Signaling Protocols for Internet Telephony

Architectures based on H.323 and SIP

Nicklas Beijar
Helsinki University of Technology
Laboratory of Telecommunications Technology
Otakaari 5 A, 02150 ESPOO
Nicklas.Beijar@hut.fi

Contents

Printed: October 20, 1998

List of Figures

List of Tables

Abbreviations

ASN.1	Abstract Syntax Notation One
BMA	Brokered Multicast Advertisements
CIF	Common Intermediate Format
Codec	Coder Decoder
CPL	Call Processing Language
DTMF	Dual-tone Multifrequency
ETSI	European Telecommunications Standards Institute
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IANA	Internet Assigned Numbers Authority

IETF	Internet Engineering Task Force
IP	Internet Protocol
IPSec	IP Security Protocol
ISAKMP	Internet Security Association and Key Management Protocol
ISP	Internet Service Provider
ITSP	Internet Telephony Service Provider
ITU	International Telecommunications Union
LAN	Local Area Network
Mbone	Multicast Backbone
MC	Multipoint Controller
MCU	Multipoint Control Unit
MIME	Multipurpose Internet Mail Extensions
MP	Multipoint Processor
NIC	Network Interface Card
PCT	Private Communication Technology
PDU	Protocol Data Unit
PER	Packet Encoding Rules
PGP	Pretty Good Privacy
POTS	Plain Old Telephone System
PRI	Primary Rate Interface
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RAS	Registration, Admission, Status
RSVP	Resource Reservation Protocol
RTP	Real-Time Protocol
RTCP	Real-Time Control Protocol
RTSP	Real-Time Streaming Protocol
SAP	Session Announcement Protocol
SCP	Session Control Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SMTP	Simple Mail Transfer Protocol
SS7	Signaling System 7
SSL	Secure Socket Layer
TCP	Transport Control Protocol
TLS	Transport Layer Security
TSAP	Transport Service Access Point
UDP	User Datagram Protocol
URL	Universal Resource Locator
WAN	Wide Area Network

1. Abstract

The aim of this report is to give an overview of the signaling protocols for Internet telephony and multimedia conferencing. Currently two protocol suites exist: the H.32x series designed by the International Telecommunication Union and the Session Initiation Protocol (SIP) by Internet Engineering Task Force. H.323 already has an established position while SIP is a new protocol trying to get into the market. Contrary to many other approaches, we will try to give an overview over both protocols. We will present how the protocols have chosen to solve different problems of signaling. Some issues that are addressed by both protocols will be compared. In this report, we will mainly concentrate on wide area telephony in the Internet. The term telephony also refers to multimedia communication involving video and data. We will make the conclusion that H.323 is best suited for telephony in private intranets and SIP in the public Internet, though the differences are small and both protocols can be used in both scenarios. The discussed protocols are still immature and development is in progress.

2. Introduction

2.1 Signaling protocols

Signaling protocols are used to establish and control multimedia sessions or calls. These sessions include multimedia conferences, telephony, distance learning and similar applications. The IP signaling protocols are used to connect software and hardware based clients through a local area network (LAN) or the Internet.

The main functions of call establishment and control are: user location lookup, name and address translation, connection set up, feature negotiation, feature change, call termination and call participant management such as invitation of more participants. A number of additional services, such as security, billing, session announcement and directory services, can also be included in the protocols. Signaling is closely related to the transmitted data streams, but data transmission is not a part of the signaling protocols.

2.2 H.323 and SIP

Currently two standardized protocols exist on the market: H.323 and SIP. These two protocols represent different approaches of the same problem: signaling and control of multimedia conferences.

H.323 is an umbrella standard from the *International Telecommunications Union* (ITU) for multimedia communications over local area networks (LANs) that do not provide a guaranteed quality of service (QoS). H.323 is a part of a larger series of communication standards called the H.32x series, for multimedia conferencing over different types of networks, including ISDN and PSTN. The H.323 specification was approved in 1996, but the first standards of the H.32x series were approved as early as 1990. Version 2 of the standard addresses conferencing over wide area networks (WAN) and was approved in January 1998. The recommendation represents a traditional circuit-switched approach to IP telephony, based on the ISDN signaling protocol Q.931.

The *Session Initiation Protocol* (SIP) is developed by the *Multiparty Multimedia Session Control* (MMUSIC) working group of the *Internet Engineering Task Force* (IETF). This protocol is still under development and is not as well known as H.323. SIP is a more lightweight protocol based on HTML. It was originally designed for multimedia conferencing on the Internet. In addition to SIP, we will also consider the other signaling protocols by the IETF as parts of the SIP architecture. These include the *Session Description Protocol* (SDP) and the *Session Announcement Protocol* (SAP).

3. Overview of the protocols

3.1 Network architecture

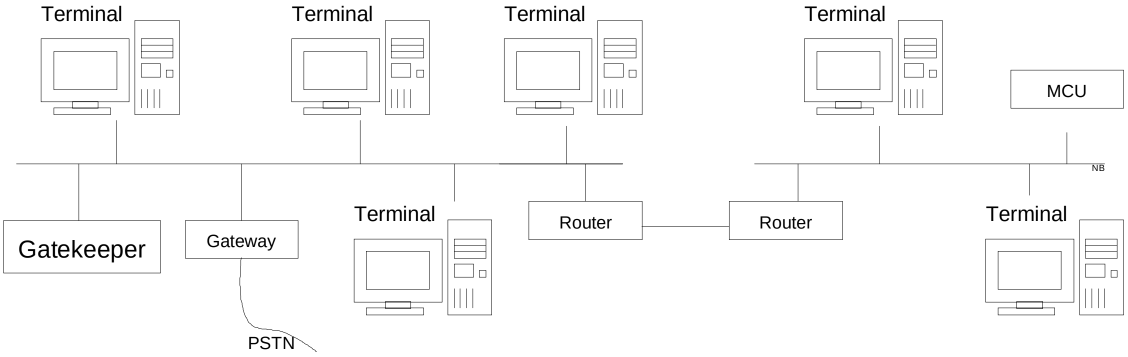

Figure 1. H.323 zone

Though different looking definitions, H.323 and SIP have a basically similar network architecture. In H.323 four types of entities are defined: terminals, gatekeepers, gateways and multipoint control units (MCU). SIP defines three types of entities: terminals (user agent servers), proxy servers and redirect servers. Here gateways are considered as special cases of user agents. The basic configuration in both H.323 and SIP consists of at least two terminals connected to a local area network. However, in practical applications it is necessary to add some of the other entities in order to get an efficient communications system that is connected to the outside world. Each entity brings some more functionality to the network.

Terminals or ***user agents*** are the client endpoints able to receive and place calls. The endpoints generate and receive bi-directional real-time information streams. A terminal can either be software run in a personal computer or a dedicated hardware appliance. In both H.323 and SIP, a terminal must support voice communications, whereas video and data are optional.

Gatekeepers in H.323 and ***servers*** in SIP have a similar function. Calls are set up through the gatekeeper or server, which provides address resolution and routing services. They also provide security services and means to control the traffic on the network through admission control and bandwidth management.

A H.323 network is divided into *zones* (Figure 1). A zone is a collection of all terminals, gateways and multipoint control units that are managed by a single gatekeeper. The gatekeeper acts as a central point of its zone, and provides call control function to the endpoints. Calls within a zone are managed by the gatekeeper. Calls between zones may involve several gatekeepers.

Gatekeepers are responsible for address translation, admissions control, bandwidth control and zone management. Address translation in this case means translation of alias addresses to transport addresses using a translation table. Admissions control is used to determine whether an endpoint is allowed to terminate or originate a call. It may be based on authorization, bandwidth or some other criteria. Bandwidth control lets a given amount of bandwidth be reserved for H.323 traffic and distributed between the connections. When the limit is reached no more connections can be opened, leaving capacity for other protocols. Terminals within a zone register to their gatekeeper, which adds the corresponding address to the registration table.

The optional functions of the gatekeeper include call control signaling, call authorization, bandwidth management and call management. Call control signaling means that the gatekeeper may process the control signals (Q.931) in point to point conferences instead of passing them directly between the endpoint. Call authorization allows the gatekeeper to reject a call depending on its properties. The reasons for rejection can be user defined, for instance restricted access from/to particular terminals or gateways, or restricted access during certain periods of time. Bandwidth management is closely related to bandwidth control, allowing the gatekeeper to reject calls from a terminal if the available free bandwidth is low. In call management the gatekeeper keeps a list of ongoing calls to indicate that a terminal is busy or to provide information for the bandwidth management function. The H.323 gatekeeper functions are resumed in Table 1.

Table 1. H.323 gatekeeper functions

Address translation	Required
Admissions control	Required
Call control signaling	Optional
Call authorization	Optional
Bandwidth control	Required
Bandwidth management	Optional
Zone management	Required
Call management	Optional

Call routing is an optional feature of the H.323 gatekeeper. With this feature the calls may be controlled more effectively and service providers can bill for calls placed through their network. The routing service may also be used to redirect calls to other endpoints if a called terminal is unavailable. In addition, the gatekeepers can balance the load among multiple gatekeepers based on some routing logic. The gatekeeper acts like an interface to other H.323 networks, possibly owned by different Internet service providers. Though the gatekeepers are important and highly recommended components, they are not necessary in a H.323 network. However, if a gatekeeper is present, the terminals must use the services it provides.

Calls can be signaled in different ways. In the *direct signaling model*, the gatekeepers participate in call admission but has limited knowledge of the connected calls. A large number of simultaneous calls can be processed, due to limited involvement. The service management functions and call detail recording functions are limited. Both call signaling (Q.931) and call control (H.245) messages are sent directly between the endpoints. In the *gatekeeper-routed call signaling model*, call control messages are still sent directly between the endpoints but call signaling is routed through the gatekeeper. The gatekeeper is more loaded, as it is must process Q.931 signaling. In this model, the gatekeeper is aware of the connection state but not the media usage. In the *gatekeeper-routed model*, only the media streams are sent directly between the endpoints. All signaling information is processed by the gatekeeper, which has to handle all calls. The gatekeeper is aware of connection state and media usage and can perform service management functions, such as connection and media usage statistics.

Some of the functionality of the H.323 gatekeeper can be found in the SIP server. However, the SIP server is not as important as the gatekeeper is. The call does not have to go through any servers, and there is no equivalent to the zone concept. Servers are mainly used to route and redirect calls. Some simple authentication functions can be implemented in the servers, but the best-suited places for these are in the endpoints or in the firewalls.

A SIP server can operate in either proxy or redirect mode, depending on how the next hop server is connected if the user is not located on the contacted server. A redirect server informs the caller to contact another server directly. A proxy server contacts one or more next hop servers itself and passes the call request further. The proxy server has to maintain a call state whereas a redirect server can forget the call request after it has been processed. A user agent or a server does not have to know whether it is communicating with another server or a user agent. It is recommended that servers should be able to operate in both modes. For example, they could proxy calls to local domain user and redirect calls to other users.

Gateways connect Internet telephony to other types of networks. A gateway can for instance connect a SIP network with the public switched telephone network (PSTN) or a H.323 terminal with H.320-compliant terminals over the ISDN network. The gateway translates between different transmission formats, for example H.225.0 to H.221, and between communications procedures, for example H.245 and H.242. It also translates between audio and video codecs and sets up and clears calls on both sides.

In H.323, the terminals communicate with the gateways using the H.245 and Q.931 protocols. In SIP the gateways are implemented as user agent servers that receive and establish calls on either side and translate the streams and control information.

An additional entity is defined in H.323: the ***Multipoint Control Unit*** (MCU). This optional unit is needed in centralized and hybrid multipoint conferences for distribution of media streams. A MCU consists of a *Multipoint Controller* (MC) and optionally a number of *Multipoint Processors* (MP). The MC handles control information and the MPs handles the streams. Terminals send their streams to the MCU, which mixes and redistributes streams back to the terminals. In decentralized multipoint conferences, no MCU is needed.

In the practical implementation, several entities can be combined into the same physical unit, both in SIP and in H.323 architectures. Vendors might incorporate gatekeeper functionality into the gateways and MCUs. MCUs can also be implemented into the endpoints, in order to allow multipoint conferences without any separate MCU unit. In SIP the same servers can operate as redirect and proxy servers, depending on the type of incoming call request.

3.2 Protocol architecture

3.2.1 H.323

The H.32x series consists of different protocols for different network types. H.320 is designed for narrowband switched digital ISDN, H.321 for broadband ISDN and ATM, H.322 for guaranteed bandwidth packet switched networks, H.323 for non-guaranteed bandwidth packet switched networks and finally H.324 for the analog phone system (POTS). Every protocol supports a set of video and audio codecs, depending on the bandwidth of the network and the approval date of the standard. Also different multiplexing, control and multipoint methods are used in the different standards [14]. In this report we will mainly concentrate on non-guaranteed bandwidth IP networks and thus on the H.323 protocol suite.

