



INTERNATIONAL TELECOMMUNICATION UNION

**ITU-T**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

**H.323**

(09/99)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Systems and  
terminal equipment for audiovisual services

---

**Packet-based multimedia communications  
systems**

ITU-T Recommendation H.323

(Previously CCITT Recommendation)

---

ITU-T H-SERIES RECOMMENDATIONS  
**AUDIOVISUAL AND MULTIMEDIA SYSTEMS**

Characteristics of transmission channels used for other than telephone purposes	H.10–H.19
Use of telephone-type circuits for voice-frequency telegraphy	H.20–H.29
Telephone circuits or cables used for various types of telegraph transmission or simultaneous transmission	H.30–H.39
Telephone-type circuits used for facsimile telegraphy	H.40–H.49
Characteristics of data signals	H.50–H.99
CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
<b>Systems and terminal equipment for audiovisual services</b>	<b>H.300–H.399</b>
Supplementary services for multimedia	H.450–H.499

*For further details, please refer to ITU-T List of Recommendations.*

## ITU-T RECOMMENDATION H.323

### PACKET-BASED MULTIMEDIA COMMUNICATIONS SYSTEMS

#### Summary

This Recommendation describes terminals and other entities that provide multimedia communications services over Packet Based Networks (PBNs) which may not provide a guaranteed Quality of Service. H.323 entities may provide real-time audio, video and/or data communications. Support for audio is mandatory, while data and video are optional, but if supported, the ability to use a specified common mode of operation is required, so that all terminals supporting that media type can interwork.

The packet based network over which H.323 entities communicate may be a point-to-point connection, a single network segment, or an internetwork having multiple segments with complex topologies.

H.323 entities may be used in point-to-point, multipoint, or broadcast (as described in Recommendation H.332) configurations. They may interwork with H.310 terminals on B-ISDN, H.320 terminals on N-ISDN, H.321 terminals on B-ISDN, H.322 terminals on Guaranteed Quality of Service LANs, H.324 terminals on GSTN and wireless networks, V.70 terminals on GSTN, and voice terminals on GSTN or ISDN through the use of Gateways.

H.323 entities may be integrated into personal computers or implemented in stand-alone devices such as videotelephones.

Products claiming compliance with Version 1 of H.323 shall comply with all of the mandatory requirements of H.323 (1996) which references Recommendations H.225.0 (1996) and H.245 (1996). Version 1 products can be identified by H.225.0 messages containing a **protocolIdentifier** = {itu-t(0) recommendation(0) h(8) 2250 version(0) 1} and H.245 messages containing a **protocolIdentifier** = {itu-t(0) recommendation(0) h(8) 245 version(0) 2}. Products claiming compliance with Version 2 of H.323 shall comply with all of the mandatory requirements of this Recommendation, H.323 (1998), which references Recommendations H.225.0 (1998) and H.245 (1998). Version 2 products can be identified by H.225.0 messages containing a **protocolIdentifier** = {itu-t(0) recommendation(0) h(8) 2250 version(0) 2} and H.245 messages containing a **protocolIdentifier** = {itu-t(0) recommendation(0) h(8) 245 version(0) 3}. Products claiming compliance with Version 3 of H.323 shall comply with all of the mandatory requirements of this Recommendation, H.323 (1999), which references Recommendation H.225.0 (1999) and H.245 (1999). Version 3 products can be identified by H.225.0 messages containing a **protocolIdentifier** = {itu-t(0) recommendation(0) h(8) 2250 version(0) 3} and H.245 messages containing a **protocolIdentifier** = {itu-t(0) recommendation(0) h(8) 245 version(0) 5}.

Support of H.225.0 (1998) and H.245 (1998 or later) as identified in messages above, shall be the singular requirement and definition of H.323 systems which are H.323 Version 2 compliant. Support of H.225.0 (1999) and H.245 (1999 or later) as identified in messages above, shall be the singular requirement and definition of H.323 systems which are H.323 Version 3 compliant.

Note that the title of H.323 (1996) was "Visual telephone systems and equipment for local area networks which provide a non-guaranteed quality of service". The title changed in Version 2 to be consistent with its expanded scope.

#### Source

ITU-T Recommendation H.323 was revised by ITU-T Study Group 16 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on 30 September 1999.

## FOREWORD

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

## NOTE

In this Recommendation the term *recognized operating agency (ROA)* includes any individual, company, corporation or governmental organization that operates a public correspondence service. The terms *Administration*, *ROA* and *public correspondence* are defined in the *Constitution of the ITU (Geneva, 1992)*.

## INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. The ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, the ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

© ITU 2000

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the ITU.

## CONTENTS

	<b>Page</b>
1	Scope..... 1
2	Normative references ..... 2
3	Definitions ..... 4
4	Symbols and abbreviations ..... 9
5	Conventions ..... 12
6	System description ..... 12
6.1	Information streams ..... 12
6.2	Terminal characteristics ..... 13
6.2.1	Terminal elements outside the scope of this Recommendation..... 14
6.2.2	Terminal elements within the scope of this Recommendation..... 14
6.2.3	Packet based network interface ..... 14
6.2.4	Video codec ..... 15
6.2.5	Audio codec ..... 15
6.2.6	Receive path delay ..... 17
6.2.7	Data channel ..... 17
6.2.8	H.245 control function..... 19
6.2.9	RAS signalling function ..... 23
6.2.10	Call signalling function ..... 23
6.2.11	H.225.0 layer ..... 23
6.3	Gateway characteristics..... 24
6.4	Gatekeeper characteristics..... 26
6.5	Multipoint controller characteristics..... 27
6.6	Multipoint processor characteristics ..... 29
6.7	Multipoint control unit characteristics..... 29
6.8	Multipoint capability..... 30
6.8.1	Centralized multipoint capability ..... 30
6.8.2	Decentralized multipoint capability..... 30
6.8.3	Hybrid multipoint-centralized audio..... 30
6.8.4	Hybrid multipoint-centralized video..... 31
6.8.5	Establishment of common mode ..... 31
6.8.6	Multipoint rate matching ..... 31
6.8.7	Multipoint lip synchronization ..... 31
6.8.8	Multipoint encryption ..... 32
6.8.9	Cascading multipoint control units..... 32

	<b>Page</b>
7	Call signalling ..... 32
7.1	Addresses ..... 32
7.1.1	Network address ..... 32
7.1.2	TSAP identifier..... 32
7.1.3	Alias address ..... 33
7.2	Registration, Admission and Status (RAS) channel ..... 33
7.2.1	Gatekeeper discovery..... 33
7.2.2	Endpoint registration ..... 34
7.2.3	Endpoint location..... 36
7.2.4	Admissions, bandwidth change, status and disengage ..... 37
7.2.5	Access tokens ..... 37
7.3	Call signalling channel..... 38
7.3.1	Call signalling channel routing..... 38
7.3.2	Control channel routing..... 39
7.3.3	Call Signalling and control protocol revisions ..... 40
7.4	Call reference value ..... 41
7.5	Call ID..... 41
7.6	Conference ID and Conference Goal ..... 41
8	Call signalling procedures..... 41
8.1	Phase A – Call setup ..... 41
8.1.1	Basic call setup – Neither endpoint registered..... 42
8.1.2	Both endpoints registered to the same Gatekeeper ..... 42
8.1.3	Only calling endpoint has Gatekeeper ..... 44
8.1.4	Only called endpoint has Gatekeeper ..... 45
8.1.5	Both endpoints registered to different Gatekeepers..... 47
8.1.6	Optional Called Endpoint Signalling..... 51
8.1.7	Fast Connect Procedure ..... 53
8.1.8	Call setup via gateways..... 55
8.1.9	Call setup with an MCU ..... 56
8.1.10	Call forwarding..... 57
8.1.11	Broadcast call setup ..... 57
8.1.12	Overlapped Sending..... 57
8.1.13	Call setup to conference alias ..... 57
8.2	Phase B – Initial communication and capability exchange..... 59
8.2.1	Encapsulation of H.245 messages within Q.931 messages ..... 59
8.2.2	Tunnelling through intermediate signalling entities ..... 60
8.2.3	Switching to a separate H.245 connection..... 60

	<b>Page</b>
8.3	Phase C – Establishment of audiovisual communication ..... 61
8.3.1	Mode changes ..... 61
8.3.2	Exchange of video by mutual agreement..... 61
8.3.3	Media stream address distribution..... 62
8.3.4	Correlation of media streams in multipoint conferences..... 62
8.3.5	Communication Mode Command Procedures..... 62
8.4	Phase D – Call services..... 63
8.4.1	Bandwidth changes..... 63
8.4.2	Status ..... 65
8.4.3	Ad hoc conference expansion..... 66
8.4.4	Supplementary services ..... 75
8.4.5	Multipoint cascading ..... 75
8.4.6	Third party initiated pause and re-routing ..... 76
8.5	Phase E – Call termination..... 77
8.5.1	Call clearing without a Gatekeeper..... 77
8.5.2	Call clearing with a Gatekeeper..... 78
8.5.3	Call clearing by Gatekeeper..... 78
8.6	Protocol failure handling..... 79
9	Interoperation with other terminal types ..... 80
9.1	Speech-only terminals..... 80
9.2	Visual telephone terminals over the ISDN (H.320)..... 80
9.3	Visual telephone terminals over GSTN (H.324)..... 80
9.4	Visual telephone terminals over mobile radio (H.324/M)..... 81
9.5	Visual telephone terminals over ATM (H.321 and H.310 RAST) ..... 81
9.6	Visual telephone terminals over guaranteed quality of service LANs (H.322) ..... 81
9.7	Simultaneous voice and data terminals over GSTN (V.70)..... 81
9.8	T.120 terminals on the packet based network..... 82
9.9	Gateway for H.323 Media Transport Over ATM ..... 82
10	Optional enhancements..... 82
10.1	Encryption..... 82
10.2	Multipoint operation ..... 82
10.2.1	H.243 Control and Indication ..... 82
11	Maintenance..... 82
11.1	Loopbacks for maintenance purposes ..... 82
11.2	Monitoring methods..... 83

	<b>Page</b>
Annex A – H.245 messages used by H.323 endpoints.....	83
Annex B – Procedures for layered video codecs.....	88
B.1 Scope.....	88
B.2 Introduction.....	88
B.3 Scalability methods.....	89
B.4 Call establishment.....	89
B.5 Use of RTP sessions and codec layers.....	89
B.5.1 Associate base to audio for lip synchronization.....	89
B.5.2 Enhancement layer dependency.....	90
B.6 Possible layering models.....	91
B.6.1 Multiple logical channels and RTP sessions for a layered stream.....	91
B.6.2 Impact of one layer per logical channel and per RTP session.....	91
B.7 Impact on multipoint conferences.....	91
B.7.1 MC Impartial model.....	92
B.7.2 MC Decision model.....	92
B.7.3 Multipoint conference containing endpoints on different bandwidths.....	92
B.8 Use of network QOS for layered video streams.....	94
Annex C – H.323 on ATM.....	94
C.1 Introduction.....	94
C.2 Scope.....	94
C.2.1 Point-to-point conferencing.....	94
C.2.2 MCU-based multipoint.....	94
C.2.3 H.323 interoperability with endpoints using IP.....	95
C.3 Architecture.....	95
C.3.1 Overview of system.....	95
C.3.2 Interoperation with other ITU-T H-Series endpoints.....	95
C.3.3 H.225.0 on IP over ATM.....	95
C.3.4 H.245 on TCP/IP over ATM.....	96
C.3.5 Addressing for A/V streams.....	96
C.3.6 Transport Capabilities added to TransportCapability Set.....	96
C.3.7 Elements of ATM signalling.....	96
C.3.8 A/V streams on RTP on AAL 5.....	97
C.3.9 QOS considerations (Optional).....	98
C.4 Protocol section.....	99
C.4.1 ATM signalling information elements.....	99
C.4.2 H.245 Usage.....	101
C.4.3 RTP usage.....	103
C.4.4 Interoperation with H.323 on IP.....	103



	<b>Page</b>
Annex D – Real-time facsimile over H.323 systems .....	103
D.1 Introduction.....	103
D.2 Scope.....	104
D.3 Normative references .....	104
D.4 Procedures for opening channels to send T.38 packets .....	105
D.4.1 Opening the voice channel.....	105
D.4.2 Opening the facsimile channels .....	105
D.4.3 DTMF transmission.....	105
D.5 Non-Fast Start procedures.....	105
D.6 Usage of the MaxBitRate in messages.....	105
D.7 Interactions with gateways and T.38/Annex B devices .....	106
Appendix I – Sample MC to Terminal Communication Mode Command.....	106
I.1 Sample conference Scenario A.....	106
I.2 CommunicationModeTable sent to all Endpoints .....	107
I.3 Sample conference Scenario B .....	107
I.4 CommunicationModeTable sent to all Endpoints .....	108
Appendix II – Transport Level Resource Reservation Procedures .....	109
II.1 Introduction.....	109
II.2 QOS Support for H.323 .....	109
II.3 RSVP background.....	111
II.4 The H.245 capability exchange phase.....	112
II.5 Open logical channel and setting up reservations.....	113
II.6 Close logical channel and tearing down reservations .....	115
II.7 Resource reservation for multicast H.323 logical channels.....	115
Appendix III – Gatekeeper Based User Location.....	116
III.1 Introduction.....	116
III.2 Signalling .....	116
Appendix IV – Signalling Prioritized Alternative Logical Channels in H.245.....	118
IV.1 Introduction.....	118
IV.2 Signalling .....	118



## **Recommendation H.323**

### **PACKET-BASED MULTIMEDIA COMMUNICATIONS SYSTEMS**

*(revised in 1999)*

#### **1 Scope**

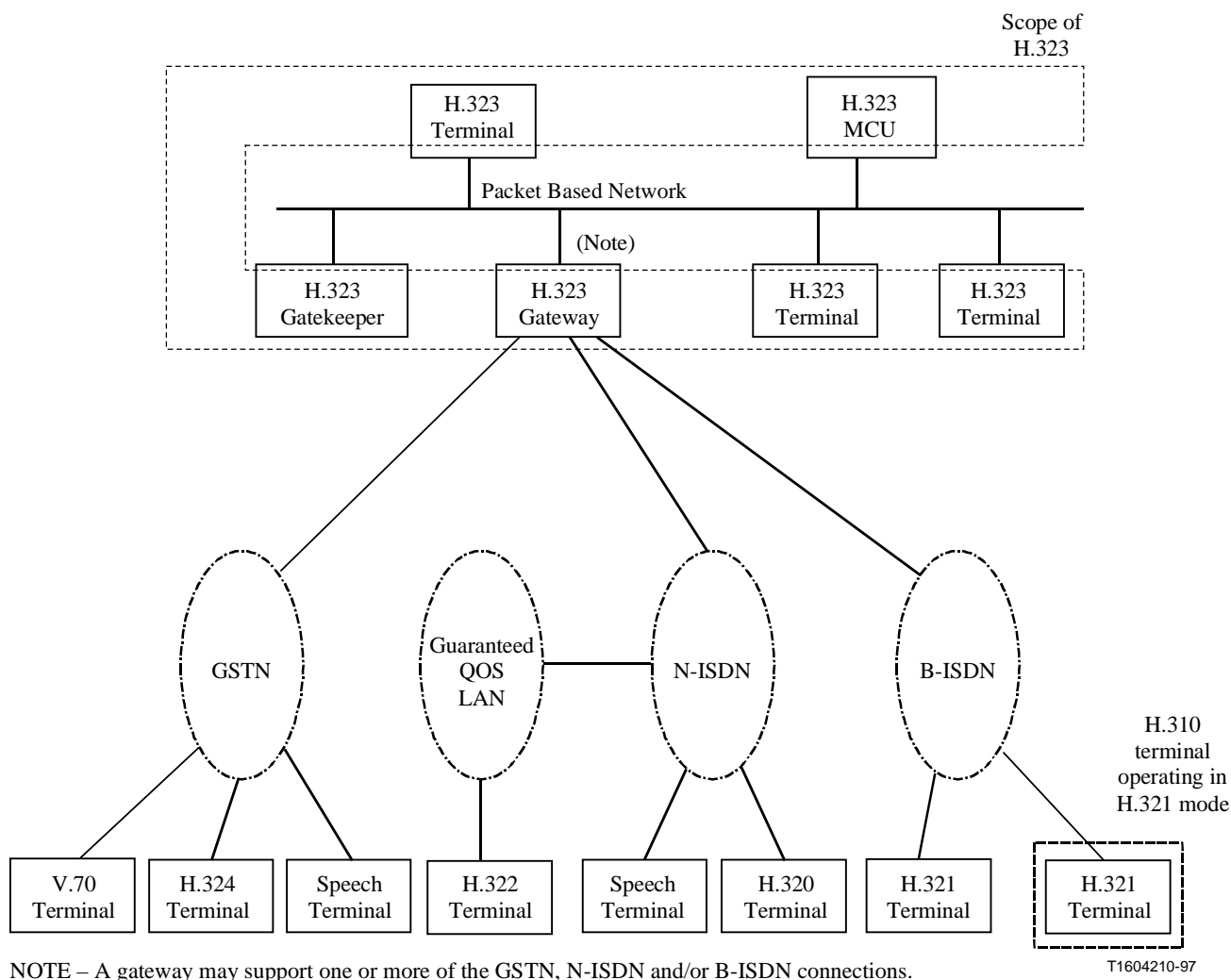
This Recommendation covers the technical requirements for multimedia communications systems in those situations where the underlying transport is a Packet Based Network (PBN) which may not provide a guaranteed Quality of Service (QOS). These packet based networks may include Local Area Networks, Enterprise Area Networks, Metropolitan Area Networks, Intra-Networks, and Inter-Networks (including the Internet). They also include dialup connections or point-to-point connections over the GSTN or ISDN which use an underlying packet based transport such as PPP. These networks may consist of a single network segment, or they may have complex topologies which incorporate many network segments interconnected by other communications links.

