper implementation (see Section 5.1.2). On receipt of RRQ (Registration Request) the gateway sends a SIP REGISTER message to the SIP server where the To SIP header field is derived from the terminalAlias parameter; the Contact SIP header field indicates the gateway's location. The callSignalAddress received in RRQ message is stored internally by the gateway. The gateway may send multiple REGISTER requests if the sequence of terminalAlias can be mapped to multiple SIP addresses

Once the gateway receives a 2xx response to this REGISTER, it sends a RAS RCF (registration confirmation) message to the H.323 entity. If it receives any other status response or the REGISTER request times out, the gateway sends a RRJ (registration reject) to the H.323 entity.

A.2.3 Gateway Receives RAS ARQ

Receipt of this message indicates that the H.323 entity knows that the destination is reachable via this gateway. One simple implementation is to accept the admission request giving the callSignalAddress of the gateway itself. Alternatively, a procedure similar to that given for RAS LRQ, below, can be followed.

A.2.4 Gateway Receives RAS LRQ

If the gateway receives a RAS LRQ (Location Request) message, the gateway sends an OPTIONS message to the SIP entity, where the SIP entity address is resolved from the H.323 address. The To SIP header field is derived from the destinationAddress. The gateway MAY send multiple forking OPTIONS requests if the sequence of destinationAddresses can be mapped to multiple SIP addresses.

If it receives a 2xx response for the OPTIONS request, it sends a RAS LCF message to the H.323 with the CallSignalAddress of the gateway itself. If any other response is received or the request times out, the gateway MAY choose to remain silent or it may send a RAS LRJ to the H.323 entity.

A.2.5 Gateway Receives a Q.931 Setup

The gateway generates an ARQ/ACF sequence if required here as per H.323 standard. However, that is local to the H.323 stack and does not affect translation.

If fastStart is present, convert it to H.323 capability set, else build some default H.323 capability set. The gateway MAY send a Q.931 CallProceeding message to H.323 entity.

The gateway then sends an INVITE, where the To SIP header field is derived from the Q.931 destinationAddress and/or destCallSignalAddress. If destinationAddress is the gateway itself, then use remoteExtensionAddress. The From SIP header field is derived from sourceAddress and/or srcCallSignalAddress. The session description is constructed from the H.323 capability set.

If the gateway receives a 2xx response for the INVITE, it updates the SIP capability set using the session description in the response body. It then sends a Q.931 Connect message to the H.323 entity.

Then, the gateway sends an ACK request to the SIP entity.

Then, it sends an H.245 TCS to the H.323 entity using the SIP capability set.

If it receives a TCS, it updates the H.323 capability set and calculates the maximal intersection of the H.323 and SIP capability sets, called C. From C, the gateway derives a suitable operating mode (say M). For each element in M in the direction from SIP to H.323, send a H.245 OLC to the H.323 entity. The OLC messages use the transport addresses of the SIP capability set, derived from the session description in the 2xx response body.

If the gateway receives an OLC and the logical channel is present in the operating mode from H.323 to SIP, it responds with an OLCAck. The OLCAck uses the transport addresses of the SIP capability set. If the logical channel is not present in the operating mode, the gateway sends an OLCReject

Once the gateway has received OLCAck or OLCRej for all the requests, update the operating mode. Then, the gateway sends a re-INVITE. The session description is formed using the new operating mode if it is different from what was sent in the first INVITE message and the transport addresses received in OLCAcks. The gateway should wait for a 2xx response from the SIP entity and respond with an ACK request. If it times out or if it fails, it should close the call.

If the gateway receives a 180 (Alerting) SIP response, send a Q.931 Alerting message to the H.323 entity.

If the gateway receives any other 1xx SIP response, it sends a Q.931 CallProceeding message to H.323, but only if not already sent for this call.

If no response is received or a failure response, the gateway sends a Q.931 ReleaseComplete message to the H.323 entity.

A.2.6 Gateway Receives Mode Request or Change in Logical Channels

Update operating modes, Send re-INVITE to SIP entity. If that fails then reject the Mode Request or Open Logical Channel request.

A.2.7 Gateway Receives H.245 EndSession

If the gateway receives a H.245 EndSession, it closes the H.245 call. Send H.245 EndSession and Q.931 ReleaseComplete to H.323 entity and send RAS DRQ to gatekeeper if it admitted the call.

A.2.8 Gateway Receives Q.931 ReleaseComplete

If the gateway receives a Q.931 ReleaseComplete, the H.323 side of the call is closed. The gateway sends a BYE to the SIP entity if the call has been established.

A.2.9 Gateway Receives RAS DRQ

If the call is active, close it. Send RAS DCF (disengage confirm) to H.323 entity.

A.2.10 Gateway Receives RAS URQ

If the gateway receives a RAS URQ (unregister request) message, the behavior depends on whether the gateway also acts as a gatekeeper. If the gateway also contains a gatekeeper, unregister the endpoint as specified by RAS. otherwise the request must have come from a gatekeeper. Close all the associated calls on both SIP and H.323 sides and send a RAS UCF (unregister confirm) to the H.323 entity.

B H.323 Call Without Fast-Connect

Message flow for normal call connect in H.323 between two terminals registered with different gatekeepers is shown in Fig. 13.

C Acknowledgments

We would like to thank Chris Kang, Jonathan Lennox and Gautam Nair for their help in implementation and general discussions. The work described here was supported by Sylantro.