H.323 uses both reliable and unreliable communications, and both types are required to be provided by the network. Control signals (the H.245 Control Channel and the Call Signaling Channel) and data (the T.120 Data Channel) requires reliable transport and uses a connection oriented mode. In the IP stack this is accomplished with TCP. Unreliable transport, accomplished with UDP, is used for the audio, video and the RAS channel [14]. H.323 is independent of the underlying network topology. Terminals can communicate through hubs, routers, bridges and dial-up connections [19].

H.323 is an umbrella standard consisting of several ITU-T recommendations. The structure of a H.323 terminal can be seen in Figure 2. Recommendation H.323 defines the system, control procedures, media descriptions and call signaling. In the lowest layer, *H.225.0* defines media packetization, stream synchronization, control message packetization and control message formats. It handles the call control to initiate and terminate calls between terminals, gateways and other entities. H.225.0 performs packetization of the media streams using RTP (Real Time Protocol). In the next layer, *H.245* defines negotiation of channel usage and capabilities exchange. It is used for opening and closing of channels for audio, video, data, camera mode requests. It also handles mode changes, flow control and general commands and indications. Another system control component, *Q.931,* defines call signaling and call setup. The *RAS* (Registration, Admission, Status) component is used for communication with gatekeepers. As an optional feature, data channels can be supported with the T.120 series of recommendations. T.120 is used for data applications, for example text messages or a shared whiteboard. Also the audio codecs (G.711, G.722, H.728, H.723.1, G.729) and the video codecs (H.261, H.263) are included in recommendation H.323 [19, 14].

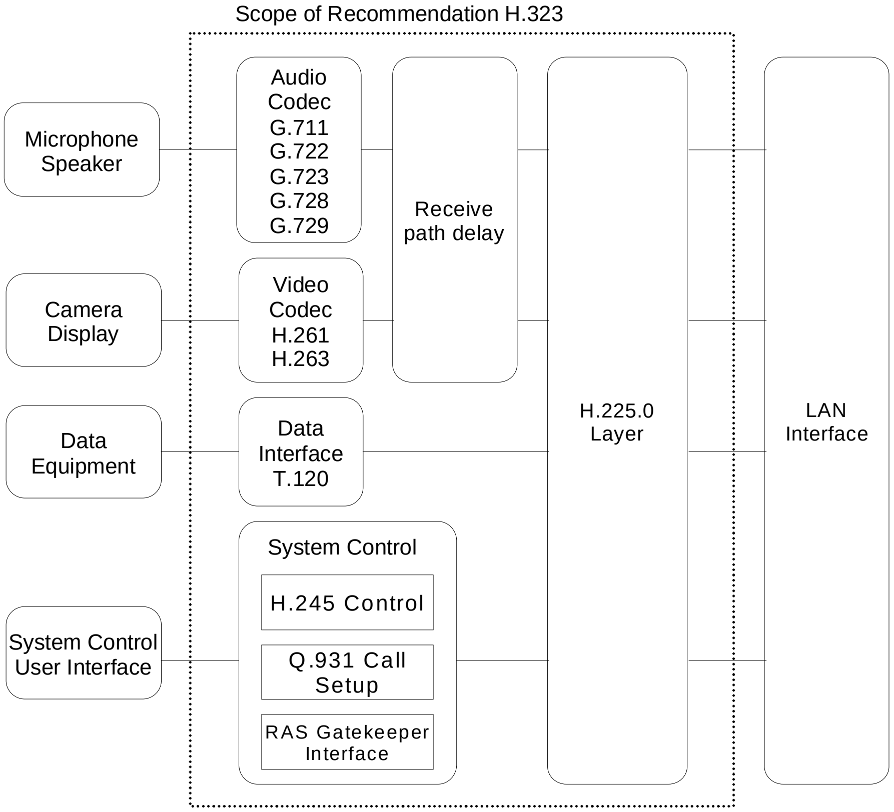

Figure 2. Structure of a H.323 terminal

The control functions in H.323 are provided by three separate signaling channels: the *H.245 control channel*, the *Q.931 call signaling channel* and the *RAS channel*. The H.245 control channel is a reliable channel that carries control messages governing operation of the H.323 entity, including capability exchange, opening and closing of logical channels, preference requests, flow control messages and general commands and indications. Each call is equipped with one H.245 control channel. The call signaling channel is used to establish calls using the Q.931 protocol. RAS is only used in communication with gatekeepers. It performs registration, admission, bandwidth changes, status and disengage procedures between the endpoints and the gatekeepers.

The network interface in a TCP/IP stack is illustrated in Figure 3. Real-time transport is based on the Real-Time Protocol (RTP), which is controlled by the Real-Time Control Protocol (RTCP). Reservation of network resources is handled by the Resource Reservation Protocol (RSVP). These are not a part of the H.323 standard.

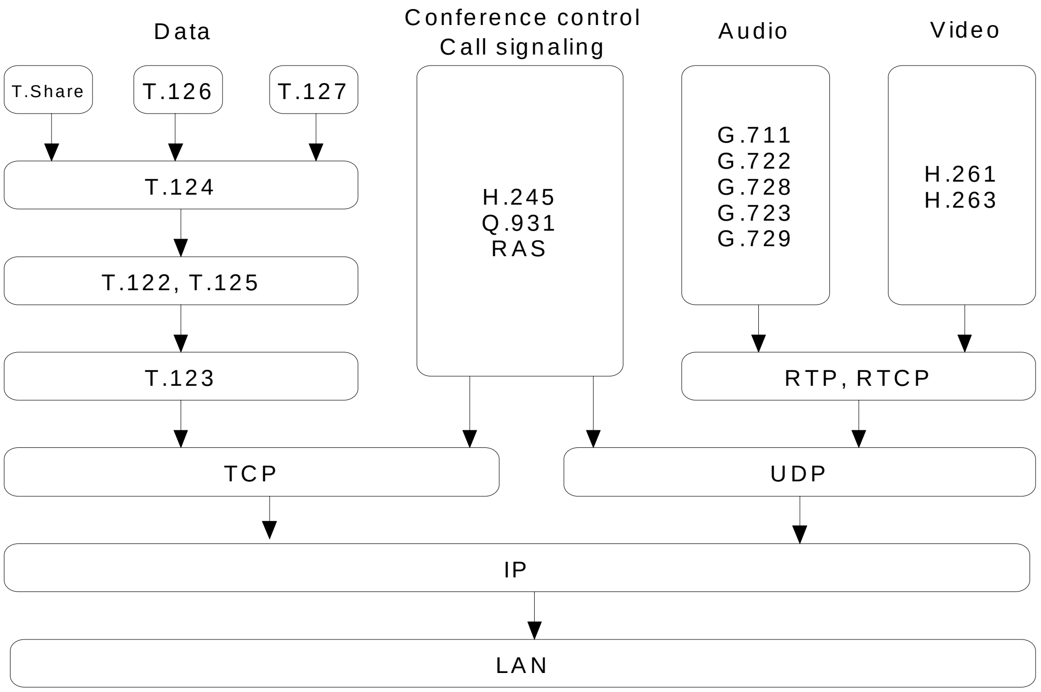

Figure 3. H.323 Protocol architecture in a TCP/IP stack

3.2.2 SIP

SIP is rather independent of the environment and can be used with several transport protocols. In fact any datagram or stream protocol that delivers a whole SIP request or response in full can be used. Such protocols are UDP and TCP in the Internet and X.25, ATM AAL5, CLNP, TP4, IPX or PP elsewhere [12].

Contrary to H.323, SIP does not require any reliable transport protocol and simple clients can be implemented using only UDP transport. However, it is recommended that servers should support both UDP and TCP. A TCP connection is opened only if a UDP connection cannot be established [3]. Reliable transport is achieved by retransmitting requests every ½ second until a response is returned. The system works much like a three-way handshake. The use of application layer reliability has the advantage that the timers can be adjusted according to the requirements. Standard TCP has too long retransmit delays if a packet is lost. The other features of TCP, such as sequence numbers, flow- and congestion control, are not needed. UDP also allows multicast. Further, session not tied to any TCP connection allows participants to reboot, as long as the call identifiers are maintained. However, TCP must be used in some cases, for example with some firewalls and transport layer security protocols, such as TLS [12].

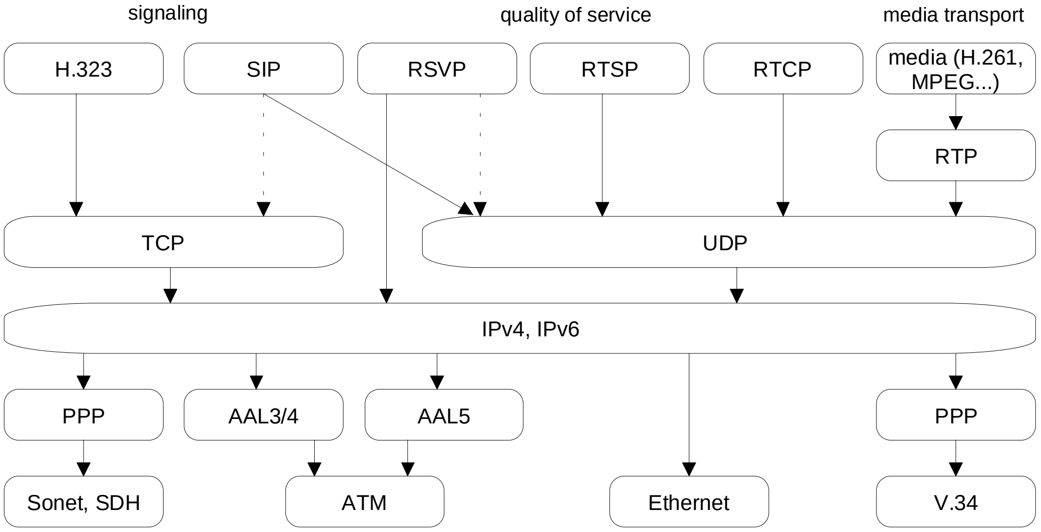

Figure 4. Protocol architecture in SIP

The functionality of SIP is concentrated to signaling, compared to H.323 where the protocol includes all the required functions of conferencing. The SIP protocol includes basic call signaling, user location, registration and as an extension also advanced signaling. The other services, such as quality of service, directory access, service discovery, session content description and conference control, are orthogonal and reside in separate protocols. SIP has a modular architecture, where different functions are performed in different protocols. Protocols can easily be replaced, and even components of H.323 can be integrated into the SIP environment.

SIP uses the *Session Description Protocol* (SDP) to describe the capabilities and media types supported by the terminals. SDP is, like SIP, a text based protocol developed by IETF. SDP messages lists the features that must be implemented in the endpoints [18]. SDP messages are mainly sent within SIP messages, but can also be sent in other ways.

Sessions can also be announced to a larger group of people using another IETF protocol, the *Session Announcement Protocol* (SAP). It is primarily used for announcing large public conferences and broadcast streams like Internet television and radio. However, also in this case the SIP protocol can be utilized as well, thanks to the multicast signaling feature.

3.2.3 Transport of real-time streams

Both H.323 and SIP are used on packet networks that do not provide guaranteed quality of service (QoS). Telephony and videoconferencing are real-time applications that require low delay and delay variation, and small packets with small packet overhead. In addition, real-time data requires handling of timing and synchronization. SIP and H.323 uses the same protocols for transport of real-time streams on packet networks.

In the Internet real-time data is transported with the *Real-Time Protocol* (RTP) developed by the Internet Engineering Task Force (IETF). The protocol allows any real-time data to be transported, but is mainly used for the audio and video streams. RTP uses an unreliable protocol, UDP, that does not re-transmit lost packets and has less overhead. It adds a header containing a timestamp and sequence number to the UDP packets. With RTP and latency buffering at the receiving endpoint, the timing, packet ordering, synchronization of multiple streams, duplicate packet elimination and continuity of the streams are handled. Furthermore, RTP-based protocols can operate on the Internet's Multicast Backbone (*Mbone*), which provides a multicast facility and supports video, voice and data conferencing.

RTP is controlled by the *Real-Time Control Protocol* (RTCP). RTCP collects information about the session participants and the quality of service, and redistributes the information to the endpoints. It also provides minimal control and identification functionality. Multicast conferences with multiple audio and video streams are handled by the *IP Multicast* protocol. The protocol works in a layer under RTP.

Another IETF protocol, the *Resource Reservation Protocol* (RSVP) allows an endpoint to request a specified amount of bandwidth for a media stream. RSVP reserves resources within the routers along the path, for example buffers, queues and interfaces. The endpoint receives a reply indicating whether the request has been granted. RSVP can be considered as a component of the future "integrated services" Internet, which provides both best-effort and real-time qualities of service.