This Recommendation describes the components of an H.323 system. This includes Terminals, Gateways, Gatekeepers, Multipoint Controllers, Multipoint Processors, and Multipoint Control Units. Control messages and procedures within this Recommendation define how these components communicate. Detailed descriptions of these components are contained in clause 6.

H.323 terminals provide audio and optionally video and data communications capability in point-to-point or multipoint conferences. Interworking with other H-series terminals, GSTN or ISDN voice terminals, or GSTN or ISDN data terminals is accomplished using Gateways, (see Figure 1). Gatekeepers provide admission control and address translation services. Multipoint Controllers, Multipoint Processors and Multipoint Control Units provide support for multipoint conferences.

The scope of H.323 does not include the network interface, the physical network, or the transport protocol used on the network. Examples of these networks include but are not limited to:

- Ethernet (IEEE 802.3);
- Fast Ethernet (IEEE 802.3u);
- FDDI;
- Token Ring (IEEE 802.5);
- ATM.



**Figure 1/H.323 – Interoperability of H.323 terminals**

## 2 Normative references

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

- [1] ITU-T Recommendation H.225.0 (1998), *Call signalling protocols and media stream packetization for packet based multimedia communication systems*.
- [2] ITU-T Recommendation H.245 (1999), *Control protocol for multimedia communication*.
- [3] CCITT Recommendation G.711 (1988), *Pulse Code Modulation (PCM) of voice frequencies*.
- [4] CCITT Recommendation G.722 (1988), *7 kHz audio-coding within 64 kbit/s*.
- [5] ITU-T Recommendation G.723.1 (1996), *Speech coders: Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s*.

- [6] CCITT Recommendation G.728 (1992), *Coding of speech at 16 kbit/s using low-delay code excited linear prediction.*
- [7] ITU-T Recommendation G.729 (1996), *Coding of speech at 8 kbit/s using conjugate structure algebraic-code-excited linear-prediction.*
- [8] ITU-T Recommendation H.261 (1993), *Video codec for audiovisual services at  $p \times 64$  kbit/s.*
- [9] ITU-T Recommendation H.263 (1998), *Video coding for low bit rate communication.*
- [10] ITU-T Recommendation T.120 (1996), *Data protocols for multimedia conferencing.*
- [11] ITU-T Recommendation H.320 (1999), *Narrow-band visual telephone systems and terminal equipment.*
- [12] ITU-T Recommendation H.321 (1998), *Adaptation of H.320 visual telephone terminals to B-ISDN environments.*
- [13] ITU-T Recommendation H.322 (1996), *Visual telephone systems and terminal equipment for local area networks which provide a guaranteed quality of service.*
- [14] ITU-T Recommendation H.324 (1998), *Terminal for low bit-rate multimedia communication.*
- [15] ITU-T Recommendation H.310 (1998), *Broadband audiovisual communication systems and terminals.*
- [16] ITU-T Recommendation Q.931 (1998), *ISDN user-network interface layer 3 specification for basic call control.*
- [17] ITU-T Recommendation Q.932 (1998), *Digital Subscriber Signalling System No. 1 – Generic procedures for the control of ISDN supplementary services.*
- [18] ITU-T Recommendation Q.950 (1997), *Supplementary services protocols, structure and general principles.*
- [19] ISO/IEC 10646-1:1993, *Information technology – Universal Multiple-Octet Coded Character Set (USC) – Part 1: Architecture and Basic Multilingual Plane.*
- [20] ITU-T Recommendation E.164 (1997), *The international public telecommunication numbering plan.*
- [21] ITU-T Recommendation H.246 (1998), *Interworking of H-Series multimedia terminals with H-Series multimedia terminals and voice/voiceband terminals on GSTN and ISDN.*
- [22] ITU-T Recommendation H.235 (1998), *Security and encryption for H-Series (H.323 and other H.245-based) multimedia terminals.*
- [23] ITU-T Recommendation H.332 (1998), *H.323 extended for loosely-coupled conferences.*
- [24] ITU-T Recommendation H.450.1 (1998), *Generic functional protocol for the support of supplementary services in H.323.*
- [25] ITU-T Recommendation I.363.5 (1996), *B-ISDN ATM adaptation layer specification: Type 5 AAL.*
- [26] ITU-T Recommendation Q.2931 (1995), *Digital Subscriber Signalling System No. 2 (DSS 2) – User-network interface (UNI) layer 3 specification for basic call/connection control.*
- [27] ITU-T Recommendation I.356 (1996), *B-ISDN ATM layer cell transfer performance.*
- [28] ITU-T Recommendation I.371 (1996), *Traffic control and congestion control in B-ISDN.*
- [29] ITU-T Recommendation I.371.1 (1997), *Traffic control and congestion control in B-ISDN: conformance definitions for ABT and ABR.*

- [30] ITU-T Recommendation Q.2961.2 (1997), *Digital Subscriber Signalling System No. 2 – Additional traffic parameters: Support of ATM Transfer capability in the broadband bearer capability information element.*
- [31] ITU-T Recommendation H.282 (1999), *Remote device control protocol for multimedia applications.*
- [32] ITU-T Recommendation H.283 (1999), *Remote device control logical channel transport.*
- [33] ATM Forum Technical Committee, AF-SAA-0124.000, *Gateway for H.323 Media Transport Over ATM*, 1999.
- [34] ITU-T Recommendation Q.2941.2 (1999), *Digital Subscriber Signalling System No. 2 – Generic identifier transport extensions.*
- [35] ITU-T Recommendation H.450.2 (1998), *Call transfer supplementary service for H.323.*
- [36] ITU-T Recommendation H.450.4 (1999), *Call hold supplementary service for H.323.*

### 3 Definitions

For the purposes of this Recommendation the definitions given in clause 3/H.225.0 [1] and clause 3/H.245 [2] apply along with those in this clause. These definitions apply to the packet based network side only. Other terms may be appropriate when referring to the Switched Circuit Network (SCN) side. See clause 5, Conventions, for information on the use of terms in this Recommendation.

**3.1 active MC:** An MC that has won the master-slave determination procedure and is currently providing the multipoint control function for the conference.

**3.2 ad hoc multipoint conference:** An Ad Hoc Multipoint conference was a point-to-point conference that had been expanded into a multipoint conference at some time during the call. This can be done if one or more of the terminals in the initial point-to-point conference contains an MC, if the call is made using a Gatekeeper that includes MC functionality, or if the initial call is made through an MCU as a multipoint call between only two terminals.

**3.3 addressable:** An H.323 entity on the network having a Transport Address is addressable. Not the same as being callable. A terminal, Gateway, or MCU is addressable and callable. A Gatekeeper is addressable but not callable. An MC or MP is neither callable nor addressable but is contained within an endpoint or Gatekeeper that is.

**3.4 audio mute:** Suppressing of the audio signal of a single or all source(s). Send muting means that the originator of an audio stream mutes its microphone and/or does not transmit any audio signal at all. Receive mute means that the receiving terminal ignores a particular incoming audio stream or mutes its loudspeaker.

**3.5 broadcast conference:** A Broadcast conference is one in which there is one transmitter of media streams and many receivers. There is no bidirectional transmission of control or media streams. Such conferences should be implemented using network transport multicast facilities, if available. Also see Recommendation H.332 [23].

**3.6 broadcast panel conference:** A Broadcast Panel conference is a combination of a Multipoint conference and a Broadcast conference. In this conference, several terminals are engaged in a multipoint conference, while many other terminals are only receiving the media streams. There is bidirectional transmission between the terminals in the multipoint portion of the conference and no bidirectional transmission between them and the listening terminals. Also see Recommendation H.332.

**3.7 call:** Point-to-point multimedia communication between two H.323 endpoints. The call begins with the call set-up procedure and ends with the call termination procedure. The call consists of the collection of reliable and unreliable channels between the endpoints. A call may be directly

between two endpoints, or may include other H.323 entities such as a Gatekeeper or MC. In case of interworking with some SCN endpoints via a Gateway, all the channels terminate at the Gateway where they are converted to the appropriate representation for the SCN end system. Typically, a call is between two users for the purpose of communication, but may include signalling-only calls. An endpoint may be capable of supporting multiple simultaneous calls.

**3.8 call signalling channel:** Reliable channel used to convey the call setup and teardown messages (following Recommendation H.225.0) between two H.323 entities.

**3.9 callable:** Capable of being called as described in clause 8 or in the supplementary services Recommendations (H.450.x). In other words, an H.323 entity is generally considered callable if a user would specify the entity as a destination. Terminals, MCUs and Gateways are callable, but Gatekeepers and MCs are not.

**3.10 centralized multipoint conference:** A Centralized Multipoint conference is one in which all participating terminals communicate in a point-to-point fashion with an MCU. The terminals transmit their control, audio, video, and/or data streams to the MCU. The MC within the MCU centrally manages the conference. The MP within the MCU processes the audio, video, and/or data streams, and returns the processed streams to each terminal.

**3.11 Control and Indication (C&I):** End-to-end signalling between terminals, consisting of Control, which causes a state change in the receiver, and Indication which provides for information as to the state or functioning of the system (see also Recommendation H.245 [2] for additional information and abbreviations).

**3.12 data:** Information stream other than audio, video, and control, carried in the logical data channel (see Recommendation H.225.0 [1]).

**3.13 decentralized multipoint conference:** A Decentralized Multipoint conference is one in which the participating terminals multicast their audio and video to all other participating terminals without using an MCU. The terminals are responsible for:

- a) summing the received audio streams; and
- b) selecting one or more of the received video streams for display.

No audio or video MP is required in this case. The terminals communicate on their H.245 Control Channels with an MC which manages the conference. The data stream is still centrally processed by the MCS-MCU which may be within an MP.

**3.14 endpoint:** An H.323 terminal, Gateway, or MCU. An endpoint can call and be called. It generates and/or terminates information streams.

**3.15 gatekeeper:** The Gatekeeper (GK) is an H.323 entity on the network that provides address translation and controls access to the network for H.323 terminals, Gateways and MCUs. The Gatekeeper may also provide other services to the terminals, Gateways and MCUs such as bandwidth management and locating Gateways.

**3.16 gateway:** An H.323 Gateway (GW) is an endpoint on the network which provides for real-time, two-way communications between H.323 Terminals on the packet based network and other ITU Terminals on a switched circuit network, or to another H.323 Gateway. Other ITU Terminals include those complying with Recommendations H.310 (H.320 on B-ISDN), H.320 (ISDN), H.321 (ATM), H.322 (GQOS-LAN), H.324 (GSTN), H.324M (Mobile), and V.70 (DSVD).

**3.17 H.323 entity:** Any H.323 component, including terminals, Gateways, Gatekeepers, MCs, MPs, and MCUs.

**3.18 H.245 control channel:** Reliable Channel used to carry the H.245 control information messages (following Recommendation H.245) between two H.323 endpoints.

**3.19 H.245 session:** The part of the call that begins with the establishment of an H.245 Control Channel, and ends with the receipt of the H.245 **EndSessionCommand** or termination due to failure. Not to be confused with a call, which is delineated by the H.225.0 Setup and Release Complete messages.

**3.20 hybrid multipoint conference – centralized audio:** A Hybrid Multipoint – Centralized Audio conference is one in which terminals multicast their video to other participating terminals, and unicast their audio to the MP for mixing. The MP returns a mixed audio stream to each terminal.

**3.21 hybrid multipoint conference – centralized video:** A Hybrid Multipoint – Centralized Video conference is one in which terminals multicast their audio to other participating terminals, and unicast their video to the MP for switching or mixing. The MP returns a video stream to each terminal.

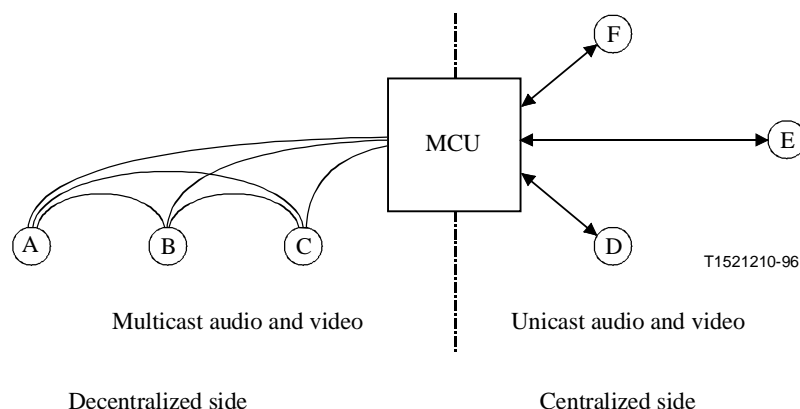
**3.22 information stream:** A flow of information of a specific media type (e.g. audio) from a single source to one or more destinations.

**3.23 lip synchronization:** Operation to provide the feeling that speaking motion of the displayed person is synchronized with his speech.

**3.24 Local Area Network (LAN):** A shared or switched medium, peer-to-peer communications network that broadcasts information for all stations to receive within a moderate-sized geographic area, such as a single office building or a campus. The network is generally owned, used, and operated by a single organization. In the context of this Recommendation, LANs also include internetworks composed of several LANs that are interconnected by bridges or routers.

**3.25 logical channel:** Channel used to carry the information streams between two H.323 endpoints. These channels are established following the H.245 **OpenLogicalChannel** procedures. An unreliable channel is used for audio, audio control, video, and video control information streams. A reliable channel is used for data and H.245 control information streams. There is no relationship between a logical channel and a physical channel.

**3.26 mixed multipoint conference:** A Mixed Multipoint conference (see Figure 2) has some terminals (D, E and F) participating in a centralized mode while other terminals (A, B and C) are participating in a decentralized mode. A terminal is not aware of the mixed nature of the conference, only of the type of conference it is participating in. The MCU provides the bridge between the two types of conferences.



**Figure 2/H.323 – Mixed multipoint conference**

**3.27 multicast:** A process of transmitting PDUs from one source to many destinations. The actual mechanism (i.e. IP multicast, multi-unicast, etc.) for this process may be different for different network technologies.



**3.28 multipoint conference:** A Multipoint conference is a conference between three or more terminals. The terminals may be on the network or on the SCN. The multipoint conference shall always be controlled by an MC. Various multipoint conference types are defined in this subclause but they all require a single MC per conference. They may also involve one or more H.231 MCUs on the SCN. A terminal on the network may also participate in an SCN multipoint conference by connecting via a Gateway to an SCN-MCU. This does not require the use of an MC.

**3.29 multipoint control unit:** The Multipoint Control Unit (MCU) is an endpoint on the network which provides the capability for three or more terminals and Gateways to participate in a multipoint conference. It may also connect two terminals in a point-to-point conference which may later develop into a multipoint conference. The MCU generally operates in the fashion of an H.231 MCU; however, an audio processor is not mandatory. The MCU consists of two parts: a mandatory Multipoint Controller and optional Multipoint Processors. In the simplest case, an MCU may consist only of an MC with no MPs. An MCU may also be brought into a conference by the Gatekeeper without being explicitly called by one of the endpoints.

**3.30 multipoint controller:** The Multipoint Controller (MC) is an H.323 entity on the network which provides for the control of three or more terminals participating in a multipoint conference. It may also connect two terminals in a point-to-point conference which may later develop into a multipoint conference. The MC provides for capability negotiation with all terminals to achieve common levels of communications. It may also control conference resources such as who is multicasting video. The MC does not perform mixing or switching of audio, video and data.

**3.31 multipoint processor:** The Multipoint Processor (MP) is an H.323 entity on the network which provides for the centralized processing of audio, video, and/or data streams in a multipoint conference. The MP provides for the mixing, switching, or other processing of media streams under the control of the MC. The MP may process a single media stream or multiple media streams depending on the type of conference supported.