D Authors' Addresses

Kundan Singh
Dept. of Computer Science
Columbia University
1214 Amsterdam Avenue, MC 0401
New York, NY 10027
USA
electronic mail: kns10@cs.columbia.edu

Henning Schulzrinne
Dept. of Computer Science
Columbia University
1214 Amsterdam Avenue, MC 0401
New York, NY 10027
USA
electronic mail: schulzrinne@cs.columbia.edu

```
H.323 Terminal 1                         H.323 Terminal 2
128.59.21.152       GK1           GK2          128.59.19.194
|                    |             |                       |
|~~~~~ GRQ ~~~~~~~~~>|             |<~~~~~~~ GRQ ~~~~~~~~~~| (Gatekeeper
|<~~~~ GCF ~~~~~~~~~~|             |~~~~~~~~ GCF ~~~~~~~~~>|  Discovery)
|                    |             |                       |
|~~~~~ RRQ ~~~~~~~~~>|             |<~~~~~~~ RRQ ~~~~~~~~~~| (Registration)
|<~~~~ RCF ~~~~~~~~~~|             |~~~~~~~~ RCF ~~~~~~~~~>|
|                    |             |                       |
|~~~~~ ARQ ~~~~~~~~~>|             |                       | (Admission)
|                    |~~~~ LRQ ~~~>|                       |
|                    |<~~~ LCF ~~~~|                       |
|<~~~~ ACF ~~~~~~~~~~|             |                       |
|                                                          |
|--------------------- Setup ----------------------------->| (Q.931 setup)
|                                                          |
|<-------------------- Call Proceeding --------------------|
|                                                          |
|                                  |<~~~~~~~ ARQ ~~~~~~~~~~|
|                                  |~~~~~~~~ ACF ~~~~~~~~~>| (Admission)
|                                                          |
|<-------------------- Alerting ---------------------------| (Ringing)
|                                                          |
|<-------------------- Connect ----------------------------| (Q.931 successful)
|                                                          |
|-+--+--+--+--+--+--+- Terminal Capability Set -+--+--+--->| (H.245/CESE)
|<---+--+--+--+--+--+- Terminal Capability Set Ack +--+----|
|                                                          |
|<---+--+--+--+--+--+- Terminal Capability Set -+--+--+----|
|-+--+--+--+--+--+--+- Terminal Capability Set Ack +--+--->|
|                                                          |
|-+--+--+--+--+--+--+- Master Slave Determination -+--+--->| (H.245/MSDSE)
|<---+--+--+--+ Master Slave Determination Ack +---+--+----|
|                                                          |
|-+--+--+--+--+--+--+- Round Trip Delay --+--+--+--+--+--->| (H.245/RTDSE)
|<---+--+--+--+--+--+- Round Trip Delay Ack -+--+--+--+----|
|                                                          |
|<---+--+--+--+--+--+- Round Trip Delay --+--+--+--+--+----|
|-+--+--+--+--+--+--+- Round Trip Delay Ack -+--+--+--+--->|
|                                                          |
|-+--+--+--+--+--+--+- Open Logical Channel -+--+--+--+--->| (H.245/LCSE)
|<---+--+--+--+--+--+- Open Logical Channel Ack +--+--+----|
|                                                          |
|<---+--+--+--+--+--+- Open Logical Channel -+--+--+--+----|
|-+--+--+--+--+--+--+- Open Logical Channel Ack +--+--+--->|
|                                                          |
|<---+--+--+--+--+--+- EndSessionCommand -+--+--+--+--+----| (Terminating)
|-+--+--+--+--+--+--+- EndSessionCommand -+--+--+--+--+--->|
|                                                          |
|--------------------- Release Complete ------------------>| (Q.931 closed)
|                                                          |
|~~~~~ DRQ ~~~~~~~~~>|             |<~~~~~~~ DRQ ~~~~~~~~~~| (RAS Disengage)
|<~~~~ DCF ~~~~~~~~~~|             |~~~~~~~~ DCF ~~~~~~~~~>|
```

Figure 13: H.323 Call Without Fast-connect

References

[1] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, "SIP: session initiation protocol," Request for Comments (Proposed Standard) 2543, Internet Engineering Task Force, Mar. 1999.

[2] M. Handley and V. Jacobson, "SDP: session description protocol," Request for Comments (Proposed Standard) 2327, Internet Engineering Task Force, Apr. 1998.

[3] International Telecommunication Union, "Packet based multimedia communication systems," Recommendation H.323, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Feb. 1998.

[4] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," Request for Comments (Proposed Standard) 1889, Internet Engineering Task Force, Jan. 1996.

[5] International Telecommunication Union, "Digital subscriber signalling system no. 1 (dss 1) - isdn user-network interface layer 3 specification for basic call control," Recommendation Q.931, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Mar. 1993.

[6] International Telecommunication Union, "Control protocol for multimedia communication," Recommendation H.245, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Feb. 1998.

[7] S. Bradner, "Key words for use in RFCs to indicate requirement levels," Request for Comments (Best Current Practice) 2119, Internet Engineering Task Force, Mar. 1997.

[8] International Telecommunication Union, "Media stream packetization and synchronization on non-guaranteed quality of service LANs," Recommendation H.225.0, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Nov. 1996.

Full Copyright Statement