3.3 Message structure

3.3.1 H.323

Most of the control messages in H.323 are encoded in the *Abstract Syntax Notation One* (ASN.1) scheme using the *Packet Encoding Rules* (PER). ASN.1 is a complex encoding scheme where data is put into hierarchical structures. Structures can be optional, variable length and nested. The messages are compiled into a binary format. ASN.1 requires a parser both for the descriptions and for the binary format. ASN.1 extensions are backward compatible by upgrading the central description. However, this requires that the upgrades are coordinated to avoid incompatibility [2, 12].

The message types are commands, responses and indications. In addition to the standard fields, there are also places for nonstandard parameters and identifiers. An example ASN.1 structure is shown in Figure 5.

```
Capability ::=CHOICE
{
   nonStandard                        NonStandardParameter,
   receiveVideoCapability             VideoCapability,
   transmitVideoCapability            VideoCapability,
   receiveAndTransmitVideoCapability  VideoCapability,
   receiveAudioCapability             AudioCapability,
   transmitAudioCapability            AudioCapability,
   receiveAndTransmitAudioCapability  AudioCapability,
   receiveDataApplicationCapability   DataApplicationCapability,
   transmitDataApplicationCapability  DataApplicationCapability,
   receiveAndTransmitDataApplicationCapability DataApplicationCapability,
   h233EncryptionTransmitCapability   BOOLEAN,
   h233EncryptionReceiveCapability    SEQUENCE
   {
                                      h233IVResponseTime INTEGER (0..255),
                                      -- units milliseconds
   },
   ...,
   conferenceCapability               ConferenceCapability
}
```

Figure 5. The Capability structure of the H.245 specification [2]

3.3.2 SIP

SIP is a client-server protocol, with requests sent from the client and responses returned by the server. A single call may involve several clients and servers.

The SIP protocol reuses the message structures found in HTML. The messages are in text format using ISO 10646 in UTF-8 encoding. As in HTML, the client requests invoke *methods* on the server. The messages consist of a start-line specifying the method and the protocol, a number of header fields specifying the call properties and service information, and an optional message body. The body can contain a session description. The following methods are used in SIP:

- *Invite* invites a user to a conference
- *Bye* terminates a connection between two users
- *Options* signals information about capabilities
- *Status* informs the server about the progress of signaling
- *Ack* is used for reliable message exchanges
- *Register* conveys location information to a SIP server [12]

The syntax of the response codes is similar to HTML. The three digit codes are hierarchically organized, with the first digit representing the result class and the other two digits providing additional information. The first digit controls the protocol operation and the other two gives useful but not critical information. A textual description and even a whole HTML document can be attached to the result message.

3.3.3 Call setup in H.323

A conference in H.323 consists of the following five phases: call set-up, initial communication and capability exchange, establishment of audiovisual communication, call services and call termination [1].

Call setup takes place using the H.225.0 protocol. The client sends a *set-up* message to the called endpoint. The endpoint responds with a *call proceeding* message when the call request has been received and then an *alerting* message when the user is informed. If the call goes through a gateway, the gateway sends the alerting message when the ring tone is heard. A *connect* message will be sent when the user answers. If a gatekeeper is used, the endpoints have to register with it before the actual call setup is performed. We get different cases depending on which endpoint, if any, has a gatekeeper: the calling terminal, the called terminal, both to different gatekeepers or both terminals to the same gatekeeper. The gatekeeper registration consists of a *ARQ request* to the gatekeeper, and either an *ACF confirm* or an *ARJ reject* message in response. The registration procedure uses the RAS signaling channel. RAS messages are also used in gatekeeper-to-gatekeeper conversations.

After the call has been accepted by the callee, the media channels are set up using the H.245 protocol. The feature negotiation is performed. This requires several round-trips and results in a significant delay before the negotiation is completed. In version 2 of the standard, this delay problem is solved. Using the *Fast Connect* procedure the media channels can be established with only one round-trip and the media transmission can start immediately after sending a fast start message. Certain billing procedures require the media channels to be operational before the connect message is sent, thus fast connect is a requirement.

The H.245 channel is set up by sending *terminal capability set* messages. After the capability exchange, the endpoints perform a master-slave determination to determine which of the terminals is the active MC. Also the master-slave determination is a part of the H.245 specification. To conserve resources, synchronize call signaling and control and reduce call setup time and alternative method exists: The H.245 messages can be sent within the Q.931 channel, a process known as “tunneling”. To conclude, there are three ways to set up a H.323 call: a separate H.245 channel, fast connect and tunneling. However, it is not always possible to use fast connect or tunneling and a terminal can at any time switch back to a separate H.245 channel.

3.3.4 Call setup in SIP

A successful call setup consists of an *INVITE* request from the client and an *ACK* reply from the called endpoint. A negative response can be sent with a *BYE* reply. The invite request usually contains a session description written in the *Session Description Protocol* (SDP) format. The session description provides information about which features and media formats are supported by the client.

```
INVITE sip:henning@cs.columnia.edu SIP/2.0
From: N. Beijar <nbeijar@keskus.hut.fi>
To: H. Schulzrinne <henning@cs.columnia.edu>
Call-ID: 19980622@lion.cs
Subject: SIP Invitation
Content-Type: application/sdp
Content-Length: 67

v=0
o=bell 76525365 41540546 IN IP4 128.3.45.67
c=IN IP4 135.180.144.94
m=audio 3456 RTP/AVP 0 3 4 5
```

Figure 6. Example SIP invitation message with a SDP body

The invitation may pass through several servers on the way to the callee. There are three types of servers: *proxy*, *redirect* and *user agent servers*. The proxy server receives the request and forwards it towards the location of the callee. It can also forward the request to multiple servers at once, in the hopes of contacting the user at one of the locations, or to multicast groups. A *Via* header traces the path of the request, allowing responses to find their way back and helps to detect loops. A redirect server only informs the caller about the next hop, and the caller sends a new request to the suggested receiver directly. The user agent server resides on the host where the user is situated. It informs the user about the call and waits for a respond what to do: accept, reject or forward.

A SIP system may also include *location servers*, keeping a database of the locations of the users. This will let the user move between a number of different end systems over time. A location server may use *finger*, *rwhois*, *LDAP* or any other protocol to determine the end system where the user can be reached. The location server sends *REGISTER* messages to the servers to inform about changes.

When the user has been contacted, a response consisting of a response code and message is sent back to the caller. The codes are given in a manner similar to HTTP. Response codes 1xx indicate the progress of the call and are always followed by other responses indicating the final result. Codes 2xx indicate successful results, 3xx indicate redirection, 4xx, 5xx and 6xx indicate client, server and global failures, respectively. Responses are always sent to the entity that sent the message to the server, not to the originator of the request. In this way the responses find their way back through firewalls. The message is repeated regularly until the destination acknowledges with an *ACK* message. A positive response to a setup message also contains a session description, describing the supported media types. Call identifiers are used to indicate messages belonging to the same conference.

Calls have the following properties:

- *logical call source* contained in the *From* field is the originator of the call, that is the entity that requests the call.
- *logical call destination* contained in the *To* field is the party, whom the originator wishes to contact.
- *media destination* is the destination of the media (audio, video, data) for a particular recipient, which may not be the same as the logical call destination.
- *media capabilities* are currently specified using the Session Description Protocol (SDP), which expresses a list of capabilities for audio and video and indicates the destination addresses of the specific media.
- *call identifier* contained in the *Call-ID* field is a unique identifier created by the creator of the call and used by all the participants.

A part of the properties is contained in specific fields and other are conveyed as a part of the payload of the SIP message [12].

3.3.5 Capability exchange

Capability exchange is a fundamental function in Internet telephony, due to the large range of different terminal types, codecs and implementations available. The calling parties have to agree about what features and media types to use. Also the available bandwidth and the decoding performance of the receiver should be taken into account. Capability exchange is performed in the call setup, but can also be needed during the call, for example due to changes in the available bandwidth. Further capability exchange is needed when more terminals enter the conference.

The capability exchange functions of H.323 are based on the H.245 control channel. H.245 provides for separate receive and transmit capabilities as well as for methods to describe these details to other H.323 terminals [14].

SIP uses the Session Description Protocol to describe the capabilities. The capability exchange in SIP is simple: The caller provides a list of its supported features and media types and the callee chooses from the list the ones it supports and sends a SDP message in return.

The capability exchange of H.323 provides a much richer set of functionality than the one used in SIP. Terminals can express their ability to perform various encodings and decodings based on parameters of the codec and depending on which other codecs are in use. SIP has a simpler receiver capability indication that allows a terminal to choose a subset of encodings for a given list of media streams [13]. SIP can, however, use any session description protocol, including H.245.

Media parameters can also be changed during the conference in both protocols. SIP does this by sending a new invitation message with a new media description. In H.323 the bandwidth can be changed during the session. Version 2 allows for changing from one codec to another without the need for two decoders and without any media dropout [20].

3.4 Media streams

3.4.1 Audio codecs

The recommended minimum set of audio codecs for a SIP agent consists of G.711 (μ-law), DVI4 and GSM [18]. In H.323 only the G.711 standard must be supported. Because of its low bandwidth requirements the G.723 standard is also being considered as required, and will be the predominant audio codec in H.323 applications [14]. Support for all other standards are optional in both H.323 and SIP. SIP can use any IANA-registered or privately named codec, while H.245 is currently restricted to only ITU-T codecs.

Table 2. Some standardized audio codecs [19]

Codec	Rate	Audio bandwidth
G.711	48, 56, 64 kbit/s	3 kHz
G.722	48, 56, 64 kbit/s	7 kHz
G.723.1	5.3, 6.3 kbit/s	3 kHz
G.726	16, 24, 32, 40 kbit/s	3 kHz
G.728	16 kbit/s	3 kHz
G.729	8 kbit/s	3 kHz
G.4K	4 kbit/s	
GSM 06.10	13 kbit/s	
RT24	2.4 kbit/s	Voice
RT29	2.9 kbit/s	Voice
Elemedia SX8300P	8.3 kbit/s	
VoxWare AC6…AC96	6…96 kbit/s	3.4…22 kHz
PT724	24 kbit/s	7 kHz

3.4.2 Video codecs

Video capabilities are optional in H.323 and SIP terminals. If a H.323 terminal is video enabled it should, however, support at least the H.261 video codec with image size QCIF [14]. Also SIP video agents should support H.261 with conditional replenishment and with image sizes QCIF and CIF [18].

The used formats are multiples of the *Common Intermediate Format* (CIF) which is a non-interlaced format of size 352 x 288 pixels. Usually a quarter of this size is used (QCIF), which corresponds to the bit rate of one ISDN channel. The H.261 stream contains two types of frames: DCT-based intraframes and predictive interframes.

H.263 is an improved and backwards-compatible update to H.261. H.263 includes the picture formats from H.261 and adds two sizes, as can be seen in Table 3. The picture quality is improved using a half pixel motion-estimation technique, predicted frames and a Huffman coding table optimized for low bit rate transmissions [14].

Table 3. ITU image formats for video conferencing [14, 17]

Picture Format	Image size (pixels)	Max. frame rate	Video source rate	Average coded bit rate	H.261	H.263
sub-QCIF	128 x 96	30 f/s	1.3 Mb/s	26 Kb/s	Optional	required
QCIF	176 x 144	30 f/s	9 Mb/s	64 Kb/s	Required	required
CIF	352 x 288	30 f/s	36 Mb/s	384 Kb/s	Optional	optional
4CIF	702 x 576	30 f/s	438 Mb/s	3-6 Mb/s	Not defined	optional
16CIF	1408 x 1152	30 f/s	2.9 Mb/s	20-60 Mb/s	Not defined	optional

3.4.3 Data channels

Data channels are used in real-time multipoint applications, such as file transfer, shared whiteboard, application sharing, collaborative browsing, virtual reality and multi-player gaming. H.323 supports this kind of data conferencing through the T.120 specification. T.120 is a platform and network independent set of standards, providing a general data sharing interface to application designers. T.120 supports reliable point-to-point and multipoint data transfer, possibly utilizing multicast features of the network. T.120 capabilities are incorporated into the clients and multipoint control units (MCU). The data channels can be mixed and controlled by the MCU [14, 15]. Version 2 of the H.323 specification integrates T.120 into H.323 as a media channel, while it was considered as a separate protocol in the first version.

In SIP the data channels are supported as media streams. Data communication can take place using the T.120 protocol or any other protocol, as indicated in the session description. The standard does not specify which protocol should be used for data channels.

3.5 Multipoint conferences

3.5.1 Multipoint conferences in H.323

In multipoint conferences, with three or more endpoints participating, a *Multipoint Control Unit* (MCU) is required. Multipoint control units can be a part of a H.323 component or exist as a separate component. A MCU consists of a *Multipoint Controller* (MC) and optionally a number of *Multipoint Processors* (MP). The MC handles H.245 negotiations between all terminals to determine common capabilities for audio and video processing. It also controls the conference resources by determining which, if any, of the audio and video streams will be multicast. The MP handles the media streams by mixing, switching and processing the information [14].