**3.32 multi-unicast:** A process of transferring PDUs where an endpoint sends more than one copy of a media stream, but to different endpoints. This may be necessary in networks which do not support multicast.

**3.33 network address:** The network layer address of an H.323 entity as defined by the (inter)network layer protocol in use (e.g. an IP address). This address is mapped onto the layer one address of the respective system by some means defined in the (inter)networking protocol.

**3.34 packet based network (also network):** Any shared, switched, or point-to-point medium which provides peer-to-peer communications between two or more endpoints using a packet based transport protocol.

**3.35 point-to-point conference:** A Point-to-Point conference is a conference between two terminals. It may be either directly between two H.323 terminals or between an H.323 terminal and an SCN terminal via a Gateway. A call between two terminals (see 3.7).

**3.36 RAS channel:** Unreliable channel used to convey the registration, admissions, bandwidth change, and status messages (following Recommendation H.225.0) between two H.323 entities.

**3.37 reliable channel:** A transport connection used for reliable transmission of an information stream from its source to one or more destinations.

**3.38 reliable transmission:** Transmission of messages from a sender to a receiver using connection-mode data transmission. The transmission service guarantees sequenced, error-free, flow-controlled transmission of messages to the receiver for the duration of the transport connection.

**3.39 RTP session:** For each participant, the session is defined by a particular pair of destination Transport Addresses (one Network Address plus a TSAP identifier pair for RTP and RTCP). The destination Transport Address pair may be common for all participants, as in the case of IP multicast, or may be different for each, as in the case of individual unicast network addresses. In a

multimedia session, the media audio and video are carried in separate RTP sessions with their own RTCP packets. The multiple RTP sessions are distinguished by different Transport Addresses.

**3.40 Switched Circuit Network (SCN):** A public or private switched telecommunications network such as the GSTN, N-ISDN, or B-ISDN.

NOTE – While B-ISDN is not strictly a switched circuit network, it exhibits some of the characteristics of an SCN through the use of virtual circuits.

**3.41 terminal:** An H.323 Terminal is an endpoint on the network which provides for real-time, two-way communications with another H.323 terminal, Gateway, or Multipoint Control Unit. This communication consists of control, indications, audio, moving colour video pictures, and/or data between the two terminals. A terminal may provide speech only, speech and data, speech and video, or speech, data and video.

**3.42 transport address:** The transport layer address of an addressable H.323 entity as defined by the (inter)network protocol suite in use. The Transport Address of an H.323 entity is composed of the Network Address plus the TSAP identifier of the addressable H.323 entity.

**3.43 transport connection:** An association established by a transport layer between two H.323 entities for the transfer of data. In the context of this Recommendation, a transport connection provides reliable transmission of information.

**3.44 TSAP identifier:** The piece of information used to multiplex several transport connections of the same type on a single H.323 entity with all transport connections sharing the same Network Address, (e.g. the port number in a TCP/UDP/IP environment). TSAP identifiers may be (pre)assigned statically by some international authority or may be allocated dynamically during the setup of a call. Dynamically assigned TSAP identifiers are of transient nature, i.e. their values are only valid for the duration of a single call.

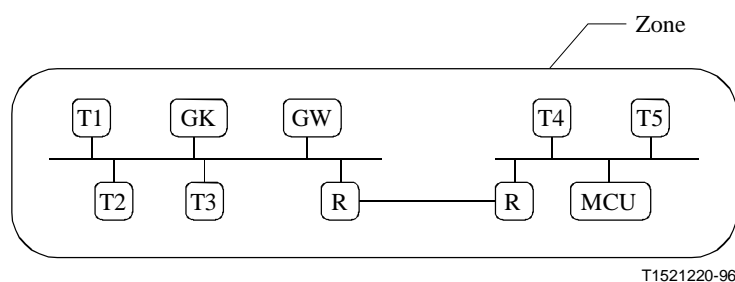
**3.45 unicast:** A process of transmitting messages from one source to one destination.

**3.46 unreliable channel:** A logical communication path used for unreliable transmission of an information stream from its source to one or more destinations.

**3.47 unreliable transmission:** Transmission of messages from a sender to one or more receivers by means of connectionless-mode data transmission. The transmission service is *best-effort* delivery of the PDU, meaning that messages transmitted by the sender may be lost, duplicated, or received out of order by (any of) the receiver(s).

**3.48 well-known TSAP identifier:** A TSAP identifier that has been allocated by an (international) authority that is in charge of the assignment of TSAP identifiers for a particular (inter)networking protocol and the related transport protocols (e.g. the IANA for TCP and UDP port numbers). This identifier is guaranteed to be unique in the context of the respective protocol.

**3.49 zone:** A Zone (see Figure 3) is the collection of all terminals (Tx), Gateways (GW), and Multipoint Control Units (MCUs) managed by a single Gatekeeper (GK). A Zone includes at least one terminal, and may or may not include Gateways or MCUs. A Zone has one and only one Gatekeeper. A Zone may be independent of network topology and may be comprised of multiple network segments which are connected using routes (R) or other devices.



**Figure 3/H.323 – Zone**

#### **4 Symbols and abbreviations**

This Recommendation uses the following abbreviations:

4CIF	4 times CIF
16CIF	16 times CIF
ABR	Available Bit Rate
ABT/DT	ATM Block Transfer/Delayed Transmission
ABT/IT	ATM Block Transfer/Immediate Transmission
ACF	Admission Confirmation
APE	Application Protocol Entity
ARJ	Admission Reject
ARQ	Admission Request
ATC	ATM Transfer Capability
ATM	Asynchronous Transfer Mode
BAS	Bit rate Allocation Signal
BCF	Bandwidth Change Confirmation
BCH	Bose, Chaudhuri, and Hocquengham
B-HLI	Broadband High Layer Information
B-ISDN	Broadband Integrated Services Digital Network
BRJ	Bandwidth Change Reject
BRQ	Bandwidth Change Request
BTC	Broadband Transfer Capability
C&I	Control and Indication
CBR	Constant Bit Rate
CDV	Cell Delay Variation
CER	Cell Error Ratio
CID	Conference Identifier
CIF	Common Intermediate Format
CLR	Cell Loss Ratio

CMR	Cell misinsertion rate
CTD	Cell Transfer Delay
DBR	Deterministic Bit Rate
DCF	Disengage Confirmation
DID	Direct Inward Dialling
DRQ	Disengage Request
DSVD	Digital Simultaneous Voice and Data
DTMF	Dual-Tone MultiFrequency
ECS	Encryption Control Signal
EIV	Encryption Initialization Vector
FAS	Frame Alignment Signal
FIR	Full Intra Request
GCC	Generic Conference Control
GCF	Gatekeeper Confirmation
GIT	Generic Identifier Transport
GK	Gatekeeper
GQOS	Guaranteed Quality of Service
GRJ	Gatekeeper Reject
GRQ	Gatekeeper Request
GSTN	General Switched Telephone Network
GW	Gateway
IACK	Information Acknowledgment
IANA	Internet Assigned Number Authority
IE	Information Element
INAK	Information Negative Acknowledgment
IP	Internet Protocol
IPX	Internetwork Protocol Exchange
IRQ	Information Request
IRR	Information Request Response
ISDN	Integrated Services Digital Network
ITU-T	International Telecommunication Union – Telecommunication Standardization Sector
LAN	Local Area Network
LCF	Location Confirmation
LCN	Logical Channel Number
LRJ	Location Reject
LRQ	Location Request
MC	Multipoint Controller

MCS	Multipoint Communications System
MCU	Multipoint Control Unit
MP	Multipoint Processor
MPEG	Motion Picture Experts Group
MSN	Multiple Subscriber Number
MTU	Maximum Transport Unit
MTU	Maximum Transmission Unit
N-ISDN	Narrow-band Integrated Services Digital Network
NACK	Negative Acknowledge
NSAP	Network Layer Service Access Point
OLC	H.245 <b>openLogicalChannel</b> message
PBN	Packet Based Network
PDU	Packet Data Unit
PPP	Point-to-Point Protocol
QCIF	Quarter CIF
QOS	Quality of Service
RAS	Registration, Admission and Status
RAST	Receive and Send Terminal
RCF	Registration Confirmation
RIP	Request in Progress
RRJ	Registration Reject
RRQ	Registration Request
RTCP	Real Time Control Protocol
RTP	Real Time Protocol
SBE	Single Byte Extension
SBR1	Statistical Bit Rate configuration 1
SBR2	Statistical Bit Rate configuration 2
SBR3	Statistical Bit Rate configuration 3
SCM	Selected Communications Mode
SCN	Switched Circuit Network
SECBR	Severely Errored Cell Block Ratio
SPX	Sequential Protocol Exchange
SQCIF	Sub QCIF
SSRC	Synchronization Source Identifier
TCP	Transport Control Protocol
TSAP	Transport layer Service Access Point
UCF	Unregister Confirmation

UDP	User Datagram Protocol
URJ	Unregister Reject
URQ	Unregister Request
VBR	Variable Bit Rate
VC	Virtual Channel
VC	Virtual Circuit

## 5 Conventions

In this Recommendation, the following conventions are used:

"Shall" indicates a mandatory requirement.

"Should" indicates a suggested but optional course of action.

"May" indicates an optional course of action rather than a recommendation that something take place.

References to clauses, subclauses, annexes and appendices refer to those items within this Recommendation unless another specification is explicitly listed. For example, 1.4 refers to 1.4 of this Recommendation; 6.4/H.245 refers to 6.4 in Recommendation H.245.

Throughout this Recommendation, the term "network" is used to indicate any packet based network regardless of the underlying physical connection or the geographic scope of the network. This includes Local Area Networks, internetworks, and other packet based networks. The term "Switched Circuit Network" or "SCN" is used explicitly when referring to switched circuit networks such as GSTN and ISDN.

Where items exist on both the packet based network and the SCN, references to the SCN item will be explicit. For example, an MCU is an H.323 MCU on the packet based network, an SCN-MCU is an MCU on the SCN.

This Recommendation describes the use of three different message types: H.245, RAS and Q.931. To distinguish between the different message types, the following convention is followed. H.245 message and parameter names consist of multiple concatenated words highlighted in bold typeface (**maximumDelayJitter**). RAS message names are represented by three letter abbreviations (ARQ). Q.931 message names consist of one or two words with the first letters capitalized (Call Proceeding).

## 6 System description

This Recommendation describes the elements of the H.323 components. These are Terminals, Gateways, Gatekeepers, MCs and MCUs. These components communicate through the transmission of Information Streams. The characteristics of these components are described in this clause.

### 6.1 Information streams

Visual telephone components communicate through the transmission of Information Streams. These Information Streams are classified into video, audio, data, communications control and call control as follows.

Audio signals contain digitized and coded speech. In order to reduce the average bit rate of audio signals, voice activation may be provided. The audio signal is accompanied by an audio control signal.

Video signals contain digitized and coded motion video. Video is transmitted at a rate no greater than that selected as a result of the capability exchange. The video signal is accompanied by a video control signal.

Data signals include still pictures, facsimile, documents, computer files and other data streams.

Communications control signals pass control data between remote-like functional elements and are used for capability exchange, opening and closing logical channels, mode control and other functions that are part of communications control.

Call control signals are used for call establishment, disconnect and other call control functions.

The information streams described above are formatted and sent to the network interface as described in Recommendation H.225.0.

## 6.2 Terminal characteristics

An example of an H.323 terminal is shown in Figure 4. The diagram shows the user equipment interfaces, video codec, audio codec, telematic equipment, H.225.0 layer, system control functions and the interface to the packet based network. All H.323 terminals shall have a System Control Unit, H.225.0 layer, Network Interface and an Audio Codec Unit. The Video Codec Unit and User Data Applications are optional.

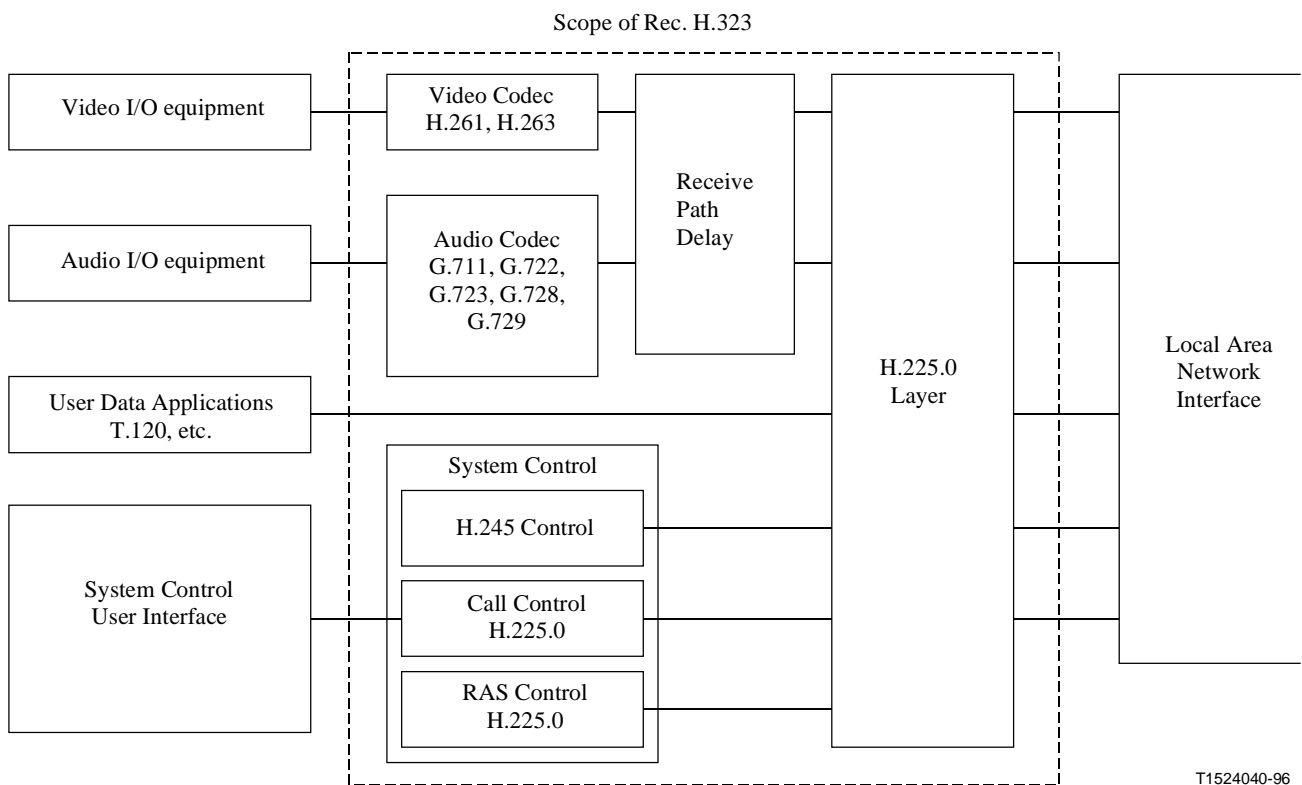


Figure 4/H.323 – H.323 terminal equipment

T1524040-96

### **6.2.1 Terminal elements outside the scope of this Recommendation**

The following elements are not within the scope of this Recommendation and are therefore not defined within this Recommendation:

- Attached audio devices providing voice activation sensing, microphone and loudspeaker, telephone instrument or equivalent, multiple microphones mixers, and acoustic echo cancellation.
- Attached video equipment providing cameras and monitors, and their control and selection, video processing to improve compression or provide split screen functions.
- Data applications and associated user interfaces which use T.120 or other data services over the data channel.
- Attached Network Interface, which provides the interface to the packet based network, supporting appropriate signalling and voltage levels, in accordance with national and international standards.
- Human user system control, user interface and operation.

### **6.2.2 Terminal elements within the scope of this Recommendation**

The following elements are within the scope of this Recommendation, and are therefore subject to standardization and are defined within this Recommendation:

- The Video Codec (H.261, etc.) encodes the video from the video source (i.e. camera) for transmission and decodes the received video code which is output to a video display.
- The Audio Codec (G.711, etc.) encodes the audio signal from the microphone for transmission and decodes the received audio code which is output to the loudspeaker.
- The Data Channel supports telematic applications such as electronic whiteboards, still image transfer, file exchange, database access, audiographics conferencing, etc. The standardized data application for real-time audiographics conferencing is Recommendation T.120. Other applications and protocols may also be used via H.245 negotiation as specified in 6.2.7.
- The System Control Unit (H.245, H.225.0) provides signalling for proper operation of the H.323 terminal. It provides for call control, capability exchange, signalling of commands and indications, and messages to open and fully describe the content of logical channels.
- H.225.0 Layer (H.225.0) formats the transmitted video, audio, data and control streams into messages for output to the network interface and retrieves the received video, audio, data, and control streams from messages which have been input from the network interface. In addition, it performs logical framing, sequence numbering, error detection and error correction as appropriate to each media type.