Multipoint conferences can be arranged in a centralized or decentralized way. The centralized version requires the existence of a Multipoint Control Unit to facilitate a multipoint conference. The audio, video, data and control streams from each participant are sent to the MCU, where the MC manages the conference using H.245 functions and the MP mixes, switches and distributes the streams. The MP may also perform conversion between different codecs and bit rates. The resulting streams are sent back to the participating endpoints.

The decentralized version can make use of multicast technology. The terminals can send audio and video streams to each other without sending through the MCU. Only the control data is processed by the MCU. The terminals have to decode and mix the multiple incoming audio and video streams. This means that the terminals sum all the received audio streams and selects one or more video streams to display. The number of simultaneous streams a terminal can decode is reported to the MC using the H.245 control channels. The total number of multicast streams is not limited by the capabilities of individual terminals.

It is also possible to combine the models to create hybrid multipoint conferences, where some of the streams are processed through point-to-point messages to the MCU and the remaining streams are transmitted through multicast. As an alternative in the centralized model, the resulting mixed and processed streams from the MCU can be transmitted to the endpoints through multicast, conserving network bandwidth.

In addition to the above methods, conferences can also be broadcast. Broadcast operation is defined in a separate specification named H.332, formerly called “H.Loose multipoint”. The conference may include thousands of participants. These conferences are usually one-to-many conferences, where one source transmits unidirectional streams to several recipients using network multicast capabilities. Another application is a broadcast panel type conference. Here some terminals are discussing in a normal multipoint conference, while a number of listeners are receiving the streams with unidirectional transmission.

3.5.2 Multipoint conferences in SIP

The multi-party conferences in SIP work in a basically similar way to the multipoint conferences in H.323. An exception is the lack of an MCU. In SIP a bridge performs a corresponding function, but it is not required for multipoint conferences as in H.323.

The multipoint conferences in SIP can be grouped into three groups: *multicast conferences, bridged conferences* and *full-mesh conferences*. The full-mesh conferences are decentralized with each participant sending media stream to every other participant and performing the mixing of incoming streams locally. This method is suitable for conferences with three or more endpoints. In a bridged conference each user is connected to a bridge, which mixes the media from all users and transmits the resulting streams back. A more bandwidth efficient method than meshes and bridges is the multicast conference, which can be used on multicast enabled network.

In the practical case these methods are mixed. The terminals that are on the same multicast enabled network use the multicast method. The terminals learn automatically about each other, which ones are multicast capable, by inviting them to a multicast group and then listening to their responses. The terminals that are not on a multicast enabled network will continue to use unicast. New terminals always enter the conference in unicast mode.

A *loosely coupled conference mode* can be used on multicast enabled networks when the number of participants is large. In this mode the new participant does not need to invite itself with all the members of the conference. Thus, it does not know anything about the other members before it eventually learns about them through RTCP or other media-specific membership announcement mechanism.

3.5.3 Multicast signaling

Both SIP and H.323 can utilize multicast capabilities of the network in the transmission of *streams*. However, multicast *signaling* [12] is a technique only supported by SIP. A client can send request via multicast to a group of receivers, to reach all members or to reach any of the members. In the second case the call is established to the first who answers. Conferences can also be advertised with SIP, although a separate protocol (SAP) exists for that purpose. The receiving servers do not respond to advertise invitations. SIP uses an algorithm called *reconsideration* to distribute bandwidth among senders. The sender listens to the group and schedules its transmission based on the number of other senders heard. Replies to a reach-any conference, where the first reply is considered the “winner”, are delayed by a skewed distributed random time. In this way response floods are avoided.

4. A Comparison of H.323 and SIP

4.1 Addressing

Each physical H.323 entity has one network address, which is dependent on the network environment. Entities may have several TSAP identifiers, allowing multiplexing of several channels sharing the same network address. Dynamic TSAP identifiers are used for the H.245 control channel and media channels. The call signaling channels and RAS channels uses well-known TSAP identifiers.

Endpoints may also have one or several alias addresses, representing an endpoint or a conference that an endpoint is hosting. The second version of the standard extends the alias concept, so the alias can consist of E.164 addresses (telephone numbers), email addresses, URLs, transport IDs, party number or text identifiers [20]. A gatekeeper is needed to resolve all aliases, limiting small H.323 networks without gatekeepers to use only host names.

The addressing format chosen in SIP is an email-like identifier in the form “user@host”. The user part is a civil name or a telephone number. The host part is either a domain name or a numeric network address. Email-like names can be mapped by any device on the Internet. In many cases the address is the same as the user’s email address. In addition to individual persons, an address may specify the first available person from a group of individuals or a whole group.

A user a location will be reached after a number of translations, as shown in Figure 7. A single address may lead to different host locations depending on the time of day, media to be used and other factors. The SIP client resolves addresses using DNS SRV, MX or CNAME records. If these fail, a client may contact an SMTP server to obtain an alternate address. A client might even send the session description as email, if all of the above fails.

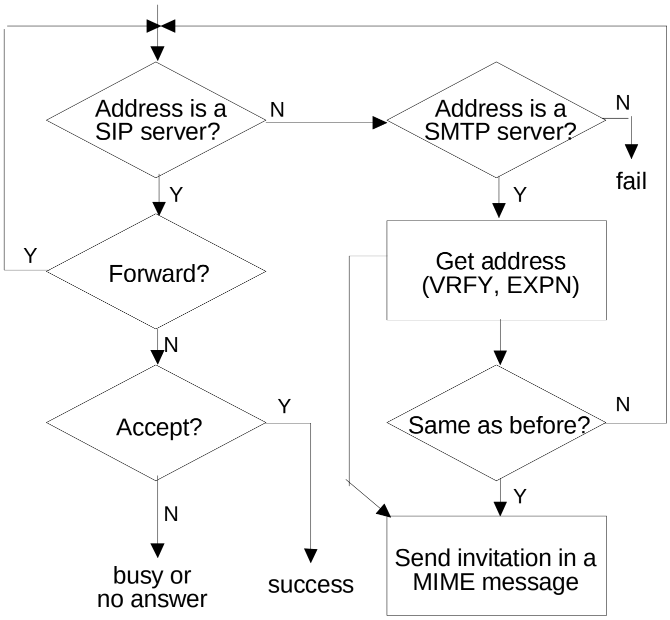

Figure 7. SIP address resolution

SIP and H.323 calls can be established from the web using URLs in the same way as the *mailto* URL. SIP uses the form "sip:name@company.com". H.323, primarily represented by Microsoft NetMeeting, uses the format "callto:machine.company.com" for calls to a given machine or "callto:directoryserver.com/user@company.com" for calls through a directory server. Also calls to PSTN phones can be made from the web. In SIP this could for example be done with a URL in the format "sip:+358-9-4515303" [12, 31].

Work has been done to standardize the URL format for telephone numbers. Internet draft [10] describes URL schemes for telephone calls, telefaxes and data calls. The URL consists of an identifier (tel, fax or modem), an E.164 number and necessary parameters. The fax identifier is the same as in recommendation E.123. The draft represents a more location and equipment independent approach. The URL is independent of the protocol used to establish the call and does not specify any specific gateway. The calling terminal establishes the call in an appropriate way, depending on how it is configured to handle telephone, telefax and data calls.

4.2 Complexity

One of SIP's main advantages is its simple and straightforward operation. H.323 has often been criticized for being too heavy, complex and inflexible. Contrary to the rather complex H.323, SIP is a noticeably simpler protocol, and a basic implementation of a SIP client can be implemented in a short program.

Much of the complexity of H.323 comes from the multiple protocol components it consists of. The components are tightly intertwined and cannot be separately used or exchanged.

Also the message format is a complexity issue. The H.323 protocol is based on ASN.1 and PER (packet encoding rules) and uses a binary representation. Generally, this requires large and expensive code-generators to parse. In SIP the messages are in text format, similarly to HTTP, Real Time Streaming Protocol (RTSP) and most other Internet protocols. Parsers for SIP can be implemented with text processing languages, such as Perl, Tcl and Java. In addition, the similarity to HTTP allows for easy code reuse and SIP parsers can easily be integrated into HTTP parsers and web browsers. In text based protocols debugging and manual entry of messages is easy.

Generally text based protocols are less storage efficient than binary. ASN.1 can be storage efficient, especially if the packet encoding rules (PER) are used. Packet size is not so important in control protocol, but in SIP it is desirable to avoid packet fragmentation, limiting the UDP packet size to 1500 bytes.

The ASN.1 encoding used in H.323 makes firewalls and proxies very complex. The H.323 protocol suite consists of several components almost without clear separation. This also makes encryption difficult. More on this later.

Furthermore, H.323 duplicates functions present in other parts of the protocol. In particular, H.323 uses both RTP and RTCP for feedback and conference control. The RTP mechanisms are engineered for up to thousand-party broadcast conferences, while the H.245 mechanisms are engineered for small to medium conference sizes only, and are therefore redundant.

H.323 provides multiple methods for accomplishing a single task. This contributes to the complexity, but can also be useful in some cases. An example of this is the three distinct ways in which H.245 and H.225.0 can be used together: separate connection, H.245 tunneling through H.225.0 and FastStart in H.323 version 2. All methods have to be supported by all end systems, gatekeepers, gateways and firewalls.

4.3 Call setup delay

H.323 is a complex protocol based on the H.320 standard for video conferencing on ISDN, originally designed for special purpose hardware. A consequence of the complexity and the ISDN background is a rather long setup delay. Call setup requires about 6 to 7 round-trip times, depending on whether a gatekeeper is being used or not. This includes setup of the Q.931 and H.245 connections. In the case of a modem connection, where the transmission delays are substantial, it can take several seconds. A packet loss might cause a larger delay, because TCP waits 6 seconds to retransmit SYN packets [31].

Call setup consists of two stages: first an equivalent of an ISDN circuit (H.225 and Q.931) is set up, then the end points negotiate how to multiplex channels on the circuit. The two-phase setup has some undesired consequences when an H.323 terminal is connected to a plain telephone through an Internet telephony gateway. In the first phase the circuit and thus the phone call is set up. The phone user starts speaking and listening. However, the H.323 user will not be able to speak and listen until the second phase is completed. In the Internet this may take one or two seconds [32]. The call setup is illustrated in Figure 8. Notice that also the setup of TCP connections requires three packets.

A similar problem was found in SIP, where the remote terminal can be alerting but the connection might not be established because of lack of resources, for example in PSTN gateways. It can, however, be solved if the receiving terminal checks and even reserves the resources before ringing and sending a “200 OK” -result.

In version 2 of H.323 the call setup delays are significantly minimized. Fast connect reduces the number of round-trips and allows the media channels to be operational before the connect message is sent and the telephone rings. Further improvement is accomplished by sending partial addresses to the gatekeeper as the user is keying in the address, so the routing process can take place simultaneously [20].

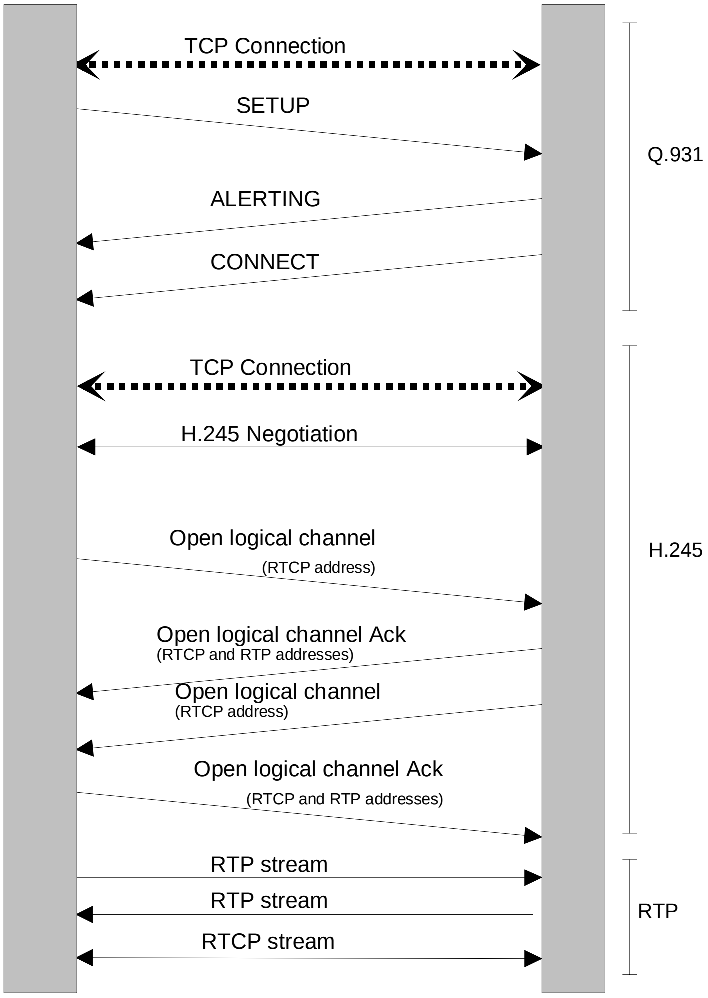

Figure 8. Normal setup of a H.323 call [20].

Setting up a SIP connection using UDP takes four packets and 1½ round trip times. If a server (gatekeeper) is used, the time will consequently increase [31].

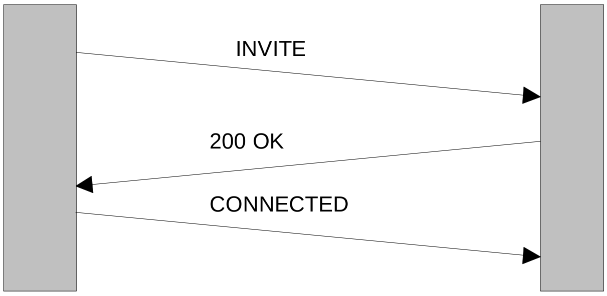

Figure 9. Setup of a SIP call [31].

4.4 Extensibility

Since Internet telephony is still immature and continuously under development, it is likely that additional signaling capabilities will be needed in the future as new applications appear. Vendors may want to add new features and support for their equipment. Also new codecs and video formats are developed. It is important that extensions could be added to the existing protocols. Moreover, the new versions have to be backward compatible. As Internet is an open and evolving network, it could be expected that additions will develop uncoordinatedly. All vendors will develop their own extensions.

H.323 provides *nonstandardParam* fields placed in various locations in the ASN.1 structures. These fields consist of a vendor code and a value that only has meaning to the vendor. However, the extensions are limited to only those places where a non-standard parameter has been added. A vendor cannot add a parameter to places where there is no placeholder for a nonstandard parameter. Currently there is no way for terminals to exchange information about which extensions they support, so the interoperability among terminals from different vendors is restricted to the H.323 standard capabilities.

SIP has taken the same approach as HTTP and SMTP, both of which are widely used and have been developed over a long time by many people. New headers can be added to the SIP messages. By default, unknown headers and values are ignored. Using the *Require* header, the client can require specified headers to be understood by the other endpoint. If this does not support the named services an error message containing the problematic feature is returned and the client can fall back to a simpler operation. Feature names are based on a hierarchical namespace and new features can be centrally registered with IANA. The textual names also mean, that the fields of the headers are self-describing and that different developers can understand and support each others new features easily, in a way that has been common in SMTP. In contrary to H.323, these additions are not specific to a certain vendor only, but common to all developers that chooses to implement the extension. Also the numerical result codes are hierarchically organized, similarly to HTML. The first digit represents the result class and other two digits provide additional information. New result codes can be added, and understood by older clients because of the hierarchy. Only the first digit determines the operation.

New extensions tend to replace older ones, which gradually disappears. However, the terminals have to be backward compatible with older terminals. In H.323 full backward compatibility is needed, which means that the implementations have to support all older features. As various features come and go the size of the implementations will increase. SIP solves the problem through optional headers. Features may be left omitted when they are no longer needed.

4.4.1 Codecs

Hundreds of audio and video codecs have been developed. In H.323 each codec has to be centrally registered and standardized before they can be included in the H.323 applications. Currently, only codecs developed by ITU have codepoints, and there are no free under 28.8 kbit/s codecs available. This is a significant limitation for small software developers. In SIP the codecs supported by an endpoint are listed using the Session Description Protocol (SDP) during connection setup. Codecs are identified by string names, which do not limit the codecs to a range of predefined code values. Also in SIP codecs can be registered (with IANA) to standardize their names and improve compatibility. Free codecs are available.

The H.323 standard specifies the codec requirement very exactly. It requires the terminals to support at least the G.711 voice codec. All H.323 terminals providing video communications shall support H.261 QCIF video encoding [1]. Other codecs and picture formats may be used through H.245 negotiation. If a participant does not understand a codec, the default G.711 will be used. The situation with SIP is quite similar. A SIP client should at least support the G.711 (μ-law), DVI4 and GSM audio codecs [18]. SIP video agents should support H.261. Both protocols have a set of required codecs. This guarantees that two terminals will be able to connect.

4.4.2 Modularity

Both H.323 and SIP are used as a part of a larger entity together with other protocols. They consist of several components. The components perform services like basic signaling, conference control, directory access, quality of service and service discovery. Internet telephony and video conferencing is a continuously developing area, needing new services to be included to the protocols. Especially support for quality of service (QoS) is expected to evolve over time. Therefore, it is an advantage if components are separate and can be swapped in and out. For example, it is more efficient to have a single application independent QoS mechanism than to invent a new QoS protocol for each application.

H.323 is an umbrella standard consisting of recommendation H.225.0 for packet format and synchronization, H.245 for control, H.261 and H.263 video codecs, G.711, G.722, G.728, G.729 and G.723 audio codecs, and the T.120 series of multimedia communications protocols. Together they form a vertically integrated protocol suite. The provided mix of services includes capability exchange, conference control, maintenance operations, basic signaling, quality of service, registration and service discovery. The protocol is split into several components, which is an advantage. However, the services of the components are intertwined in such a way that all components are required and components are difficult to replace.

The SIP protocol includes basic call signaling, user location, registration and as an extension also advanced signaling. The other services, such as quality of service, directory access, service discovery, session content description and conference control, are orthogonal and reside in separate protocols. It is possible to replace individual components. For example, the H.245 capability description can be used in SIP without changes to the protocol. The modularity of SIP also makes it possible to make connections to H.323 compatible terminals. A user can use SIP to locate another user and then use a redirect response to an H.323 URL, indicating that H.323 will be used for the actual communication [13].

4.5 Limitations

4.5.1 Network size

H.323 was originally designed for use on a single local area network, where wide area addressing and user location were not a concern. That version did not scale very well for wide area operation. In the newest version, the concept of zones is defined. A H.323 Zone is a collection of all terminals, gateways and multipoint control units managed by a single gatekeeper [14]. The version also defines procedures for user location across zones for email names. For large number of domains and for complex location operations it still has some scalability problems, for example the lack of loop detection methods.

SIP is engineered for wide area networks. Loop detection is implemented. Firewalls and proxies are easy to configure for SIP connections. The addressing and user location mechanisms are designed for use on the Internet. Thus, the maximum network size is not limited.

4.5.2 Conference size

Both the protocols work basically in the same way. However, in H.323 the multipoint controller (MC) is required for signaling, even in the smallest conferences. This introduces some difficulties. Only one of the users in a conference provides the MC functionality. If this user would leave the conference, the whole conference terminates. Since MC functionality is optional, not all clients include it. This makes even three party conferences impossible in some cases. Furthermore, the MC is a bottleneck for very large conferences.

In version 2 of H.323, these problems are partly solved. The concept of cascaded MC’s is defined, allowing for a limited multicast distribution tree of control messaging. For larger conferences, additional procedures are defined. This results in three different mechanisms for conferences of different sizes.

The conference coordination in SIP is fully distributed and there is no requirement for a MC. SIP scales to conferences in different sizes [13].

4.5.3 Bottlenecks

H.323 and SIP systems used by large, IP telephony backbone providers have to be able to handle a large number of simultaneous calls. This will load both gateways and gatekeepers (or servers in SIP).

H.323 uses a stateful model. This means that gatekeepers must keep call state for the entire duration of a call. Because of TCP based connections the gatekeeper must also hold the TCP connections for the ongoing calls. For large gatekeepers this might be a scalability problem.

In SIP, on the other hand, either a stateful or a stateless model can be used. In the stateless model the server receives a call request, performs some operation, forwards the request and then completely forgets it. SIP messages contain sufficient state to allow the response to be forwarded correctly. Furthermore, both TCP and UDP connections can be used, where the UDP requires no connection state, which will reduce memory requirement and improve scalability.

H.323 version 2 defines the zone concept, which allows for efficient control of network resources. The bandwidth management functions can be used to limit the traffic and the risk for overloaded gatekeepers is reduced.

4.6 Supplementary services

Both protocols offer roughly equivalent services and new services are still added. The table shows the services available in May 1998.

Table 4. Supplementary services [13, 14]

Service	H.323	SIP
Blind transfer	Yes	Yes
Operator assisted transfer	No	Yes
Unconditional call forwarding	Yes (Ver. 2)	Yes
Call forwarding on busy	Yes (Ver. 2)	Yes
Call forwarding on no reply	Yes (Ver. 2)	Yes
Call deflection	Yes (Ver. 2)	?
Hold	Not yet	Yes, through SDP
Multicast conferences	Yes	Yes
Multi-unicast conferences	Yes	Yes
Bridged conferences	Yes	Yes
Forward	Yes	Yes
Call park	No	Yes
Directed call pickup	No	Yes

4.6.1 Mobility and redirection

Personal mobility based on redirection of calls is supported by both protocols. A callee can redirect a caller to a number of destinations. Some differences can be noted regarding personal mobility service, though. H.323 was not originally designed for wide area networking, and its supported mobility services are more limited. A *facility* message can be used to redirect a caller to trying several other addresses. Neither the caller nor the redirecting party can, however, express any preferences. In SIP a list of callee priorities can be conveyed for each location, including information on language spoken, business or home number, mobile phone or fixed. The most suitable terminal is selected.

The lack of loop detection is another limiting factor in H.323. This is supported by SIP. Furthermore, H.323 does not allow a gatekeeper to proxy a request to multiple servers [13].

As an addition to normal redirection, SIP also support multi-hop searches for a user. If a contacted SIP server is not the machine that the caller is residing at, the server can proxy the request to one or more additional servers. These servers may then further proxy the request until the party is connected. Each server can proxy the request to multiple servers in parallel, allowing for fast searches. The request may be picked up by several endpoints, including answering machines, and the caller can decide which party to speak to.

Call forwarding in SIP is implemented in a simple way. The party that transfers the call sends a BYE message that contains the new destination.

4.6.2 Conference control

H.323 has better support for conference control services. These include chair selection, “mike passing” and conference participant determination. A gatekeeper hosting multiple conferences can also provide a list of available conferences [20]. SIP does not provide any conference control in addition to the RTCP based services such as sending notes and obtaining a participant list. The other services are left to supplementary protocols.

4.6.3 SIP call control services

Additional call control services can be integrated into SIP as an extension [4]. The call control service extension is included by adding the extension’s name, *org.ietf.sip.call,* in a *Require* header. It defines five new headers. With an *Also* header a client or server may request the recipient of a message to invite more endpoints. With the *Call-disposition* header, a client may indicate how the server is to handle the call. With this feature a client may prohibit forwarding, have a request queued rather than rejected, prohibit responding or request a group name to be resolved into individual members.

Using the extension it is possible to retrieve information about the terminal, which can be used when choosing from a list of terminals that received the call request. These properties include the class (personal or business), duplex, features, language, media (audio, video, text or application), mobility, priority (e.g. only emergency calls) and service (e.g. voice-mail, ISDN, pager or text). When a person is called, all the clients belonging to the callee replies. The caller can then choose, manually or automatically, which one to use and what media format. This integrates mail, chat and other applications into the multimedia conferencing domain.

4.6.4 Playback of stored media

The multimedia communications architecture can be extended with functions for playback of media streams. These streams can be used for playback of background music while waiting, listening to voice mail (answering machine) messages, music or video on demand, automated information services and similar services. The services are implemented by the IETF *Real-Time Streaming Protocol* (RTSP).

RTSP is designed for playback of real-time media that is stored on a server or generated live. The RTSP media is accessed through a session description retrieved by HTTP, FTP or mail. RTSP uses session descriptions in the same way as SDP to describe the properties of the streams. An example RTSP session description is show in Figure 10.

```
<title>Twister</title>
<session>
  <group language=en lipsync>
     <switch>
        <track type=audio e=”PCMU/8000/1”
               src=”rtsp://audio.company.com/twister/audio/lofi”>
        <track type=audio e=”DVI4/16000/2 pt=”90 DVI4/8000/1”
               src=”rtsp://audio.company.com/twister/audio/hifi”>
     </switch>
     <track type=”video/jpeg”
        src=”rtspu://video.company.com/twister/video”>
  </group>
</session>
```

Figure 10. Example RTSP session description [25]

Media can be played into a multimedia conference using one of several methods. The RTSP server acts like a member, and has to be invited. The server must also understand either H.323 or SIP. In a H.323 conference, the member has to register with the RAS or an MCU. In SIP the server is invited with an invitation message. If the conference is encrypted the key must be transferred. The server can then be controlled by the conference participants. The control functions include play, pause, stop and record. A given range can be played. The sessions are controlled by the *Session Control Protocol* (SCP) [25].