### **6.2.3 Packet based network interface**

The packet based network interface is implementation-specific and is outside the scope of this Recommendation. However, the network interface shall provide the services described in Recommendation H.225.0. This includes the following: Reliable (e.g. TCP, SPX) end-to-end service is mandatory for the H.245 Control Channel, the Data Channels, and the Call Signalling Channel. Unreliable (e.g. UDP, IPX) end-to-end service is mandatory for the Audio Channels, the Video Channels, and the RAS Channel. These services may be duplex or simplex, unicast or multicast depending on the application, the capabilities of the terminals, and the configuration of the network.



## 6.2.4 Video codec

The video codec is optional. If video capability is provided, it shall be provided according to the requirements of this Recommendation. All H.323 terminals providing video communications shall be capable of encoding and decoding video according to H.261 QCIF. Optionally, a terminal may also be capable of encoding and decoding video according to the other modes of H.261 or H.263. If a terminal supports H.263 with CIF or higher resolution, it shall also support H.261 CIF. All terminals which support H.263 shall support H.263 QCIF. The H.261 and H.263 codecs, on the network, shall be used without BCH error correction and without error correction framing.

Other video codecs, and other picture formats, may also be used via H.245 negotiation. More than one video channel may be transmitted and/or received, as negotiated via the H.245 Control Channel. The H.323 terminal may optionally send more than one video channel at the same time, for example, to convey the speaker and a second video source. The H.323 terminal may optionally receive more than one video channel at the same time, for example, to display multiple participants in a distributed multipoint conference.

The video bit rate, picture format and algorithm options that can be accepted by the decoder are defined during the capability exchange using H.245. The encoder is free to transmit anything that is within the decoder capability set. The decoder should have the possibility to generate requests via H.245 for a certain mode, but the encoder is allowed to simply ignore these requests if they are not mandatory modes. Decoders which indicate capability for a particular algorithm option shall also be capable of accepting video bit streams which do not make use of that option.

H.323 terminals shall be capable of operating in asymmetric video bit rates, frame rates, and, if more than one picture resolution is supported, picture resolutions. For example, this will allow a CIF capable terminal to transmit QCIF while receiving CIF pictures.

When each video logical channel is opened, the selected operating mode to be used on that channel is signalled to the receiver in the H.245 **openLogicalChannel** message. The header within the video logical channel indicates which mode is actually used for each picture, within the stated decoder capability.

The video stream is formatted as described in Recommendation H.225.0.

### 6.2.4.1 Terminal-based continuous presence

H.323 terminals may receive more than one video channel, particularly for multipoint conferencing. In these cases, the H.323 terminal may need to perform a video mixing or switching function in order to present the video signal to the user. This function may include presenting the video from more than one terminal to the user. The H.323 terminal shall use H.245 simultaneous capabilities to indicate how many simultaneous video streams it is capable of decoding. The simultaneous capability of one terminal should not limit the number of video streams which are multicast in a conference (this choice is made by the MC).

## 6.2.5 Audio codec

All H.323 terminals shall have an audio codec. All H.323 terminals shall be capable of encoding and decoding speech according to Recommendation G.711. All terminals shall be capable of transmitting and receiving A-law and  $\mu$ -law. A terminal may optionally be capable of encoding and decoding speech using other audio codecs which can be signalled via H.245 negotiation. The audio algorithm used by the encoder shall be derived during the capability exchange using H.245. The H.323 terminal should be capable of asymmetric operation for all audio capabilities it has declared within the same capability set, e.g. it should be able to send G.711 and receive G.728 if it is capable of both.

If G.723.1 audio is provided, the audio codec shall be capable of encoding and decoding according to both the 5.3 kbit/s mode and the 6.3 kbit/s mode.

The audio stream is formatted as described in Recommendation H.225.0.

The H.323 terminal may optionally send more than one audio channel at the same time, for example, to allow two languages to be conveyed.

Audio packets should be delivered to the transport layer periodically at an interval determined by the audio codec Recommendation in use (audio frame interval). The delivery of each audio packet shall occur no later than 5 ms after a whole multiple of the audio frame interval, measured from delivery of the first audio frame (audio delay jitter). Audio coders capable of further limiting their audio delay jitter may so signal using the H.245 **maximumDelayJitter** parameter of the **h2250Capability** structure contained within a terminal capability set message, so that receivers may optionally reduce their jitter delay buffers. This is not the same as the RTCP interarrival jitter field.

NOTE – The testing point for the maximum delay jitter is at the input to network transport layer. Network stack, network, driver, and interface card jitter are not included.

#### **6.2.5.1 Audio mixing**

H.323 terminals may receive more than one audio channel, particularly for multipoint conferencing. In these cases, the H.323 terminal may need to perform an audio mixing function in order to present a composite audio signal to the user. The H.323 terminal shall use H.245 simultaneous capabilities to indicate how many simultaneous audio streams it is capable of decoding. The simultaneous capability of one terminal should not limit the number of audio streams which are multicast in a conference.

#### **6.2.5.2 Maximum audio-video transmit skew**

To allow H.323 terminals to appropriately set their receive buffer(s) size, H.323 terminals shall transmit the **h2250MaximumSkewIndication** message to indicate the maximum skew between the audio and video signals as delivered to the network transport. **h2250MaximumSkewIndication** shall be sent for each pair of associated audio and video logical channels. This is not required for audio only or hybrid conferences. Lip synchronization, if desired, shall be achieved via use of time-stamps.

#### **6.2.5.3 Low bit rate operation**

G.711 audio cannot be used in an H.323 conference being carried over low bit rate (< 56 kbit/s) links or segments. An endpoint used for multimedia communications over such low bit rate links or segments should have an audio codec capable of encoding and decoding speech according to Recommendation G.723.1. An endpoint used for audio-only communications over such low bit rate links or segments should have an audio codec capable of encoding and decoding speech according to Recommendation G.729. An endpoint may support several audio codecs in order to provide the widest possible interoperability with those endpoints which only support one low bitrate audio codec. The endpoint shall indicate in the H.245 Capability Exchange procedures at the beginning of each call the capability to receive audio according to the available audio Recommendations which can be supported within the known bit rate limitations of the connection. An endpoint which does not have this low bit rate audio capability may not be able to operate when the end-to-end connection contains one or more low bit rate segments.

The endpoint shall also comply with the requirement of 6.2.5 to be capable of encoding and decoding speech according to Recommendation G.711. However, the endpoint need not indicate this capability if it is sure that it is communicating through a low bit rate segment. If an endpoint is unaware of the presence, in the end-to-end connection, of any links or segments with insufficient capacity to support G.711 audio (along with other intended media streams, if any), then the endpoint shall declare the capability to receive audio according to Recommendation G.711.

## 6.2.6 Receive path delay

Receive path delay includes delay added to a media stream in order to maintain synchronization and to account for network packet arrival jitter. Media streams may optionally be delayed in the receiver processing path to maintain synchronization with other media streams. Further, a media stream may optionally be delayed to allow for network delays which cause packet arrival jitter. An H.323 terminal shall not add delay for this purpose in its transmitting media path.

Intermediate processing points such as MCUs or Gateways may alter the video and audio time tag information, and shall transmit appropriately modified audio and video time tags and sequence numbers, reflecting their transmitted signals. Receiving endpoints may add appropriate delay in the audio path to achieve lip synchronization.

## 6.2.7 Data channel

One or more data channels are optional. The data channel may be unidirectional or bidirectional depending on the requirements of the data application.

Recommendation T.120 is the default basis of data interoperability between an H.323 terminal and other H.323, H.324, H.320, or H.310 terminals. Where any optional data application is implemented using one or more of the ITU-T Recommendations which can be negotiated via H.245, the equivalent T.120 application, if any, shall be one of those provided.

Note that non-standard data applications (**dataApplicationCapability.application = non-standard application**) and Transparent User Data (**dataApplicationCapability.application = userData application, dataProtocolCapability = transparent**) may be used whether the equivalent T.120 application is provided or not.

T.120 capability shall be signalled using **dataApplicationCapability.application = t120 application, dataProtocolCapability = separateLANStack**.

Within the **MediaDistributionCapability**, the **distributedData** structure shall be used if multicast T.120 is available and/or the **centralizedData** structure if unicast T.120 is available. Any node that supports the T.120 data capability shall support the standard T.123 unicast stack.

In the **openLogicalChannel** message, the **distribution** choice of the **NetworkAccessParameters** structure is set to **unicast** if T.123 is to be used or **multicast** if Annex A/T.125 is to be used. The **networkAddress** choice is set to **localAreaAddress**, which should always be **unicastAddress**. Within the **IPAddress** sequence, the **network** field is set to the binary IP address and the **tsapIdentifier** is set to the dynamic port on which the T.120 stack will be calling or listening.

The Data channel is formatted as described in Recommendation H.225.0.

### 6.2.7.1 T.120 data channels

The T.120 connection is established during an H.323 call as an inherent part of the call. Procedures for establishing the T.120 connection prior to the H.323 connection are for further study.

The normal call setup procedures of 8.1 are followed. After the capability exchange takes place, a bidirectional logical channel shall be opened for the T.120 connection according to the normal H.245 procedures indicating that a new connection shall be created as described below.

The opening of a bidirectional logical channel for T.120 may be initiated by either entity sending an **openLogicalChannel** message, and then following the bidirectional logical channel procedures of Recommendation H.245.

To actually open the logical channel, the initiating entity shall send an **openLogicalChannel** message indicating that a T.120 data channel is to be opened in the **forwardLogicalChannelParameters** as well as in the **reverseLogicalChannelParameters**. The initiator shall include a Transport Address in the **openLogicalChannel** message. The peer endpoint may choose to ignore the Transport Address. An endpoint may use a dynamic port number for the T.120 Transport Address instead of using port 1503

as specified in Recommendation T.123. If the peer (the responder) accepts this logical channel, it shall confirm the opening of the logical channel using **openLogicalChannelAck**. In the **openLogicalChannelAck**, the responder shall include a Transport Address even if it expects the initiator to originate the T.120 call. In all cases, the Transport Address for the T.120 connection shall be carried in the **separateStack** parameter, and shall remain valid for the duration of the logical channel.

In the **openLogicalChannel** message, the **t120SetupProcedure** choice of the **NetworkAccessParameters** structure can optionally be set to indicate to the responder how the initiator would like to establish the T.120 call. The responder is free to override this preference. **originateCall** indicates that the initiator would like the responder to place the call. **waitForCall** indicates that the initiator would like the responder to receive the call. **issueQuery** is not used when indicating a preference.

In the **openLogicalChannelAck** message, the **t120SetupProcedure** choice of the **NetworkAccessParameters** structure should be set to indicate to the initiator how the T.120 call will be established. If neither endpoint has a preference, the T.120 call should be established in the same direction as the H.323 call. **originateCall** tells the initiator to place the call. **waitForCall** tells the initiator that it will receive the call. Whoever originates the call will issue either a join request or an invite request, depending on which endpoint won master/slave determination (the master is always hierarchically higher in the T.120 conference). **issueQuery** can be used by a Gateway to tell the initiator that it must originate the call, and issue a query request to the remote endpoint. It must then set up the T.120 conference in accordance with the contents of the query response (as described in Recommendation T.124).

When possible, the T.120 call should be established in the same direction as the H.323 call. The OLC initiator should not indicate a preference unless there is a need to modify this default behaviour. When the initiator indicates a preference, the responder should not override it unless necessary. When no preference is indicated, the responder should specify the default unless there is a need to do otherwise.

In both the **openLogicalChannel** and the **openLogicalChannelAck** messages, the **associateConference** parameter shall be set to FALSE.

Recommendation T.120 shall follow the procedures of Recommendation T.123 for the protocol stack indicated in the **dataProtocolCapability** except that the Transport Addresses as described above shall be employed for connection setup.

If an endpoint is the Active MC or master in a conference which includes T.120, it should also be in control of the T.120 top provider node.

If an endpoint intends to create a conference which includes audio and/or video plus T.120 data, then the H.245 Control Channel shall be established before the T.120 connection is made. This applies to conference create, join, and invite and the actions of an MC. The H.323 call setup procedures shall be used to establish the Active MC (if any), before a T.120 connection is made.

In order to establish a T.120 connection using a GCC-Join request, endpoints are required to know the T.120 conference name. If an alias exists which represents an H.323 conference name (**conferenceAlias**), then the same text which is used for the conference alias should be used as the text portion of the T.120 conference name. Likewise, the H.323 CID should be used as the numeric T.120 conference name as follows. Each byte of the H.323 CID is converted into a series of three ASCII characters which represent the decimal value of the byte being converted. Note that this requires the value of some CID bytes to be converted such that "0" characters are used for padding. The result will be a string of 48 ASCII characters.

A T.120 MP may be queried for a list of existing conferences. The H.323 CID may be available by converting from the T.120 Numeric Conference name back into the 16-byte octet string. Likewise, the Text Conference name may be used as the H.323 conference alias. Note that a T.124 Conference Query may happen out-of-band from H.323 and prior to an endpoint setting up an H.323 call.

The termination of the associated T.120 conference does not imply the termination of the H.323 call. In other words, closing the T.120 channel shall only affect the Data stream of an H.323 call and shall not affect any other part of the H.323 call. By contrast, when an H.323 call or conference is terminated, then the associated T.120 conference shall also be terminated.

NOTE – The T.120 operation after completion of the connection setup is beyond the scope of this Recommendation.

### **6.2.7.2 Remote Device Control**

H.323 endpoints may support remote device control through the H.282 protocol. The H.282 protocol shall be supported in an H.245 logical channel according to Recommendation H.283. Recommendation H.283 describes logical channel transport for the H.282 protocol in an H.323 conference.

Recommendation H.282 may also be used by T.120 systems and carried in a T.120 APE. Optionally H.323 systems may also support remote device control using Recommendation H.282 over T.120. However this is an option and an H.323 system that supports H.282 shall support it with Recommendation H.283.

If both H.282 with H.283 and H.282 with T.120 are supported, then both may be used. Coordination of the two lower layer protocols under H.282 is a local matter. However, H.283 shall always be active to account for possible late joining nodes that support H.282 over H.283 but not H.282 over T.120.

### **6.2.8 H.245 control function**

The H.245 Control Function uses the H.245 Control Channel to carry end-to-end control messages governing operation of the H.323 entity, including capabilities exchange, opening and closing of logical channels, mode preference requests, flow control messages, and general commands and indications.

H.245 signalling is established between two endpoints, an endpoint and an MC, or an endpoint and a Gatekeeper. The endpoint shall establish exactly one H.245 Control Channel for each call that the endpoint is participating in. This channel shall use the messages and procedures of Recommendation H.245. Note that a terminal, MCU, Gateway, or Gatekeeper may support many calls, and thus many H.245 Control Channels. The H.245 Control Channel shall be carried on logical channel 0. Logical channel 0 shall be considered to be permanently open from the establishment of the H.245 Control Channel until the termination of this channel. The normal procedures for opening and closing logical channels shall not apply to the H.245 Control Channel.

Recommendation H.245 specifies a number of independent protocol entities which support endpoint-to-endpoint signalling. A protocol entity is specified by its syntax (messages), semantics, and a set of procedures which specify the exchange of messages and the interaction with the user. H.323 endpoints shall support the syntax, semantics, and procedures of the following protocol entities:

- Master/slave determination.
- Capability Exchange.
- Logical Channel Signalling.
- Bidirectional Logical Channel Signalling.
- Close Logical Channel Signalling.
- Mode Request.
- Round Trip Delay Determination.
- Maintenance Loop Signalling.

General commands and indications shall be chosen from the message set contained in Recommendation H.245. In addition, other command and indication signals may be sent which have been specifically defined to be transferred in-band within video, audio or data streams (see the appropriate Recommendation to determine if such signals have been defined).

H.245 messages fall into four categories: Request, Response, Command, and Indication. Request and Response messages are used by the protocol entities. Request messages require a specific action by the receiver, including an immediate response. Response messages respond to a corresponding request. Command messages require a specific action, but do not require a response. Indication messages are informative only, and do not require any action or response. H.323 terminals shall respond to all H.245 commands and requests as specified in Annex A, and shall transmit indications reflecting the state of the terminal.

H.323 terminals shall be capable of passing all H.245 **multimediaSystemControlMessage** messages, and shall send and receive all messages needed to implement required functions and those optional functions which are supported by the terminal. Annex A contains a table showing which H.245 messages are mandatory, optional, or forbidden for H.323 terminals. H.323 terminals shall send the **functionNotSupported** message in response to any unrecognized request, response, or command messages that it receives.