4.6.5 Call processing languages and IN services

The clients, servers and gatekeepers perform some processing of calls through the signaling protocols. However, many services are independent of the specific end terminal or need to be operational even when the end system is unavailable. These include user location services and redirection services, such as call forward on busy. Also different types of security and administrative functions are best implemented in a network device, instead of in the end systems. Network devices in this case are the gateways in H.323 and proxy and redirect servers in SIP.

Intelligent Network services (IN services) such as the examples given above, require a standardized signaling system. The services are programmed in a special purpose call processing language (CPL). Such languages have been already been existing in the PSTN. So far, no language has been defined for Internet telephony. The IETF has begun developing a call processing language. A good overview of the functions and problems is found in the IETF draft [11], where the requirement for a call processing language is discussed.

In the proposed architecture, network devices and endpoints has a set of programs that are triggered by incoming events. The program handles the reaction to the event, such as ringing the telephone when an incoming call is received, and forwarding the call if nobody answers within four rings. Users create these scripts, usually on end terminals, and transfer them to network systems where they are stored. In addition, Internet telephony providers can create scripts, which have a higher priority than user scripts. Other events are call termination, parameter changes and DTMF tones.

4.7 Security

4.7.1 What is security?

Internet, being an open network where everyone can receive and transmit any packets, requires advanced mechanisms to secure communications. Not only the audio or video stream itself needs protection. Signaling requires security, primarily authentication, to prevent spoofing of calls, denial-of-service attacks and disturbing use, e.g., “spam”. Security includes protecting call setup, call management and billing. In order to charge for Internet telephony calls the billing mechanisms have to be properly secured and implemented. [12]

Four basic aspects of the security issue are addressed in Internet Telephony: authentication, integrity, privacy and non-repudiation. *Authentication* is the process of ensuring that the participants really are who they claim to be. This is perhaps the most important concept, required by all other security services. Without authentication anyone can enter a conference claming that they are someone else. *Integrity* provides the means to check that the contents of the data within packet remain unchanged during the transit between the endpoints. This is similar to adding an encrypted checksum or CRC to the messages. *Privacy* is the encryption and decryption mechanisms to prevent eavesdroppers from viewing the contents of the streams. This requires lots of computational expense and added latency. *Non-repudiation* is a means of protection against someone denying that they participated in a conference when you know they were there. It will be of importance for service providers in the billing procedures.

Encryption can be symmetric or asymmetric. In the symmetric encryption, the simpler of them, the encryption key can be calculated from the decryption key and vice versa. Usually they are the same. Symmetric encryption cannot be used for authentication if several people know the key. In asymmetric encryption, or public key encryption, it is not possible to calculate one key from the other. One of the keys can then be made public and the other one is private. Symmetric algorithms are generally faster than asymmetric, but public key cryptosystems are vulnerable to chosen-plaintext attacks. In practice a combination of the methods is used. Messages are sent using a symmetric algorithm with a generated session dependent key, distributed with a public key algorithm [6].

4.7.2 Security in H.323

Hooks for authentication, integrity, privacy and non-repudiation are supported by H.323 version 2. Their usage is specified in standard H.235, formerly called “H.Secure”. The protocol covers authentication of users, protection of the integrity of streams and securing the privacy of the streams. Not only the streams are protected, but also call setup (Q.931), call management (H.245) and gatekeeper registration/admission/status (RAS).

Keys are used for encrypting and decrypting the streams. Exchange of keys between the communicating parties can be done in three ways. *Out-of-band* key exchange means that the key is sent via e-mail, word of mouth or by some means other than a H.323 protocol. This can be the most secure way, but can easily breakdown if unsecured e-mail or audio phone calls are used. The *Diffie-Hellman method* allows two hosts to create and share a secret key. The *Oakley Key Determination Protocol*, based on the Diffie-Hellman method, can establish session keys on Internet hosts and routers. Oakley can also be used in conjunction with the *Internet Security Association and Key Management Protocol* (ISAKMP), which provides a framework for Internet key management. The H.323 standard defines a protocol set that facilitates key exchange but leaves the choice of key exchange method open to the implementers. Most vendors will want to support different types of key exchange methods.

Privacy can be protected with one of two approaches. Either the software developers include security into their own private protocols or then they use the external *Transport Layer Security* (TLS) and *IP Security Protocol* (IPSec) techniques. IPSec operate as a transparent security layer below the application in the IP stack (IPSec) while TLS resides on top of the IP stack (TLS) and requires modifications in the application. TLS is based on *Secure Socket Layer* (SSL) and *Private Communication Technology* (PCT).

In addition to the streams, three components need to be secured: RAS, Q.931 and H.245. RAS uses unreliable UDP-based transport. With RAS the preservation of the integrity of data within the packets needs to be protected and the endpoints have to be authenticated. The standard is still incomplete. At this time, the integrity issue is incomplete and any privacy of RAS data is not defined. Typically Gatekeepers want to authenticate the client (one direction) but clients can also request the Gatekeeper to authenticate itself (bi-directional). Two techniques are available: symmetric encryption-based authentication, which is a one direction method requiring no prior contact between the endpoints, and bi-directional subscription based authentication, requiring some kind of prior contact.

The Q.931 protocol uses reliable transport. H.325 security is utilized for Q.931 by encrypting the Q.931 streams, protecting Q.931 packets from tampering and utilizing user authentication to verify endpoints in the call setup. In a similar way also H.245 is protected [22].

4.7.3 Security in SIP

As an HTTP-influenced protocol, SIP also has HTTP-like security mechanisms. Caller and callee authentication can be realized with HTTP mechanisms, including basic (clear-text password) and digest (challenge-response) authentication. Keys for media encryption are conveyed using SDP [12, 31].

The basic SIP draft does not include any security considerations, other than to specify a reliance on lower layer security mechanisms such as *Secure Socket Layer* (SSL). SSL supports symmetric and asymmetric authentication, i.e., either only the server authenticates itself or both client and server mutually authenticate themselves. Also hop-by-hop *Transport Layer Security* (TSL) is supported, but TLS is not directly applicable to SIP if it is run over UDP. According to [31], SIP could use any transport-layer or HTTP-like security mechanism, such as SSH or S-HTTP [6, 31].

Improved security mechanisms are found in SIP version 2.1, which defines end-to-end authentication and encryption using either *Pretty Good Privacy* (PGP) or *S/MIME*. Support for PGP is required while S/MIME is optional. These methods are used for signing and/or encrypting of messages. Required and recommended algorithms are listed in Table 5 [6].

Table 5. Security algorithms in SIP version 2.1 [6]

Method	Type	Algorithm	Requirement
PGP	Public Key Algorithms	DSA signatures	Required
(Required)		Elgamal encryption	Required
		RSA encryption	Recommended
	Symmetric Key Algorithms	Triple-DES	Required
		IDEA	Recommended
		CAST5	Recommended
		Other	Optional
	Compression Algorithms	Uncompressed	Required
		ZIP compressed	Recommended
	Hash Algorithms	SHA-1	Required
		MD5	Recommended
S/MIME	Public Key Algorithms	Diffie-Hellman	Required
(Optional)		RSA encryption	Recommended
	Signature Algorithm	ID-DSA	Required
		RSA encryption	Recommended
	Content Encryption Algorithm	DES	Required
		EDE3	Required
		CBC	Required
		RC2	Recommended
	Digest Algorithm	SHA-1 receive	Required
		SHA-1 send	Recommended
		MD5 receive	Recommended
		MD5 send	Optional

4.7.4 H.323 and firewalls

Implementation of firewalls supporting H.323 connections is a difficult problem. H.323 is a complex protocol that uses dynamic ports and multiple UDP streams. Changes are needed both to the H.323 application and to the firewall to get them working together.

A H.323 proxy works like most other proxies. It monitors calls and decides which are allowed to pass through the firewall. A proxy can be considered as a special case of a gateway, enforcing access control policies and not only bandwidth control.

H.323 uses dynamic port addresses. When the call is set up using the Q.931 protocol a port number for the H.245 connection is assigned, and a new TCP connection will be set up to that port. Also the media channels use dynamic ports. Each channel requires two UDP connections for the RTP streams and one bi-directional connection for the RTCP control stream. A typical audio-only conference requires two TCP connections and four UDP connections, and only one of them is not dynamic. The address and port number are exchanged within the data stream.

Because of the ASN.1 encoding with its optional and various length fields, it is impossible to pick the addresses from fixed offsets. Thus, the complete stream has to be interpreted. The proxy has to participate in the application protocol, which makes the firewall visible to the applications. It looks like a server to the internal client, while it looks like a client to the external server. This kind of proxy must perform address translation. It can be classified as an *application proxy*.

Simpler forms of firewalls exist, but they are not suitable for H.323. A *packet filtering router* has to open up all TCP and UDP ports above 1024 in each direction because of the dynamic port numbers. This does not provide much protection. A *circuit gateway* can disassemble the H.323 packets to examine the used port numbers and open up the used ports. This is difficult to implement. An *address translating firewall* changes address and port information, and has to be able to recompose data in the stream to modify addresses [20].

Gateways for SIP are much easier to implement. SIP only needs one TCP or UDP connection. One TCP connection is easy to add to the configuration. The similarity to HTTP allows for reuse of proxies and security mechanisms [31].

4.8 Other features

4.8.1 Bandwidth management

Because of the bandwidth-intensive nature of video and audio traffic, the network could be clogged due to heavy traffic and under dimensioned networks. This is especially important in local area networks and intranets. In H.323 this issue is addressed by providing bandwidth management. The number of simultaneous H.323 connections and the amount of available bandwidth to H.323 application can be limited by a network manager. This will ensure that other critical traffic will not be disrupted.

Similar functions for bandwidth management are not included in the SIP standard. However, such functions could possibly be implemented in the servers.

4.8.2 Loop detection

SIP traces the route of messages using *Via* headers. These headers are used to help messages to find their path back to the sender and for loop detection. When a server noticed that its own address is listed in the Via headers, it must not forward it. An error message is in this case sent back. This method is similar to the one used in BGP. The *Via* headers can be hidden for security reasons. In H.323 there is no easy way to perform loop detection in complex multi-domain searches, except for the non-scalable method of storing messages [13].

5. Connecting all networks together

5.1 Gateways to PSTN and ISDN

5.1.1 Gateways for voice only

Gateways for voice only are the simplest case of Internet telephony gateways. The gateway generally acts like an endpoint, that answers call requests on one side and establishes connections on the other side. It also translates the information streams, in this case the voice stream, between the networks. Gateways support four types of connections: Analog, T1/E1, ATM and ISDN (generally Primary Rate Interface, PRI). The basic signaling is reasonably straightforward, as illustrated in Figure 11 and Figure 12 for a SIP network. The same signaling model also applies for H.323, with the exception that H.323 already uses the same Q.931 signaling as ISDN uses.

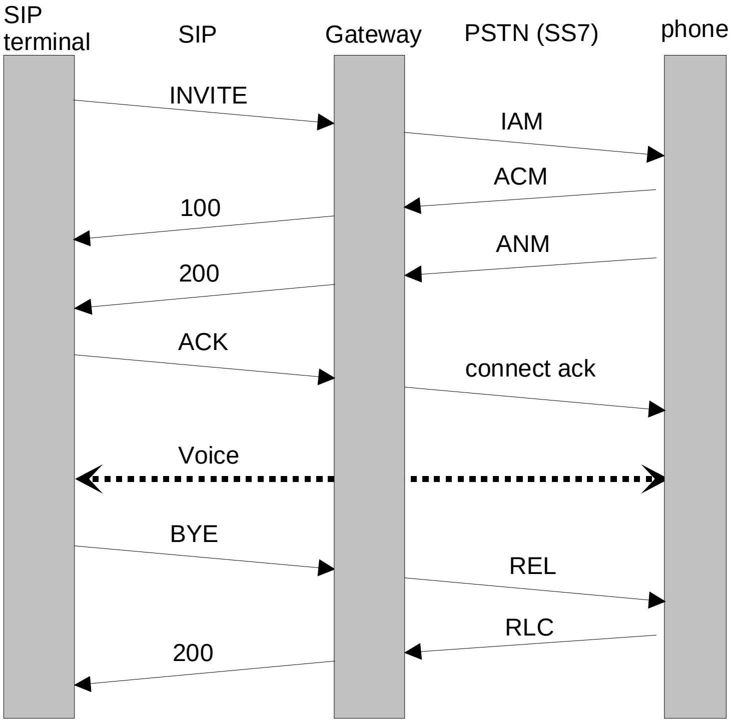

Figure 11. PSTN connection through a gateway

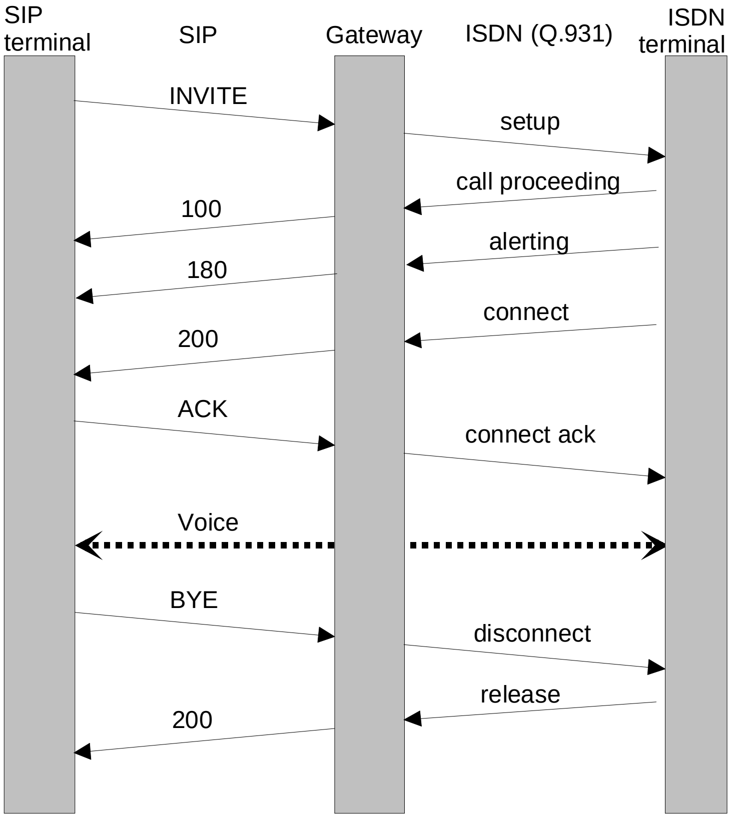

Figure 12. ISDN connection through a gateway

SIP can indicate to the Internet endpoint that the callee is reachable via an Internet telephony gateway. Also two PSTN users can establish calls through the Internet, using SIP signaling.

5.1.2 Gateways for multimedia communications

Multimedia communication is more demanding than simple voice telephony. In addition to the voice streams, the video streams have to be translated. The signaling and bandwidth requirements are bigger.

Terminals on different types of networks can be interconnected with the H.32x series of standards. Visual ISDN terminals can be connected to H.323 terminals through an H.323-H.320 gateway, and visual PSTN terminals through an H.323-H324 gateway. Visual telephone terminals over ATM networks can be connected with either a H.323-H.321 gateway or a H.323-H.320 gateway, providing there is an I.580 ISDN/ATM interworking unit in the network. Interoperation with terminals over guaranteed quality of service LANs can be provided by using an H.323-H.320 gateway assuming there is a GQOS LAN-ISDN gateway in the network. Simultaneous voice and data terminals (V.70 terminals) can be connected through an H.323-

V.70 gateway. H.323 terminals supporting T.120 data conferences can participate in T.120-only conferences.

A broadband connection requires several ISDN connections to be set up. In H.323 a list of one E.164 number for each connection is passed in the *partyNumber* field of the setup message.

Multimedia conferencing on circuit switched networks is out of the scope of SIP, and the similarities between H.323 and H.320 cannot be utilized. SIP-H.320 and SIP-H.234 gateways are thus more complicated than corresponding gateways for H.323.

5.1.3 The DTMF problem

DTMF (dual-tone multifrequency) is audio tone pairs used to control automatic devices such as answering machines. These are sent as an in-band pair of audio tones. When these tones are sent through an audio codec they may be enough distorted not to be recognized. In some applications they also need precise timing and length information. The solution is to send them as separate control messages. SIP has not yet defined such messages. In version 2 of H.323 the DTMF signals are standardized as a type of *User Input Indication* PDUs [20, 23]. Gateways have to be able to translate between DTMF and control messages.

5.2 Interoperability between H.323 and SIP

SIP terminals can only contact other SIP terminals. Correspondingly, H.323 terminals are only able to contact other H.323 terminals. If both protocols become popular, we get two islands, unable to communicate with each other. Gateways between SIP and H.323 are required, otherwise the calls must pass through the PSTN. Fortunately SIP-H.323 gateways are rather simple to implement, requiring no additional hardware than a computer translating between the signaling protocol. The information streams does not necessarily have to be translated, since the same codec can be used on both sides and the transportation is RTP based on both sides. The gateway acts like a client on both networks.

A SIP client can be used to locate any terminal and determine its capabilities, including H.323. The actual calling can then be done with a H.323 client integrated in the SIP software. In this case, the powerful hop-by-hop searches of SIP can be used to locate both H.323 and SIP endpoints and the call is established with the appropriate protocol.

Using the redirection features of SIP, it is also possible to establish calls to H.323 terminals. The SIP user could indicate that it prefers to communicate via SIP as first choice and H.323 as an alternative. If neither is possible, the call could be redirected to a plain old telephone. Both the caller and callee can indicate such preferences. The callee can also indicate a preferred time to be called back.

SIP also allows the protocols to be mixed. For example, H.323 could be used to establish a connection between end systems and gateways, while SIP might be used for gateway-to-gateway signaling.

Currently it is not possible to directly use H.323 to locate and establish calls to SIP users. However, if SIP gains popularity, it can be expected that terminals supporting both H.323 and SIP will appear.

5.3 Interoperability testing

The first Internet telephony and multimedia conferencing software that came used vendor specific signaling and was therefore incompatible with each other. As the H.323 standard was launched, vendors began to make their products H.323 compatible. However, in the practical case only a few products interoperate smoothly. According to Pulver [23], people seem to say their products "interoperate with NetMeeting" rather than being 100 percent H.323 compatible.

Not only accurately specified standards are needed to guarantee that equipment and software from different vendors work together. A lot of practical testing is also needed. ITU has the role of standards-setting body. Another organization, International Multimedia Teleconferencing Consortium (IMTC) focuses on promotion of standards and brings together organizations for practical interoperability tests. IMTC has organized tests mainly on H.324 and T.120. Interoperability of H.323 products has been tested since 1996. The test result in a multi-dimensional matrix of equipment, networks, codecs and protocols to be gradually built up.

The first generation H.323 products will mainly be used for testing. The second generation of products, based on the second version of H.323, will be available in the late 1998. The second generation will be more reliable and interoperability is improved. Problems may arise when version 1 products are conferencing with version 2 products [28].

5.4 Addressing

A hot debate about what type of addressing system should to be used in Internet telephony and on PSTN is going on. The issue is discussed in IETF (Iptel), ETSI (Tiphon) and ITU. The main topics are whether E.164 or text-format addresses should be used in the future telephone network, and how the hierarchy should be organized. The mapping of addresses from one system to another is a major question.

Telephone numbers are today limited to 10 digits and 6 special symbols (#, *, A, B, C, D), of which only 12 are available on normal telephones. On the Internet text format addresses are the common format. Text addresses are supported by both H.323 and SIP. It is simple to make calls from the Internet to the PSTN, but how do we enter text format addresses from a normal telephone? As a solution, we could give terminals both a text address and a telephone number, much as we have IP addresses and DNS textual addresses. A number translation service or a directory service is required to translate between E.164 numbers and text-format addresses. This either could be a separate service or integrated into the gateway.

In today’s implementations of H.323, the IP clients are given a telephone number in a PBX system. This requires a PBX to be used. It is also an acceptable solution for an Internet telephony service provider that has reserved a number of E.164 addresses for IP clients. However, this solution is too limited for the open Internet, where almost all connected computers run a telephony client in the future.

When a H.323 client wishes to contact a terminal using E.164 addresses through a gateway, it passes the address in the *partyNumber* field of the setup message. In SIP, addresses to PSTN and ISDN numbers are given in the email-like format “phone-number@gateway”.

This has the disadvantage that the address of the gateway has to be included in the number. To ensure that calls could be established even when a gateway is down, a list of alternative numbers is required. A more efficient solution is required, a mechanism for searching gateways.

5.5 Gateway location

In smaller companies, it can be sufficient to use one or more gateways for outgoing and incoming calls. Only gateways owned by the company are used. However, all clients does not have own gateways. Moreover, to take advantage of the low costs of transmitting the information over the Internet instead of international calls, it is often necessary to use a gateway as close to the recipient as possible. In these cases public gateways must be used. An Internet service provider (ISP) or Internet telephony service provider (ITSP) may have a large number of gateways in several regions of the country and even in several countries. In another situation, it is preferable to use a gateway as close to the caller as possible. An IP network suffers from losses and delay variation and one would prefer a local gateway when the quality is the main criteria. Gateways also have a limited capacity, which limits the maximum number of simultaneous calls that can be handled. The gateway is therefore likely to be frequently unavailable. A system for finding the location of a suitable gateway is required. Though this will be implemented as a separate protocol we will briefly discuss the topic here, since it is a significant part of the signaling.

A client has a set of requirements concerning the properties of the telephony gateway. These requirements include cost for completing a call to a specific destination on the PSTN, proximity to the client and supported protocols. The gateway location protocol has to be dynamic to quickly reflect changes in the network. In the case of a distributed database, all individual databases must be updated quickly. Gateways are expected to become available and unavailable (all lines in use or off-line) frequently. It has to be scalable and include some sort of hierarchy, in order to handle large amounts of information and many queries per time unit.

Location of services on the Internet is not a new problem and solutions already exist. Examples of such location protocols are BGP, DNS, LDAP, X.500, whois++, SAP, search engines and indexing tools. They use either central or distributed databases. Their suitability for gateway location was examined in an IETF report [24]. Unfortunately, no solution was found enough scalable and dynamic. Neither the *service location protocol*, under development by the srvloc working group of the IETF scales well for Internet telephony. In the same report [24] a new solution was proposed, *brokered multicast advertisements* (BMA), combining the best parts of all protocols. In this architecture a number of brokers keep a distributed database of the gateways. The brokers communicate with each other and update the database through multicast groups. They accept queries from the clients, which specify desired features for a gateway. BMA is scalable, bandwidth efficient and simple.

5.6 Billing

Data transfer on the Internet backbone networks is free today, but the use of gateways and the establishment of calls to PSTN telephones always involve some costs. In an intranet a gatekeeper or server can record usage and count charges. In the case of public gateways and other public services a separate billing system is needed.

Billing is to a large extent related to the security issue. Authentication is required to ensure that the callers really are the ones they claim to be. Encryption and integrity mechanisms are required to protect the billing information. With non-repudiation a telephony provider can prove that a customer has used their services.

In addition to the basic point-to-point calls, the network will be used for multipoint calls and intelligent network services, possibly involving multiple telephony providers. These require advanced billing systems. Features such as account card dialing, automatic alternative billing, credit card calling, reverse charging, freephone, premium rate and split charging need to be supported [4].

Billing will probably be supported by separate protocols, such as RADIUS. Some mechanism could, however, be integrated into the signaling protocols, such as indication of billing preferences and capabilities. Also informative messages could be sent, such as indication of charges at call setup, during the call and at the end of the call. Charges for resource reservation are best indicated by a protocol closely affiliated with the reservation protocol [4].

6. Applications

6.1 Available H.323 compatible clients

The H.323 market has so far mainly been represented by Microsoft NetMeeting and Intel Video Phone. NetMeeting is distributed with Microsoft Explorer 4 and Windows 98. It has become a standard reference application, used for testing interoperability. Also Netscape Conference has a large number of users, since it is included as a part of the Netscape Communicator suite.

There are a large number of programs for Internet telephony and videoconferencing available. Still only a few of them supports H.323. The other ones are only able to connect to client software made by the same vendor. Often even the different versions of the same program are incompatible. A large part of the non-H.323 products will be equipped with H.323 in the near future. An example is VocalTec Internet Phone 5.0, which has all the features of a H.323 but the H.323 compatibility will be added in the next version. As another example, VDO Phone Professional supports the ITU-T protocols H.324 and T.120 over telephone lines but uses an incompatible protocol over IP.

The most popular H.323 clients are presented in Table 6 and in Table 7. Nearly all software clients are able to send and receive video. The powerful computers of today make it possible to transfer video without any hardware devices. Most programs support the required G.711 audio codec, but the H.323 recommendation would require it in *all* clients. It is more probable that the clients will connect using the G.723 codec. Microsoft NetMeeting, Netscape Conference and Intel’s conferencing solutions include complete T.120 support, with file transfer, whiteboard and text chat. Features like auto-detection of voice are found in most programs, but echo cancellation is still a rare feature. One should also notice that most programs run under Windows and Windows NT; other platforms are so far only represented by Netscape.