An H.245 indication, **userInputIndication**, is available for transport of user input alphanumeric characters from a keypad or keyboard, equivalent to the DTMF signals used in analogue telephony, or SBE number messages in Recommendation H.230. This may be used to manually operate remote equipment such as voice mail or video mail systems, menu-driven information services, etc. H.323 terminals shall support the transmission of user input characters 0-9, "\*", and "#". Transmission of other characters is optional.

Three H.245 request messages conflict with RTCP control packets. The H.245 **videoFastUpdatePicture**, **videoFastUpdateGOB** and **videoFastUpdateMB** requests should be used instead of the RTCP control packets Full Intra Request (FIR) and Negative Acknowledgement (NACK). The ability to accept FIR and NACK is signalled during the H.245 capability exchange.

### 6.2.8.1 Capabilities exchange

Capabilities exchange shall follow the procedures of Recommendation H.245, which provides for separate receive and transmit capabilities, as well as a method by which the terminal may describe its ability to operate in various combinations of modes simultaneously.

Receive capabilities describe the terminal's ability to receive and process incoming information streams. Transmitters shall limit the content of their transmitted information to that which the receiver has indicated it is capable of receiving. The absence of a receive capability indicates that the terminal cannot receive (is a transmitter only).

Transmit capabilities describe the terminal's ability to transmit information streams. Transmit capabilities serve to offer receivers a choice of possible modes of operation, so that the receiver may request the mode which it prefers to receive. The absence of a transmit capability indicates that the terminal is not offering a choice of preferred modes to the receiver (but it may still transmit anything within the capability of the receiver).

The transmitting terminal assigns each individual mode the terminal is capable of operating in a number in a **capabilityTable**. For example, G.723.1 audio, G.728 audio, and CIF H.263 video would each be assigned separate numbers.

These capability numbers are grouped into **alternativeCapabilitySet** structures. Each **alternativeCapabilitySet** indicates that the terminal is capable of operating in exactly one mode listed in the set. For example, an **alternativeCapabilitySet** listing {G.711, G.723.1, G.728} means that the terminal can operate in any one of those audio modes, but not more than one.

These **alternativeCapabilitySet** structures are grouped into **simultaneousCapabilities** structures. Each **simultaneousCapabilities** structure indicates a set of modes the terminal is capable of using simultaneously. For example, a **simultaneousCapabilities** structure containing the two **alternativeCapabilitySet** structures {H.261, H.263} and {G.711, G.723.1, G.728} means that the terminal can operate either of the video codecs simultaneously with any one of the audio codecs. The **simultaneousCapabilities** set { {H.261}, {H.261, H.263}, {G.711, G.723.1, G.728} } means the terminal can operate two video channels and one audio channel simultaneously: one video channel per H.261, another video channel per either H.261 or H.263, and one audio channel per either G.711, G.723.1, or G.728.

NOTE – The actual capabilities stored in the **capabilityTable** are often more complex than presented here. For example, each H.263 capability indicates details including the ability to support various picture formats at given minimum picture intervals, and the ability to use optional coding modes. For a complete description, see Recommendation H.245.

The terminal's total capabilities are described by a set of **capabilityDescriptor** structures, each of which is a single **simultaneousCapabilities** structure and a **capabilityDescriptorNumber**. By sending more than one **capabilityDescriptor**, the terminal may signal dependencies between operating modes by describing different sets of modes which it can simultaneously use. For example, a terminal issuing two **capabilityDescriptor** structures, one { {H.261, H.263}, {G.711, G.723.1, G.728} } as in the previous example, and the other { {H.262}, {G.711} }, means the terminal can also operate the H.262 video codec, but only with the low-complexity G.711 audio codec.

Terminals may dynamically add capabilities during a communication session by issuing additional **capabilityDescriptor** structures, or remove capabilities by sending revised **capabilityDescriptor** structures. All H.323 terminals shall transmit at least one **capabilityDescriptor** structure.

Non-standard capabilities and control messages may be issued using the **nonStandardParameter** structure defined in Recommendation H.245. Note that while the meaning of non-standard messages is defined by individual organizations, equipment built by any manufacturer may signal any non-standard message, if the meaning is known.

Terminals may re-issue capability sets at any time, according to the procedures of Recommendation H.245.

### 6.2.8.2 Logical channel signalling

Each logical channel carries information from a transmitter to one or more receivers, and is identified by a logical channel number which is unique for each direction of transmission.

Logical channels are opened and closed using the **openLogicalChannel** and **closeLogicalChannel** messages and procedures of Recommendation H.245. When a logical channel is opened, the **openLogicalChannel** message fully describes the content of the logical channel, including media type, algorithm in use, any options, and all other information needed for the receiver to interpret the content of the logical channel. Logical channels may be closed when no longer needed. Open logical channels may be inactive, if the information source has nothing to send.

Most logical channels in this Recommendation are unidirectional, so asymmetrical operation, in which the number and type of information streams is different in each direction of transmission, is allowed. However, if a receiver is capable only of certain symmetrical modes of operation, it may send a receive capability set that reflects its limitations, except where noted elsewhere in this Recommendation. Terminals may also be capable of using a particular mode in only one direction of transmission. Certain media types, including data protocols such as T.120, inherently require a bidirectional channel for their operation. In such cases a pair of unidirectional logical channels, one in each direction, may be opened and associated together to form a bidirectional channel using the bidirectional channel opening procedures of Recommendation H.245. Such pairs of associated channels need not share the same logical channel number, since logical channel numbers are independent in each direction of transmission.

Logical channels shall be opened using the following procedure:

The initiating terminal shall send an **openLogicalChannel** message as described in Recommendation H.245. If the logical channel is to carry a media type using RTP (audio or video), the **openLogicalChannel** message shall include the **mediaControlChannel** parameter containing the Transport Address for the reverse RTCP channel.

The responding terminal shall respond with an **openLogicalChannelAck** message as described in Recommendation H.245. If the logical channel is to carry a media type using RTP, the **openLogicalChannelAck** message shall include both the **mediaTransportChannel** parameter containing the RTP Transport Address for the media channel and the **mediaControlChannel** parameter containing the Transport Address for the forward RTCP channel.

Media types (such as T.120 data) which do not use RTP/RTCP shall omit the **mediaControlChannel** parameters.

If a corresponding reverse channel is opened for a given existing RTP session (identified by the RTP **sessionID**), the **mediaControlChannel** Transport Addresses exchanged by the **openLogicalChannel** process shall be identical to those used for the forward channel. Should a collision occur where both ends attempt to establish conflicting RTP sessions at the same time, the master endpoint shall reject the conflicting attempt as described in Recommendation H.245. The rejected **openLogicalChannel** attempt may then be retried at a later time.

### 6.2.8.3 Mode preferences

Receivers may request transmitters to send a particular mode using the H.245 **requestMode** message, which describes the desired mode. Transmitters should comply if possible.

An endpoint receiving the **multipointModeCommand** from the MC shall then comply with all **requestMode** commands, if they are within its capability set. Note that in a decentralized conference, as in a centralized conference, all terminal **requestMode** commands are directed to the MC. The MC may grant the request or not; the basis for this decision is left to the manufacturer.

### 6.2.8.4 Master-slave determination

The H.245 Master-slave determination procedures are used to resolve conflicts between two endpoints which can both be the MC for a conference, or between two endpoints which are attempting to open a bidirectional channel. In this procedure, two endpoints exchange random numbers in the H.245 **masterSlaveDetermination** message, to determine the master and slave endpoints. H.323 endpoints shall be capable of operating in both master and slave modes. The endpoints shall set **terminalType** to the value specified in Table 1 below and set **statusDeterminationNumber** to a random number in the range 0 to  $2^{24}-1$ . Only one random number shall be chosen by the endpoint for each call, except in the case of identical random numbers, as described in Recommendation H.245.

**Table 1/H.323 – H.323 terminal types for H.245 master-slave determination**

TerminalType value table	H.323 entity			
	Terminal	Gateway	Gatekeeper	MCU
Entity with No MC	50	60	NA	NA
Entity contains an MC but no MP	70	80	120	160
Entity contains MC with data MP	NA	90	130	170
Entity contains MC with data and audio MP	NA	100	140	180
Entity contains MC with data, audio and video MP	NA	110	150	190

The Active MC in a conference shall use a value of 240.



If a single H.323 entity can take part in multiple calls, then the value used for **terminalType** in the master-slave determination process shall be based on the features that the H.323 entity has assigned or will assign to the call in which it is being signalled.

An MC that is already acting as an MC shall always remain the Active MC. Therefore, once an MC has been selected as the Active MC in a conference, it shall use the Active MC value for all subsequent connections to the conference.

If no MC is active and the entities are of the same type, then the H.323 entity with the highest feature set (as shown in Table 1) shall win the master-slave determination. If no MC is active and the entities are of different types, then an MC that is located in an MCU shall have priority over an MC that is located in a Gatekeeper, which shall have priority over an MC that is located in a Gateway, which in turn shall have priority over an MC located in a terminal.

If an H.323 entity can be associated with two or more of the classifications shown in Table 1, then it should use the highest value for which it qualifies.

#### **6.2.8.5 Timer and counter values**

All timers defined in Recommendation H.245 should have periods of at least as long as the maximum data delivery time allowed by the data link layer carrying the H.245 Control Channel, including any retransmissions.

The H.245 retry counter N100 should be at least 3.

Procedures relating to H.245 protocol error handling are covered in 8.6.

#### **6.2.9 RAS signalling function**

The RAS signalling function uses H.225.0 messages to perform registration, admissions, bandwidth changes, status, and disengage procedures between endpoints and Gatekeepers. The RAS Signalling Channel is independent from the Call Signalling Channel and the H.245 Control Channel. H.245 open logical channel procedures are not used to establish the RAS Signalling Channel. In network environments that do not have a Gatekeeper, the RAS Signalling Channel is not used. In network environments which contain a Gatekeeper (a Zone), the RAS Signalling Channel is opened between the endpoint and the Gatekeeper. The RAS Signalling Channel is opened prior to the establishment of any other channels between H.323 endpoints. This channel is described in detail in clause 7.

#### **6.2.10 Call signalling function**

The call signalling function uses H.225.0 call signalling to establish a connection between two H.323 endpoints. The Call Signalling Channel is independent from the RAS Channel and the H.245 Control Channel. H.245 open logical channel procedures are not used to establish the Call Signalling Channel. The Call Signalling Channel is opened prior to the establishment of the H.245 Channel and any other logical channels between H.323 endpoints. In systems that do not have a Gatekeeper, the Call Signalling Channel is opened between the two endpoints involved in the call. In systems which contain a Gatekeeper, the Call Signalling Channel is opened between the endpoint and the Gatekeeper, or between the endpoints themselves as chosen by the Gatekeeper. This channel is described in detail in clause 7.

#### **6.2.11 H.225.0 layer**

Logical channels of video, audio, data or control information are established according to the procedures of Recommendation H.245. Logical channels are unidirectional, and are independent in each direction of transmission. Some logical channels, such as for data, may be bidirectional, and are associated through the bidirectional open logical channel procedure of Recommendation H.245. Any number of logical channels of each media type may be transmitted, except for the H.245 Control Channel of which there shall be one per call. In addition to the logical channels, H.323 endpoints use

two signalling channels for call control, and Gatekeeper related functions. The formatting used for these channels shall conform to Recommendation H.225.0.

#### 6.2.11.1 Logical channel numbers

Each logical channel is identified by a Logical Channel Number, in the range 0 to 65 535, which serves only to associate logical channels with the transport connection. Logical channel numbers are selected arbitrarily by the transmitter, except that logical channel 0 shall be permanently assigned to the H.245 Control Channel. The actual Transport Address that the transmitter shall transmit to shall be returned by the receiver in the **openLogicalChannelAck** message.

#### 6.2.11.2 Logical channel bit rate limits

A logical channel's bandwidth shall have an upper limit specified by the minimum of the endpoint's transmit capability (if present) and the receiving endpoint's receive capability. Based on this limit, an endpoint shall open a logical channel at a bit rate at or below this upper limit. A transmitter shall transmit an information stream within the logical channel at any bit rate at or below the open logical channel bit rate. The limit applies to the information streams which are the content of the logical channel(s), not including RTP headers, RTP payload headers and network headers and other overhead.

H.323 endpoints shall obey to the **flowControlCommand** message of H.245, which commands a limit to the bit rate of a logical channel or the aggregate bit rate of all logical channels. H.323 endpoints that want to limit the bit rate of a logical channel, or the aggregate bit rate of all logical channels should send the **flowControlCommand** message to the transmitting endpoint.

When the terminal has no information to send in a given channel, the terminal shall send no information. Fill data shall not be sent on the network in order to maintain a specific data rate.

### 6.3 Gateway characteristics

The Gateway shall provide the appropriate translation between transmission formats (for example H.225.0 to/from H.221) and between communications procedures (for example H.245 to/from H.242). This translation is specified in Recommendation H.246. The Gateway shall also perform call setup and clearing on both the network side and the SCN side. Translation between video, audio, and data formats may also be performed in the Gateway. In general, the purpose of the Gateway (when not operating as an MCU) is to reflect the characteristics of a network endpoint to an SCN endpoint, and the reverse, in a transparent fashion.

An H.323 endpoint may communicate with another H.323 endpoint on the same network directly and without involving a Gateway. The Gateway may be omitted if communications with SCN terminals (terminals not on the network) are not required. It may also be possible for a terminal on one segment of the network to call out through one Gateway and back onto the network through another Gateway in order to bypass a router or a low bandwidth link.

The Gateway has the characteristics of an H.323 Terminal or MCU on the network, and of the SCN terminal or MCU on the SCN. The choice of terminal or MCU is left to the manufacturer. The Gateway provides the necessary conversion between the different terminal types. Note that the Gateway may initially operate as a terminal, but later using H.245 signalling begin to operate as an MCU for the same call that was initially point-to-point. Gatekeepers are aware of which terminals are Gateways since this is indicated when the terminal/Gateway registers with the Gatekeeper.

A Gateway which passes T.120 data between the SCN and the network may contain a T.120 MCS Provider which connects the T.120 MCS Providers on the network to the T.120 MCS Providers on the SCN.

Four examples of an H.323 Gateway are shown in Figure 5. The diagrams show the H.323 terminal or MCU function, the SCN terminal or MCU function, and the conversion function. The H.323 terminal function has the characteristics described in 6.2. The H.323 MCU function has the characteristics described in 6.5. The Gateway appears to the other H.323 terminals on the network as one or more H.323 terminals, or an H.323 MCU. It communicates with the other H.323 terminals using the procedures in this Recommendation.

The SCN terminal or MCU function has the characteristics described in the appropriate Recommendation (H.310, H.320, H.321, H.322, H.324, V.70, GSTN or ISDN speech only terminals). The Gateway appears to the terminals on the SCN as one or more of the same terminal types or MCUs. It communicates to another terminal on the SCN using the procedures described in the appropriate Recommendation for that terminal. SCN signalling procedures are beyond the scope of this Recommendation, including such topics as whether the H.323 Gateway appears as a terminal or a network to the SCN. Note that a Gateway may convert H.323 directly to H.324 or H.310 without going to H.320.

Gateways supporting interworking with speech only terminals on GSTN or ISDN should generate and detect DTMF signals corresponding to H.245 **userInputIndications** for 0-9, \*, and #.