Table 6. Some H.323 software clients

Manufacturer	Microsoft	Netscape	Intel
Product	NetMeeting	Conference	Video Phone
Version	2.1	4.06	2.1 (trial applet)
Audio codecs	G.711, G.723, TrueSpeech, GSM 06.10, PCM, ADPCM	G.711, RT24, RT29, SX8300P	G.711, G.723.1, GSM 06.10
Video codecs	H.263	No video	H.261, H.263
Data channel	T.120	T.120	T.120
Other support			

Table 7. Some more H.323 software clients

Manufacturer	Voxware	NetSpeak	White Pine Software
Product	VoxPhone Pro	Webphone	CU-SeeMe
Version	3.0	4.0 (full)	3.1
Audio codecs	RT24, RT28, RT29, G.711 (?)	G.723.1, TrueSpeech, GSM 06.10	G.723
Video codecs	No video	H.263	H.263
Data channel	File transfer	No data channel (?)	No data channel
Other support			

CU-SeeMe from White Pine Software is a special case. It is not able to establish H.323 connections directly from the client software. A special server is used for making these calls.

Most users will prefer using a telephone with a traditional design. Selsius sells an IP-phone that looks and works like a normal PBX telephone. From these devices, one could not expect H.323 specific services as video and data sharing. The telephone uses the standard audio codecs G.711 and G.723.

Many vendors sell more or less complete packages for video conferencing. One type of package consists of equipment for a computer. These include hardware audio and video codecs, video capture board, camera and microphone. Another type of package is connected to a standard television and includes a camera. Two such solutions are presented in Table 8.

Table 8. Some H.323 hardware client solutions

Manufacturer	*Intel*	*Selsius*	*Vcon*
Product	BusinessVideo Conferencing	Selsius-Phone	Armada MediaConnect 6000
Type	Software + hardware	Stand alone telephone	Complete solution
Hardware	Video capture/audio board, ISDN board, camera, microphone	Telephone	Computer, Network Interface card
Software	Client software	No computer required	Vcon MonitorPoint, Microsoft NetMeeting
Requires	Computer	Nothing	Standard TV, camera, microphone
Audio codecs	G.711, G.723, G.728	G.711, G.723	G.711, G.728, G.722, G.723
Video codecs	H.261, H.263	No video	H.261, H.263
Data channel	T.120	No data channel	T.120
Other support	H.320 (ISDN), H.234 (POTS)		H.320 (ISDN), echo cancellation

6.2 Some SIP based products

Since SIP is a rather new and unknown protocol, there are only a few implementations of SIP clients. Most of these are experimental software, developed in universities. Because of its simplicity, SIP is a suitable protocol for learning multimedia communications. A basic client or server is easy to implement, and can for example be used for testing real-time data transmission and synchronization.

Four user agents are currently available. The University of California has developed a session directory tool for multicast conferences on the Mbone, supporting SIP, SAP and SDP. The German National Research Center for Information Technology (GMD) supports SIP in their isc user agent. The University College London has implemented a user agent based in Java called Java Internet Phone. Luleå University of Technology is developing a client called mSIP. Lucent includes SIP support in their user agent. Lucent is also developing a proxy server for SIP. The only redirect and registration server so far is SIPD, developed by Columbia University.

6.3 Gateways

Gateways between Internet telephony and the PSTN will have a big market as the popularity of Internet telephony grows. Plain old telephones will be the dominant end systems for many decades, and a significant part of the calls from/to an IP phone will go through a gateway. The number of commercially available gateways is large, even larger than the number of clients.

The web page of Pulver [33] lists about 24 hardware gateway providers and 14 software gateway providers. All these are H.323 gateways.

Most of the gateway solutions are built around an ISA bus. The bus is equipped with a number of ISA cards, which handles either analog telephone lines or digital T1/E1/PRI connections. Smaller gateways usually have 4-8 analog telephone lines that can be connected to a PBX. The largest gateways support up to eight T1/E1/PRI cards resulting in 240 full duplex lines per gateway. The PSTN connections are used either for voice-only communication to POTS terminals or for multimedia conferencing to H.320 terminals. The gateways connect to the IP network through standard network interface adapters for Ethernet, Token Ring or ATM. Optional features, found in most gateways, are echo cancellation, DTMF support and interactive voice response. Some of the gateways also include gatekeeper functions.

We found three gateways supporting SIP. The Audiotrix Phone Adapter II (APA) is a hardware gateway solution for H.323 and SIP. APA connects a single telephone handset to a local area network and to the PSTN. Ericsson is developing a gateway that supports both H.323 and SIP on the Internet side, and SS7 on the PSTN side. In addition to opening connections between Internet telephones and PSTN telephones, it is also able to translate between H.323 and SIP. Lucent support SIP in their gateway. As we can see, there is no pure SIP gateway. SIP is supported as an additional feature in H.323 gateways.

7. Conclusions

Which signaling protocol will be used for Internet telephony tomorrow? It is much too early to say anything yet, as the protocols are still developing and the number of supporting applications is small. In its first version, H.323 had some important problems, which were solved in SIP. Some of them concerned wide area usage. The second version of H.323 corrected these drawbacks, and today both protocols are in practice equally well suited for Internet telephony.

With its large range of products, it is quite certain that H.323 will be the main architecture for Internet telephony for several years. All commercially available products are based on H.323, except for a large number of nonstandard systems running their own protocols. H.323 is a complete and comprising protocol, including advanced management and capability exchange functions. H.323 is also very similar to the other H.32x protocols, such as H.320 for ISDN, which simplifies connections to already available equipment. The support for SIP is almost nonexistent today. Furthermore, it is difficult to gain acceptance with all the largest software developers (Microsoft, Intel, Netscape) supporting H.323.

Simplicity is strength, especially on the Internet with many small software developers. Only a few larger developers can make H.323 clients, because of complicity and rather expensive

H.323 stacks. Internet telephony, being nearly as popular in the future as Internet email is today, is expected to be integrated in many applications. In addition to traditional telephony, the future will bring new types of services, based on speech transmission. Examples of such services are speech interfaces to applications, utilizing speech synthesis and voice recognition. An application could be remotely controlled, for example providing information at request. This kind of applications would require lightweight signaling protocols, like SIP, that can easily be integrated into the software. On the other hand, even larger software libraries, such as H.323 stacks, could be easily accessed through application programming interfaces (API). If we compare the positions of the mail transfer protocols X.400 and SMTP, we can see that simplicity, extensibility and easy access are important factors for gaining widespread usage in the Internet.

It is possible that some applications and devices will support both protocol, but on the long run, only one protocol is desired. Including two protocol stacks, especially with the size of H.323, in every device is not very practical.

In order of the protocols to co-exist, it would be desirable that the protocols would find niches of their own. Today it seems like that small developer would use SIP and larger developer would use H.323. This is not a good solution, since it results in two separate signaling systems. Another division could be as follows. H.323 is originally designed and still best suited for intranet telephony and could thus be used in the internal telephone systems of a company. Here its bandwidth control and authorization functions are important. SIP is noticeably more well suited for telephony on the Internet, especially utilizing the other IETF protocols, like gateway location protocols. SIP also has good functions for finding an endpoint and choosing a suitable terminal from a list of alternatives. Also loop detection and HTML similarity would come to use. Interconnection of these systems would require a SIP–H.323 gateway if other streams than voice are to be transmitted.

It would also be possible to mix the protocols by using different protocols in different phases of a call. For example, SIP could be used for address lookup and for finding a suitable end device, while H.323 is used for the actual call. This would be especially useful for wide area Internet calls. Though this is an interesting alternative, it is unlike to be popular in an uncoordinated market.

The choice of protocol is done by the developers, Internet telephony service providers and the customers. SIP is a powerful but new and still quite unknown protocol. It is very likely that H.323 will be the dominant protocol in the future.

8. References

ITU-T recommendations

[1] International Telecommunication Union, "Draft ITU-T Recommendation H.323, Visual telephone systems and equipment for local area networks which provide a non-guaranteed quality of service", Telecommunication Standardization Section of ITU, Geneva, Switzerland, May 1996.

[2] International Telecommunication Union, “ITU-T Recommendation H.245, Control protocol for multimedia communication”, Telecommunication Standardization Section of ITU, Geneva, Switzerland, June 1996

Internet drafts and RFC

[3] M. Handley, H. Schulzrinne, E. Shooler, “SIP: Session Initiation Protocol”, Internet draft, Internet Engineering Task Force, MMUSIC working group, May 1998, Work in progress

[4] H. Schulzrinne, J. Rosenberg, “SIP call control services”, Internet draft, Internet Engineering Task Force, Feb. 1998, Work in progress

[5] H. Schulzrinne, “SIP for Click-To-Dial-Back and Third-Party Control”, Internet draft, Internet Engineering Task Force, PINT work group, Apr 1998, Work in progress

[6] P. Kirstein, G. Montasser-Hohsari, E. Whelan, “SIP Security Using Public Key Algorithms”, Internet draft, Internet Engineering Task Force, Mar. 1998, Work in progress

[7] J. Rosenberg, H. Schulzrinne, “A Framework for a Gateway Location Protocol”, Internet draft, Internet Engineering Task Force, IPTEL working group, July 1998, Work in progress

[8] H. Schulzrinne, J. Rosenberg, “SIP Call Control Services”, Internet draft, Internet Engineering Task Force, MMUSIC working group, Mar. 1998, Work in progress

[9] J. Rosenberg, H. Schulzrinne, “A Framework for a Gateway Location Protocol”, Internet draft, Internet Engineering Task Force, IPTEL working group, July 1998, Work in progress

[10] A. Vähä-Sipilä, “URLs for Telephone Calls”, Internet draft, Aug 1998, Work in progress

[11] J. Lennox, H. Schulzrinne, “Call Processing Language Requirements”, Internet draft, Internet Engineering Task Force, IPTEL working group, July 1998, Work in progress

Other

[12] H. Schulzrinne, J. Rosenberg, “Signaling for Internet Telephony”, Technical Report CUCS—005-98, Columbia University, New York, Feb. 1998.

[13] H. Schulzrinne, J. Rosenberg, “A Comparison of SIP and H.323 for Internet Telephony”

[14] DataBeam Corporation, “A Primer on the H.323 Series Standard, Version 2.0”, Lexington, USA, May 1998

[15] DataBeam Corporation, “A Primer on the T.120 Series Standard”, Lexington, USA, May 1997

[16] E. Kerttula, “Multimedialla tiedon valtatielle”, pp. 167-182, Edita, Helsinki, 1996

[17] R. Westwater, Borko Furth, “Real-time Video Compression”, pp. 23-28, Kluwer Academic Publishers, USA, 1997

[18] H. Schulzrinne, “Requirements for SIP Servers and User Agents”, Columbia University, Mar. 1998

[19] T. Yletyinen, “The Quality of Voice over IP“, Master’s Thesis, Lab. of Telecommunications Technology, Helsinki University of Technology, Mar. 1998

[20] Anonymous, “H.323 and firewalls: The problems and pitfalls of getting H.323 safely through firewalls”, Developer note, Intel Corporation, Apr. 1997

[21] Anonymous, “H.323 Version 2 - Overview”, Developer note, DataBeam Corporation, http://www.databeam.com/h323/whatsnew_v2.html

[22] E. Newman, “Security For H.323-Based Telephony”, CTI Magazine Volume 3 Number 5, May 1998, also available at: http://www.databeam.com/newsroom/articles/h323security-cti.html

[23] P. Bernier, “Standards Adoption Key For IP Voice Services”, Interactive week, Oct. 1997

[24] J. Rosenberg, H. Schulzrinne, “Internet Telephony Gateway Location”

[25] H. Schulzrinne, “A comprehensive multimedia control architecture for the Internet”, Columbia University, New York

[26] Heiner Erne, “Announcing, Initiating and Archiving Desktop Conferences within WWW”, University of Ulm, Feb 1997

[27] Bruce Kravitz, “H.323 Technology - Part 2: The Standard in Greater Detail”, VTEL corporation, 1998

[28] Bruce Kravitz, “H.323 Technology - Part 3: Real World Deployment, Issues and Answers”, VTEL corporation, 1998

Mailing lists

[29] IETF Multiparty Multimedia Session Control (MMUSIC) working group mailing list, confctrl@isi.edu

[30] IETF IPTEL working group mailing list, majordomo@lists.research.bell-labs.com

Web-pages

[31] H. Schulzrinne, “SIP: FAQ”, http://www.cs.columbia.edu/~hgs/sip/faq.html

[32] J. Crowcroft, T. Dorcey, C. Huitema, “Comments about H.323 and SIP”, http://www.cs.columbia.edu/~hgs/sip/h323.html

[33] Pulver.com, “H.323 resources”, http://pulver.com/h323/

[34] A Crossman, “Summary of ITU-T Speech / Audio Codecs Used In The ITU-T Videoconferencing Standards”, http://pulver.com/h323/