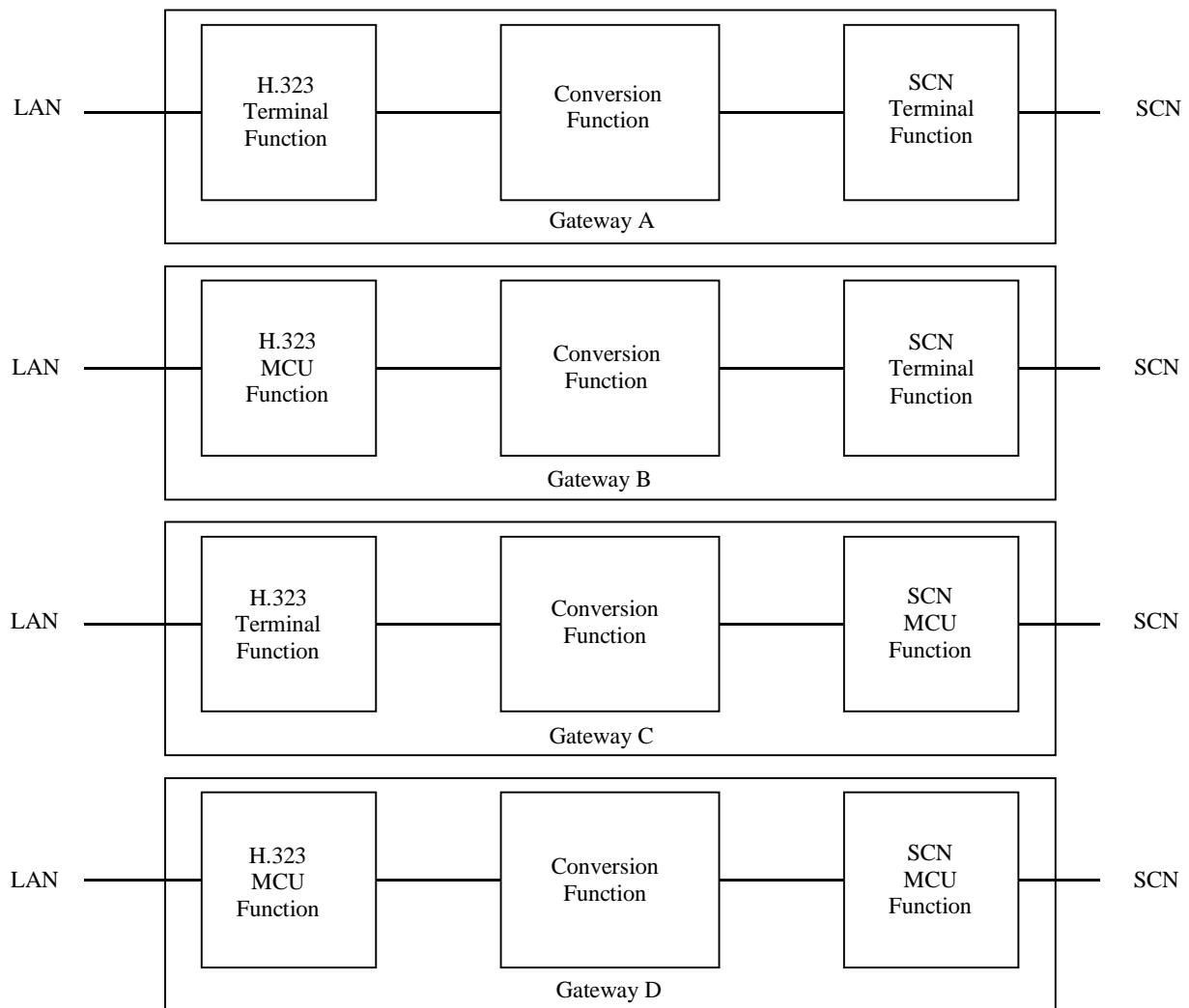
The conversion function provides the necessary conversion of transmission format, control, audio, video, and/or data streams between the different terminal Recommendations. At a minimum, the Gateway shall provide a conversion function for the transmission format, call setup signals and procedures, and communications control signals and procedures. When required, the Gateway shall provide for H.242 to H.245 conversion. The Gateway performs the appropriate conversion between the H.225.0 Call Signalling and the SCN signalling system (Q.931, Q.2931, etc.). The conversion between Q.931 messages on the network and Q.931 messages on the SCN is described in Recommendation H.246.

All call signalling received by the Gateway from an SCN endpoint and not applicable to the Gateway should be passed through to the network endpoint, and vice versa. This signalling includes, but is not limited to, Q.932, Q.950, and H.450-Series messages. This will allow H.323 endpoints to implement the Supplementary Services defined in those Recommendations. The handling of other SCN call signalling systems is for further study.

This Recommendation describes the connection of one H.323 terminal on the network to one external terminal on the SCN through the Gateway. The actual number of H.323 terminals that can communicate through the Gateway is not subject to standardization. Similarly, the number of SCN connections, number of simultaneous independent conferences, audio/video/data conversion functions, and inclusion of multipoint functions is left to the manufacturer. If the Gateway includes an MCU function on the network side, that function shall be an H.323 MCU on the network side. If the Gateway includes an MCU function on the SCN side, it may appear as an H.231/H.243 MCU, or as an MCU for H.310 or H.324 systems (these MCUs are indicated as for further study in the respective Recommendations) on the SCN side.

A Gateway may be connected via the SCN to other Gateways to provide communication between H.323 terminals which are not on the same network.

Equipment which provides transparent interconnection between networks without using H-series protocols (such as routers and remote dial in units) are not Gateways as defined within the scope of this Recommendation.



T1521240-96

**Figure 5/H.323 – H.323 gateway configurations**

#### 6.4 Gatekeeper characteristics

The Gatekeeper, which is optional in an H.323 system, provides call control services to the H.323 endpoints. More than one Gatekeeper may be present and communicate with each other in an unspecified fashion. The Gatekeeper is logically separate from the endpoints; however, its physical implementation may coexist with a terminal, MCU, Gateway, MC, or other non-H.323 network device.

When it is present in a system, the Gatekeeper shall provide the following services:

- **Address Translation** – The Gatekeeper shall perform alias address to Transport Address translation. This should be done using a translation table which is updated using the Registration messages described in clause 7. Other methods of updating the translation table are also allowed.
- **Admissions Control** – The Gatekeeper shall authorize network access using ARQ/ACF/ARJ H.225.0 messages. This may be based on call authorization, bandwidth, or some other criteria which is left to the manufacturer. It may also be a null function which admits all requests.

- Bandwidth Control – The Gatekeeper shall support BRQ/BRJ/BCF messages. This may be based on bandwidth management. It may also be a null function which accepts all requests for bandwidth changes.
- Zone Management – The Gatekeeper shall provide the above functions for terminals, MCUs, and Gateways which have registered with it as described in 7.2.

The Gatekeeper may also perform other optional functions such as:

- Call Control Signalling – The Gatekeeper may choose to complete the call signalling with the endpoints and may process the call signalling itself. Alternatively, the Gatekeeper may direct the endpoints to connect the Call Signalling Channel directly to each other. In this manner, the Gatekeeper can avoid handling the H.225.0 call control signals. The Gatekeeper may have to act as the network as defined in Recommendation Q.931 in order to support supplementary services. This operation is for further study.
- Call Authorization – Through the use of the H.225.0 signalling, the Gatekeeper may reject calls from a terminal due to authorization failure. The reasons for rejection may include, but are not limited to, restricted access to/from particular terminals or Gateways, and restricted access during certain periods of time. The criteria for determining if authorization passes or fails is outside the scope of this Recommendation.
- Bandwidth Management – Control of the number of H.323 terminals permitted simultaneous access to the network. Through the use of the H.225.0 signalling, the Gatekeeper may reject calls from a terminal due to bandwidth limitations. This may occur if the Gatekeeper determines that there is not sufficient bandwidth available on the network to support the call. The criteria for determining if bandwidth is available is outside the scope of this Recommendation. Note that this may be a null function, i.e. all terminals are granted access. This function also operates during an active call when a terminal requests additional bandwidth.
- Call Management – For example, the Gatekeeper may maintain a list of ongoing H.323 calls. This information may be necessary to indicate that a called terminal is busy, and to provide information for the Bandwidth Management function.
- Gatekeeper management information data structure – For further study.
- Bandwidth reservation for terminals not capable of this function – For further study.
- Directory services – For further study.

In order to support ad hoc Multipoint Conferences, the Gatekeeper may choose to receive the H.245 Control Channels from the two terminals in a point-to-point conference. When the conference switches to a multipoint conference, the Gatekeeper can redirect the H.245 Control Channel to an MC. The Gatekeeper need not process the H.245 signalling; it only needs to pass it between the terminals or the terminals and the MC.

Networks which contain Gateways should also contain a Gatekeeper in order to translate incoming E.164 or **partyNumber** addresses into Transport Addresses.

H.323 entities that contain a Gatekeeper shall have a mechanism to disable the internal Gatekeeper so that when there are multiple H.323 entities that contain a Gatekeeper on a network, the H.323 entities can be configured into the same Zone.

## 6.5 Multipoint controller characteristics

The MC provides control functions to support conferences between three or more endpoints in a multipoint conference. The MC carries out the capabilities exchange with each endpoint in a multipoint conference. The MC sends a capability set to the endpoints in the conference indicating the operating modes in which they may transmit. The MC may revise the capability set that it sends to the terminals as a result of terminals joining or leaving the conference, or for other reasons.

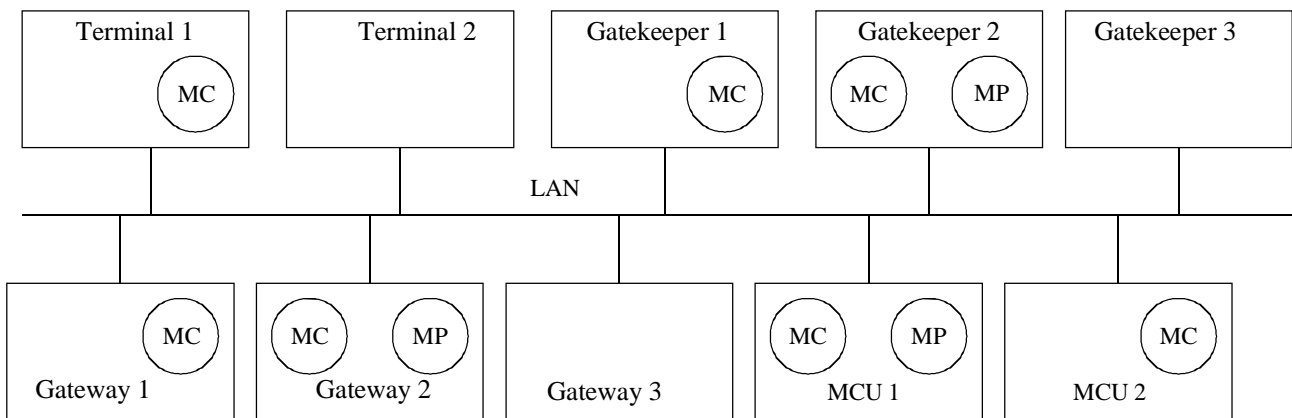
In this manner, the MC determines the Selected Communications Mode (SCM) for the conference. The SCM may be common for all endpoints in the conference. Alternatively, some endpoints may have a different SCM than other endpoints in the conference. The manner in which the MC determines an SCM is not within the scope of this Recommendation.

As part of multipoint conference setup, an endpoint will become connected to an MC on its H.245 Control Channel. This connection may occur:

- via an explicit connection with an MCU;
- via an implicit connection to the MC within a Gatekeeper;
- via an implicit connection to the MC within another terminal or Gateway in the multipoint conference;
- via an implicit connection through a Gatekeeper to an MCU.

The choice of conference mode (e.g. decentralized or centralized) occurs after connection with the MC using H.245 signalling. The choice of conference mode may be limited by the capability of the endpoints or the MC.

The MC may be located within a Gatekeeper, Gateway, terminal, or MCU. See Figure 6.



T1521250-96

NOTE – Gateway, Gatekeeper and MCU can be a single device.

**Figure 6/H.323 – Possible locations of MC and MP in H.323 system**

An MC within a terminal is not callable. It can be included in the call in order to process the H.245 signalling to support ad hoc multipoint conferences. In this case, there may be no distinction between the MC and the H.245 Control Function (see 6.2.8) of the terminal. Communications between them are outside the scope of this Recommendation.

An MC located with the Gatekeeper is not callable; however, an MCU located with a Gatekeeper may be callable. An MCU located with a Gatekeeper may function as an independent MCU. An MC located with a Gatekeeper may be used to support ad hoc multipoint conferences when the Gatekeeper receives the H.245 Control Channels from the endpoints. In this manner, the Gatekeeper can route the H.245 Control Channels to the MC at the start of the call or when the conference switches to multipoint.

The Gateway can function as a terminal or an MCU. When functioning as a terminal, the Gateway may contain an MC. This has the same characteristics as described above for an MC within a terminal.

An MCU always contains an MC. The MCU is callable and the MC processes the H.245 Control Channel from all of the endpoints.

When two or more endpoints are in a conference, the endpoints shall use the master slave resolution procedure of Recommendation H.245 to determine the MC that will control the conference.

After the capability exchange and master/slave determination, the MC may first assign a terminal number to a new endpoint using the **terminalNumberAssign**. The MC shall then notify the other endpoints of the new endpoint in the conference using **terminalJoinedConference**. The new endpoint may request a list of other endpoints in the conference using the **terminalListRequest**.

## 6.6 Multipoint processor characteristics

The MP receives audio, video and/or data streams from the endpoints involved in a centralized or hybrid multipoint conference. The MP processes these media streams and returns them to the endpoints.

Communications between the MC and the MP are not subject to standardization.

The MP may process one or more media stream types. When the MP processes video, it shall process the video algorithms and formats as described in 6.2.4. When the MP processes audio, it shall process the audio algorithms as described in 6.2.5. When the MP processes data, it shall process data streams as described in 6.2.7.

An MP which processes video shall provide either video switching or video mixing. Video switching is the process of selecting the video that the MP outputs to the terminals from one source to another. The criteria used to make the switch may be determined through detection of a change in speaker (sensed by the associated audio level) or through H.245 control. Video mixing is the process of formatting more than one video source into the video stream that the MP outputs to the terminals. An example of video mixing is combining four source pictures into a two-by-two array in the video output picture. The criteria for which sources and how many are mixed is determined by the MC until other controls are defined. The use of the T.120-series Recommendations for these control functions is for further study.

An MP which processes audio shall prepare N-audio outputs from M-audio inputs by switching, mixing, or a combination of these. Audio mixing requires decoding the input audio to linear signals (PCM or analogue), performing a linear combination of the signals and recoding the result to the appropriate audio format. The MP may eliminate or attenuate some of the input signals in order to reduce noise and other unwanted signals. Each audio output may have a different mix of input signals providing for private conversations. The terminals shall assume that their audio is not present in the audio stream returned to them. Terminal removal of its own audio from the MP audio output is for further study.

An MP which processes T.120 data shall be capable of acting as a non-leaf MCS provider and should be capable of acting as the Top MCS Provider. An MP may also process non-standard data, transparent user data and/or other types of data.

The MP may provide algorithm and format conversion, allowing terminals to participate in a conference at different SCMs.

The MP is not callable, the MCU which it is a part of is callable. The MP terminates and sources the media channels.

## 6.7 Multipoint control unit characteristics

The MCU is an endpoint which provides support for multipoint conferences. The MCU shall consist of an MC and zero or more MPs. The MCU uses H.245 messages and procedures to implement features similar to those found in Recommendation H.243.

A typical MCU that supports centralized multipoint conferences consists of an MC and an audio, video and data MP. A typical MCU that supports decentralized multipoint conferences consists of an MC and a data MP supporting Recommendation T.120. It relies on decentralized audio and video processing.

The network side of a Gateway may be an MCU. A Gatekeeper may also include an MCU. In either case, they are independent functions that happen to be co-located.

The MCU shall be callable by other endpoints using the procedures of clause 8.

## **6.8 Multipoint capability**

### **6.8.1 Centralized multipoint capability**

All endpoints shall have centralized multipoint capability. In this mode of operation they communicate with the MC of the MCU in a point-to-point manner on the control channel and with the MP on the audio, video and data channels. In this mode, the MC performs H.245 multipoint control functions, while the MP performs video switching or mixing, audio mixing, and T.120 multipoint data distribution. The MP transmits the resulting video, audio and data streams back to the endpoints. The MP may have the capability to convert between different audio, video and data formats and bit rates, allowing the endpoints to participate in the conference using different communications modes.

The MCU may use multicast to distribute the processed media streams if the endpoints in the conference can receive multicast transmissions. Multicast distribution of data is for further study.

This mode is signalled by the following H.245 capabilities: **centralizedControl**, **centralizedAudio**, **centralizedVideo**, and **centralizedData**. Optionally, **distributedAudio** and **distributedVideo** may be used to indicate multicast distribution of media streams.

### **6.8.2 Decentralized multipoint capability**

If the endpoints have decentralized multipoint capability, they communicate with the MC of an MCU, Gateway, Gatekeeper, or endpoint in a point-to-point mode on the H.245 Control Channel and optionally with an MP on data channels. The endpoints shall have the capability to multicast their audio and video channels to all other endpoints in the conference. The MC may control which endpoint or endpoints are actively multicasting audio and/or video (for example by using the **flowControlCommand** on either channel).

The endpoints receive multicast video channels and select one or more of the available channels for display to the user. The endpoints receive the multicast audio channels and perform an audio mixing function in order to present a composite audio signal to the user.

The MC may provide conference control functions such as chair control, video broadcast and video selection. This shall be done by receiving H.245 from an endpoint and then sending the appropriate control to other endpoints to enable or disable their video multicast. T.120 commands may optionally provide the same functions.

This mode is signalled by the following H.245 capabilities: **centralizedControl**, **distributedAudio**, **distributedVideo**, and **centralizedData**.

### **6.8.3 Hybrid multipoint-centralized audio**

If the endpoints and MCU have hybrid multipoint-centralized audio capability, they may use distributed multipoint for video and centralized multipoint for audio. In this mode, the endpoints communicate with the MC in a point-to-point mode on the H.245 Control Channel and optionally with an MP on data channels.



The endpoints shall have the capability to multicast their video channels to all other endpoints in the conference. The MC may control which endpoint or endpoints are actively multicasting video. The endpoints receive multicast video channels and select one or more of the available channels for display to the user.

All of the endpoints in the conference transmit their audio channels to the MP. The MP performs the audio mixing function and outputs the resulting audio streams to the endpoints. The MP may produce an exclusive audio sum for each endpoint in the conference. Multicast distribution of processed audio is for further study.

This mode is signalled by the following H.245 capabilities: **centralizedControl**, **centralizedAudio**, **distributedVideo**, and **centralizedData**.

#### **6.8.4 Hybrid multipoint-centralized video**

If the endpoints and MCU have hybrid multipoint-centralized video capability, they may use distributed multipoint for audio and centralized multipoint for video. In this mode, the endpoints communicate with the MC in a point-to-point mode on the H.245 Control Channel and optionally with an MP on data channels.

The endpoints shall have the capability to multicast their audio channels to all other endpoints in the conference. The MC may control which endpoint or endpoints are actively multicasting audio. The endpoints receive multicast audio channels and perform a mixing function in order to present a composite audio signal to the user.

All of the endpoints in the conference transmit their video channels to the MP. The MP performs the video switching, mixing, or format conversion functions and outputs the resulting video streams to the endpoints. The MP may produce an exclusive video stream for each endpoint in the conference, or it may multicast a video stream to all participating endpoints, in order to minimize the bandwidth used on the network.

This mode is signalled by the following H.245 capabilities: **centralizedControl**, **distributedAudio**, **centralizedVideo**, and **centralizedData**.

#### **6.8.5 Establishment of common mode**

The MC shall coordinate a common communications mode between the endpoints in the multipoint conference. The MC may force endpoints into a particular common mode of transmission (as allowed by their capability sets) by sending to the endpoint a receive capability set listing only the desired mode of transmission, or the MC may rely on **multipointModeCommand** and mode preference commands to enforce mode symmetry. The latter approach should be used since it allows the endpoints to know the full range of conference capabilities available that can be requested.

If the MCU has the capability to convert audio and/or video formats, it may not be necessary to force all endpoints into the same communications mode.

#### **6.8.6 Multipoint rate matching**

Since the endpoints on each link in a multipoint configuration may attempt to operate at different bit rates, the MC shall send H.245 **flowControlCommand** messages to limit the transmitted bit rates to those which can be sent to receivers.

#### **6.8.7 Multipoint lip synchronization**

An MP which is providing audio mixing in either the Centralized or Hybrid multipoint conferences shall modify the time tags of the audio and video streams, taking into account its own time base, in order to maintain audio and video synchronization. Further, when the MP processes the audio and/or video to generate a new stream sourced from the MP, the MP shall generate its own sequence numbers in the audio and video packets.

When mixing audio, the MP should synchronize each of the incoming audio streams to its own timing, mix the audio streams, and then shall generate a new audio stream based on its own timing with its own sequence numbers. If the MP is also switching video, the switched stream shall have its original time-stamp replaced with the MP time base to synchronize it with the mixed audio stream and shall have a new sequence number representing the stream from the MP.

In the case of distributed multipoint conferences, the receiving endpoint may be able to maintain lip synchronization by aligning the selected video stream and its associated audio by using the RTP time tags. Alignment of the other audio streams may not be necessary. If multiple video streams are displayed, the associated audio streams should be aligned.

It may not be possible to guarantee lip synchronization in hybrid multipoint conferences.

### **6.8.8 Multipoint encryption**

In a centralized multipoint configuration, the MP is considered to be a trusted entity. Each port of the MP decrypts the information streams from each of the H.323 endpoints and encrypts the information streams to each endpoint in accordance with 10.1. Operation of an untrusted MCU is for further study.

### **6.8.9 Cascading multipoint control units**

The multipoint control function may be distributed between several MCs. This is called cascading. Cascading allows two or more MCs to communicate with each other in order to control a multipoint conference. Cascading MCs consists of establishing an H.245 Control Channel between the MCs. One MC is defined as the Master MC while the other MCs are defined as Slave MCs.

The procedures for cascading MCs are defined in 8.4.5.

## **7 Call signalling**

Call signalling is the messages and procedures used to establish a call, request changes in bandwidth of the call, get status of the endpoints in the call, and disconnect the call. Call signalling uses messages defined in Recommendation H.225.0 and the procedures described in clause 8. This clause describes some call signalling concepts.

### **7.1 Addresses**

#### **7.1.1 Network address**

Each H.323 entity shall have at least one Network Address. This address uniquely identifies the H.323 entity on the network. Some entities may share a Network Address (i.e. a terminal and a co-located MC). This address is specific to the network environment in which the endpoint is located. Different network environments may have different Network Address formats.

An endpoint may use different Network addresses for different channels within the same call.

#### **7.1.2 TSAP identifier**

For each Network Address, each H.323 entity may have several TSAP Identifiers. These TSAP Identifiers allow multiplexing of several channels sharing the same Network Address.

Endpoints have one well-known TSAP Identifier defined: the Call Signalling Channel TSAP Identifier. Gatekeepers have one well-known TSAP Identifier defined: the RAS Channel TSAP Identifier and one well-known multicast address defined: Discovery Multicast Address. These are defined in Appendix IV/H.225.0.

Endpoints and H.323 entities should use dynamic TSAP Identifiers for the H.245 Control Channel, Audio Channels, Video Channels, and Data Channels. The Gatekeeper should use a dynamic TSAP

Identifier for Call Signalling Channels. The RAS Channels and Signalling Channels may be redirected to dynamic TSAP Identifiers during the registration procedure.

### **7.1.3 Alias address**

An endpoint may also have one or more alias addresses associated with it. An alias address may represent the endpoint or it may represent conferences that the endpoint is hosting. The alias addresses provide an alternate method of addressing the endpoint. These address include E.164 or **partyNumber** addresses (network access number, telephone number, etc.), H.323 IDs (alphanumeric strings representing names, e-mail like addresses, etc.), and any others defined in Recommendation H.225.0. Alias addresses shall be unique within a Zone. Gatekeepers, MCs, and MPs shall not have alias addresses.

When there is no Gatekeeper in the system, the calling endpoint shall address the called endpoint directly using the Call Signalling Channel Transport Address of the called endpoint. When there is a Gatekeeper in the system, the calling endpoint may address the called endpoint by its Call Signalling Channel Transport Address, or alias address. The latter shall be translated into a Call Signalling Channel Transport Address by the Gatekeeper.

The called endpoint's E.164 address may consist of an optional access code followed by the E.164 address. The access code consists of n digits from the set of 0 to 9, \*, and #. The number of digits and their meaning is left to the discretion of the manufacturer. One purpose of such an access code might be to request access to a Gateway. The Gatekeeper may alter this address prior to sending it to the destination.

The H.323 ID consists of a string of ISO/IEC 10646-1 characters as defined in Recommendation H.225.0. It may be a user name, conference name, email name, or other identifier.

An endpoint may have more than one alias address (including more than one of the same type) which is translated to the same Transport Address.

## **7.2 Registration, Admission and Status (RAS) channel**

The RAS Channel shall be used to carry messages used in the Gatekeeper discovery and endpoint registration processes which associate an endpoint's alias address with its Call Signalling Channel Transport Address. The RAS Channel shall be an unreliable channel.

Since the RAS messages are transmitted on an unreliable channel, H.225.0 recommends timeouts and retry counts for various messages. An endpoint or Gatekeeper which cannot respond to a request within the specified timeout may use the Request in Progress (RIP) message to indicate that it is still processing the request. An endpoint or Gatekeeper receiving the RIP shall reset its timeout timer and retry counter.

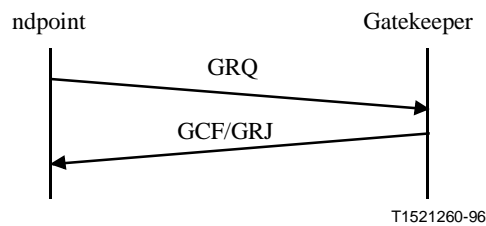
### **7.2.1 Gatekeeper discovery**

Gatekeeper discovery is the process an endpoint uses to determine which Gatekeeper to register with. This may be done manually or automatically. Manual discovery relies on methods outside the scope of this Recommendation to determine which Gatekeeper an endpoint is associated with. The endpoint is configured with the Transport Address of the associated Gatekeeper. For example, it may be entered at endpoint configuration, or it may be entered into an initialization file. In this way, the endpoint knows *a priori* which Gatekeeper it is associated with. The endpoint can now register with that Gatekeeper.

The Automatic method allows the endpoint-Gatekeeper association to change over time. The endpoint may not know who its Gatekeeper is, or may need to identify another Gatekeeper due to a failure. This may be done through auto discovery. Auto discovery allows for lower administrative overhead in configuring individual endpoints and additionally allows replacement of an existing Gatekeeper without manually reconfiguring all of the affected endpoints.

The endpoint may multicast (or use other methods as described in Appendix IV/H.225.0) a Gatekeeper Request (GRQ) message, asking "Who is my Gatekeeper?". This is sent to the Gatekeeper's well-known Discovery Multicast Address. One or more Gatekeepers may respond with the Gatekeeper Confirmation (GCF) message indicating "I can be your Gatekeeper.", and returns the Transport Address of the Gatekeeper's RAS Channel. If a Gatekeeper does not want the endpoint to register to it, it shall return Gatekeeper Reject (GRJ). See Figure 7. If more than one Gatekeeper responds, the endpoint may choose the Gatekeeper it wants to use. At this point, the endpoint knows which Gatekeeper to register with. The endpoint can now register with that Gatekeeper.

In the event that the endpoint knows the location of the Gatekeeper by some *a priori* means, the endpoint may still choose to unicast the GRQ to the Gatekeeper for the purpose of H.225.0 cryptological exchange.



**Figure 7/H.323 – Auto discovery**

In order to provide redundancy in systems which use a Gatekeeper, the Gatekeeper may indicate alternate Gatekeepers that may be used in the event of a primary Gatekeeper failure. This list of alternate Gatekeepers is provided in the **alternateGatekeeper** field of the GCF and RCF messages.

If no Gatekeeper responds within a timeout, the endpoint may retry the GRQ. An endpoint shall not send a GRQ within 5 s after sending a previous one. If no response is received, the endpoint may use the manual discovery method.

If at any time an endpoint determines it has an invalid registration with its Gatekeeper, it must rediscover its Gatekeeper. The invalid registration may be detected by either receiving an RRJ message from a Gatekeeper in response to an RRQ from the endpoint, or not receiving any response to an RRQ from the endpoint within a timeout.

The GRQ may be repeated periodically (i.e. at endpoint power-up), so the Gatekeeper shall be able to handle multiple requests from the same endpoint.

### 7.2.2 Endpoint registration

Registration is the process by which an endpoint joins a Zone, and informs the Gatekeeper of its Transport Addresses and alias addresses. As part of their configuration process, all endpoints shall register with the Gatekeeper identified through the discovery process. Registration shall occur before any calls are attempted and may occur periodically as necessary (for example, at endpoint power-up).

A Gateway or MCU may register a single Transport Address or multiple Transport Addresses as its call signalling address, and may register a single Transport Address or multiple Transport Addresses as its RAS address. The use of multiple Transport Addresses shall indicate a prioritized list of addresses to try when communicating with a given endpoint through either its RAS or Call Signalling Channel.

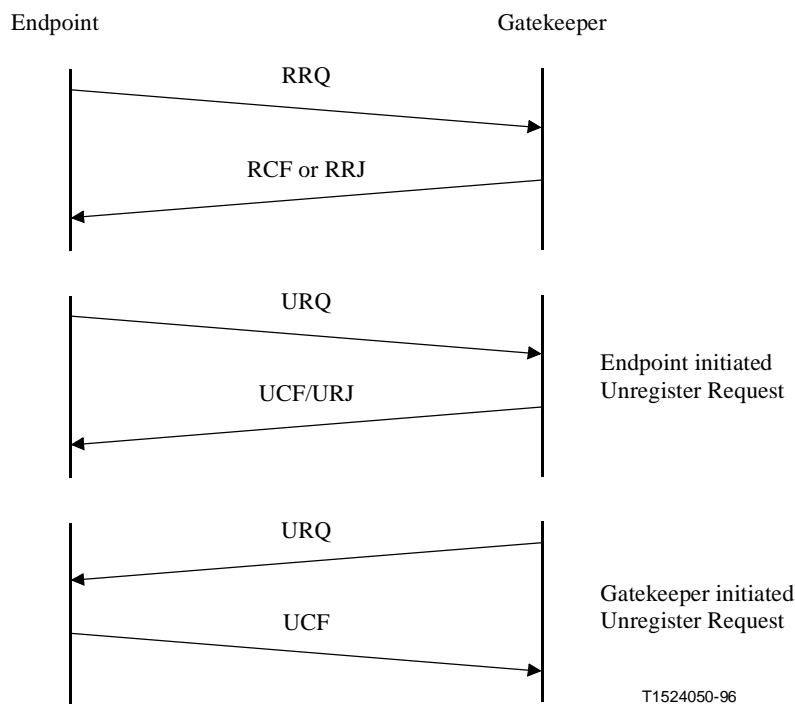
An endpoint shall send a Registration Request (RRQ) to a Gatekeeper. This is sent to the Gatekeeper's RAS Channel Transport Address. The endpoint has the Network Address of the Gatekeeper from the Gatekeeper discovery process and uses the well-known RAS Channel TSAP

Identifier. The Gatekeeper shall respond with either a Registration Confirmation (RCF) or a Registration Reject (RRJ). See Figure 8. An endpoint shall only register with a single Gatekeeper.

The RRQ may be repeated periodically (i.e. at terminal power-up), so the Gatekeeper shall be able to handle multiple requests from the same endpoint. If a Gatekeeper receives an RRQ having the same alias address (or list of alias addresses) and the same Transport Addresses as an active registration, it shall respond with RCF. If a Gatekeeper receives an RRQ having the same alias address (or list of alias addresses) as an active registration and different Transport Addresses, it may confirm the request, if it complies with the Gatekeeper's registration policy. If the request does not comply with the Gatekeeper's registration policy, the Gatekeeper should reject the registration indicating a duplicate or invalid registration. If the Gatekeeper receives an RRQ having the same Transport Addresses as an active registration and a different alias address (or list of alias addresses), it should replace the translation table entries. The Gatekeeper may have a method to authenticate these changes.

An endpoint may indicate a backup, redundant, or alternate Transport Addresses using the **alternateEndpoint** structure within the RAS messages. This allows an endpoint to have a secondary network interface or a secondary H.323 endpoint as a backup. The Gatekeeper shall reject ambiguous registrations. The Gatekeeper may reject the registration for other reasons, such as changes in discovery or security issues.

If the endpoint does not include an alias address in the RRQ message, the Gatekeeper may assign one. The Gatekeeper shall return the assigned alias address to the terminal in the RCF message.



**Figure 8/H.323 – Registration**

An endpoint may cancel its registration by sending an Unregister Request (URQ) message to the Gatekeeper. The Gatekeeper shall respond with an Unregister Confirmation (UCF) message. This allows endpoints to change the alias address associated with its Transport Address, or vice versa. If the endpoint was not registered with the Gatekeeper, it shall return an Unregister Reject (URJ) message to the endpoint.

A Gatekeeper may cancel the registration of an endpoint by sending an Unregister Request (URQ) message to the endpoint. The endpoint shall respond with an Unregister Confirmation (UCF) message. The endpoint shall re-register with a Gatekeeper prior to initiating any calls. This may require the endpoint to register with a new Gatekeeper.

An endpoint which is not registered with a Gatekeeper is called an unregistered endpoint. This type of endpoint does not request admission permission from a Gatekeeper and so cannot participate in admissions control, bandwidth control, address translation and other functions performed by the Gatekeeper.

### 7.2.2.1 Use of Lightweight RRQ

An endpoint's registration with a Gatekeeper may have a finite life. An endpoint may request a **timeToLive** in the RRQ message to the Gatekeeper. The Gatekeeper may respond with an RCF containing the same **timeToLive** or a shorter **timeToLive**. After this time, the registration shall be expired. The **timeToLive** is expressed in seconds. Prior to the expiration time, the endpoint may send an RRQ message having the **keepAlive** bit set. The keep-alive RRQ may include a minimum amount of information as described in H.225.0. The keep-alive RRQ shall reset the time to live timer in the Gatekeeper, allowing the registration to be extended. After the expiration time, the endpoint must re-register with a Gatekeeper using a full RRQ message.

If the Gatekeeper does not include a **timeToLive** value in the RCF, the registered endpoint shall consider that the Gatekeeper is not supporting the keep-alive mechanism. Endpoints shall not send RRQs with the **keepAlive** field set to Gatekeepers which have indicated that they are not supporting the keep-alive mechanism.

Gatekeepers should not treat an RRQ with the **keepAlive** field set as a full registration (i.e. for updating or initializing its translation tables).

Endpoints should consider messaging and processing delays when determining when their registration will expire (i.e. the duration of their own time-to-live timer) at the Gatekeeper.

Expiration of the time-to-live timer in the Gatekeeper results in the expiration of the registration of the endpoint. A Gatekeeper may send a URQ to the endpoint as a notification of such expiration. This allows for loss of synchronization between the time-to-live timers of the Gatekeeper and the endpoint. It also indicates a need for re-registration to endpoints which do not support the keep-alive mechanism.

An endpoint which sends a lightweight RRQ to its Gatekeeper after the time-to-live timer has expired in the Gatekeeper will receive an RRJ response with **rejectReason** of either **fullRegistrationRequired** or **discoveryRequired**, depending on Gatekeeper requirements.

An endpoint which sends an ARQ to its Gatekeeper after the time-to-live timer has expired in the Gatekeeper will receive an ARJ with **rejectReason** of either **callerNotRegistered** or **calledPartyNotRegistered**. An endpoint which initiates a new call through its Gatekeeper after expiration of the Gatekeeper's time-to-live timer will receive a Release Complete message with a **reason** of **callerNotRegistered** or **calledPartyNotRegistered**.

Disposition of existing calls upon expiration of the time-to-live timer is implementation dependent.

### 7.2.3 Endpoint location

An endpoint or Gatekeeper which has an alias address for an endpoint and would like to determine its contact information may issue a Location Request (LRQ) message. This message may be sent to a specific Gatekeeper's RAS Channel TSAP Identifier, or may be multicast like the GRQ message to the Gatekeeper's well-known Discovery Multicast Address. The Gatekeeper with which the requested endpoint is registered shall respond with the Location Confirmation (LCF) message containing the contact information of the endpoint or the endpoint's Gatekeeper. Contact information shall include the Call Signalling Channel and RAS Channel addresses to be used to reach the

endpoint and optionally additional destination information which can provide dialing information and extension information concerning the requested endpoint.

All Gatekeepers with which the requested endpoint is not registered shall return Location Reject (LRJ) if they received the LRQ on the RAS Channel. Any Gatekeeper with which the requested endpoint is not registered shall not respond to the LRQ, if it received the LRQ on the Discovery Multicast address.

An endpoint or Gatekeeper may include one or more E.164 or **partyNumber** extensions which it wishes to dial in the **destinationInfo** field of the LRQ to attempt to locate an available Gateway outside of its zone. A Gatekeeper which receives an LRQ requesting an available Gateway is not obligated to make its Gateways available to such a request.

A Gatekeeper may be aware of the alias address and connection information of endpoints on the SCN. This Gatekeeper could respond to an LRQ requesting information on the SCN endpoint with the connection information necessary to reach that endpoint. This would include the information necessary to address the Gateway as well as the SCN endpoint. Note that the SCN endpoint is not registered with the Gatekeeper in the sense that it exchanges RRQ/RCF messages with the Gatekeeper. The method by which a Gatekeeper becomes aware of the SCN endpoint information is outside the scope of this Recommendation.

#### **7.2.4 Admissions, bandwidth change, status and disengage**

The RAS Channel is also used for the transmission of Admissions, Bandwidth Change, Status and Disengage messages. These messages take place between an endpoint and a Gatekeeper and are used to provide admissions control and bandwidth management functions. The detailed use of these messages is described in clause 8.

The Admissions Request (ARQ) message specifies the requested Call Bandwidth. This is an upper limit on the aggregate bit rate for all transmitted and received, audio and video channels excluding any RTP headers, RTP payload headers, network headers, and other overhead. Data and control channels are not included in this limit. The Gatekeeper may reduce the requested Call Bandwidth in the Admissions Confirm (ACF) message. An endpoint shall assure that the aggregate bit rate, averaged over one second, for all transmitted and received, audio and video channels is at or below the Call Bandwidth. An endpoint or the Gatekeeper may attempt to modify the Call Bandwidth during a call using the Bandwidth Change Request (BRQ) message.

#### **7.2.5 Access tokens**

An Access Token is a string passed in some RAS messages and the Setup message. The Access Tokens have two uses. First, they can provide privacy by shielding an endpoint's Transport Address and Alias Address information from a calling party. A user may give out only the Access Token for a calling party to use in reaching the endpoint. The Gatekeeper will know the endpoint related to the Access Token from the registration process, so that calls using the Access Token can be routed through the Gatekeeper to the called endpoint. The use of the access token only applies to the Gatekeeper routed call model when attempting to hide the Transport Address from the endpoint.

The second use of the Access Token is in ensuring that calls are routed properly through H.323 entities. An Access Token returned by a Gatekeeper shall be used in any subsequent setup messages sent by the endpoint. This Access Token may be used by a Gateway to assure that the endpoint has permission to use the Gateway resources, or it may be used by a called endpoint to assure that the calling endpoint can signal it directly.

The Access Token may also be distributed by out-of-band methods to assure proper access to Gateways and endpoints in systems which do not have Gatekeepers.

### 7.3 Call signalling channel

The Call Signalling Channel shall be used to carry H.225.0 call control messages. The Call Signalling channel shall be a reliable channel.

In networks that do not contain a Gatekeeper, call signalling messages are passed directly between the calling and called endpoints using the Call Signalling Transport Addresses. In these networks, it is assumed that the calling endpoint knows the Call Signalling Transport Address of the called endpoint and thus can communicate directly.

In networks that do contain a Gatekeeper, the initial admission message exchange takes place between the calling endpoint and the Gatekeeper using the Gatekeeper's RAS Channel Transport Address. Within the initial admissions message exchange, the Gatekeeper indicates in the ACF message whether to send the call signalling directly to the other endpoint or to route it through the Gatekeeper. The call signalling messages are sent to either the endpoint's Call Signalling Transport Address or the Gatekeeper's Call Signalling Transport Address.

The Call Signalling Channel may carry signalling for many concurrent calls, using the call reference value to associate the message with the call. An entity indicates its ability to handle multiple concurrent calls on the same call signalling connection by setting the **multipleCalls** flag to TRUE in messages that it sends on the Call Signalling Channel. An entity that is capable of processing multiple concurrent calls on the Call Signalling Channel may indicate that it will support no additional calls on the signalling channel by sending Release Complete with **newConnectionNeeded** as the **reason**. An entity that receives Release Complete with **newConnectionNeeded** can attempt to connect a new Call Signalling Channel.

The Call Signalling Channel may be established prior to the actual need to signal a call, and the channel may remain connected between calls. An entity may indicate this capability by setting the **maintainConnection** flag to TRUE in messages that it sends on the Call Signalling Channel. In addition, an endpoint which has this capability should indicate this when it registers with a gatekeeper. This will allow a gatekeeper that utilizes gatekeeper routing to connect to the endpoint at any point after registration. If the connection drops while no call or signalling is active, neither end shall attempt to open the connection until signalling is needed.

Recommendation H.225.0 specifies the mandatory Q.931 messages that are used for call signalling in this Recommendation. Clause 8 specifies the procedures for using them.

#### 7.3.1 Call signalling channel routing

Call signalling messages may be passed in two ways. The first method is Gatekeeper Routed Call Signalling (see Figure 9). In this method, call signalling messages are routed through the Gatekeeper between the endpoints. The second method is Direct Endpoint Call Signalling (see Figure 10). In this method, the call signalling messages are passed directly between the endpoints. The choice of which methods is used is made by the Gatekeeper.

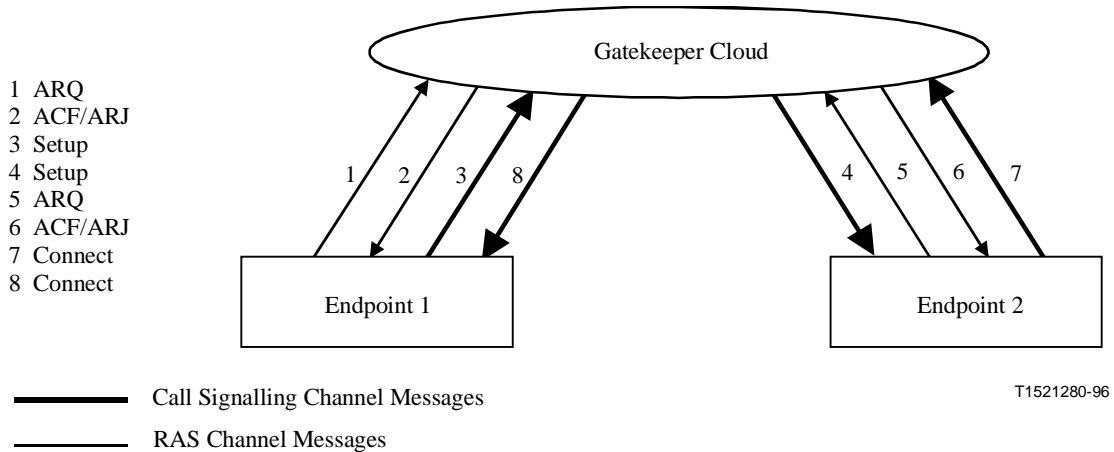
Both methods use the same kinds of connections for the same purposes, and the same messages. Admissions messages are exchanged on RAS channels with the Gatekeeper, followed by an exchange of call signalling messages on a Call Signalling channel. This is then followed by the establishment of the H.245 Control Channel. The actions of the Gatekeeper in response to the admission messages determine which call model is used; this is not under the control of the endpoint, although the endpoint can specify a preference.

For the Gatekeeper Routed method, the Gatekeeper may choose to close the Call Signalling Channel after the call setup is completed, or it may choose to keep it open for the duration of the call to support supplementary services. Only the Gatekeeper shall close the Call Signalling Channel and it should not be closed when a Gateway is involved in the call. If the Gatekeeper closes the Call Signalling Channel then the present state of the call shall be retained by the entities involved. The Gatekeeper may re-open the Call Signalling Channel at any time during the call.

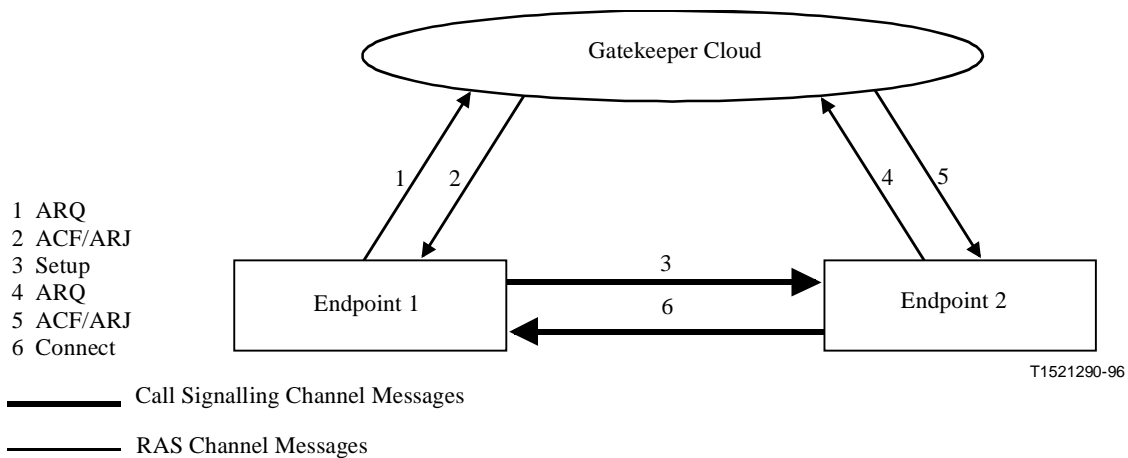


The symmetrical signalling method of Annex D/Q.931 shall be used for all mandatory call signalling procedures. This does not address the role that a Gateway might play on the SCN side using Q.931 or other call signalling protocols.

The Gatekeeper Clouds in Figures 9 through 12 contain one or more Gatekeepers which may or may not communicate with each other. The endpoints may be connected to the same Gatekeeper or to different Gatekeepers.



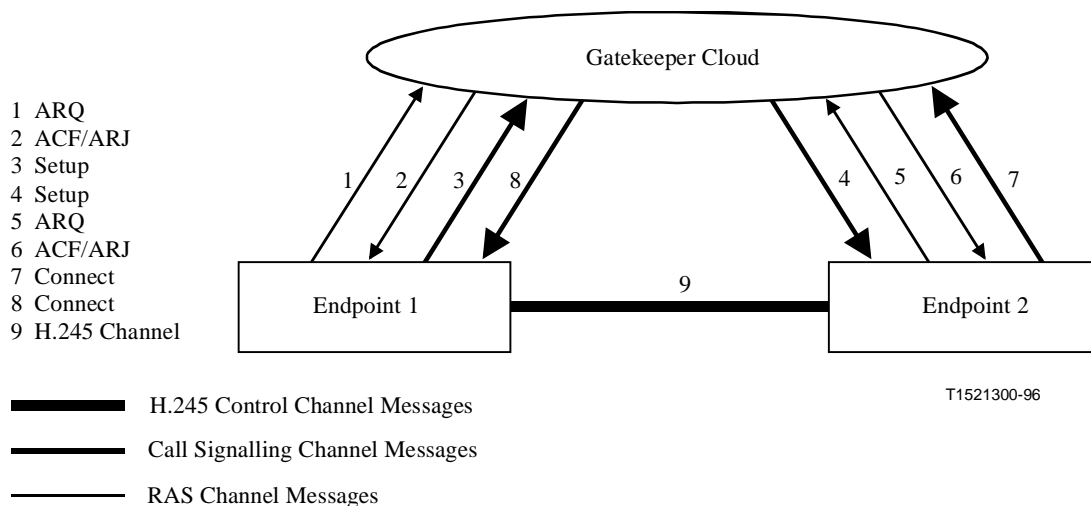
**Figure 9/H.323 – Gatekeeper routed call signalling**



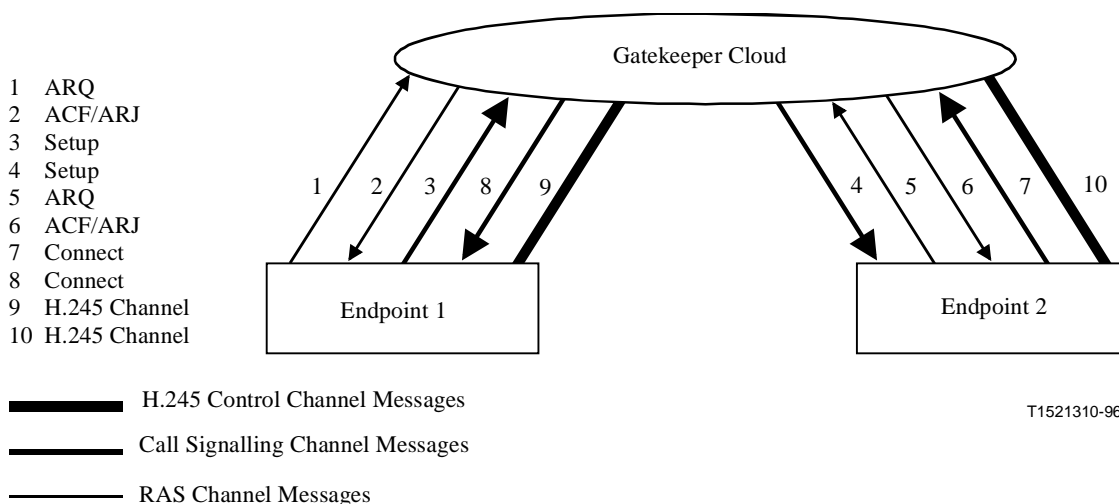
**Figure 10/H.323 – Direct endpoint call signalling**

### 7.3.2 Control channel routing

When Gatekeeper Routed call signalling is used, there are two methods to route the H.245 Control Channel. In the first method, the H.245 Control Channel is established directly between the endpoints. See Figure 11. This method is for further study. In the second method, the H.245 Control Channel is routed between the endpoints through the Gatekeeper. See Figure 12. This method allows the Gatekeeper to redirect the H.245 Control Channel to an MC when an ad hoc multipoint conference switches from a point-to-point conference to a multipoint conference. This choice is made by the Gatekeeper. When Direct Endpoint call signalling is used, the H.245 Control Channel can only be connected directly between the endpoints.



**Figure 11/H.323 – Direct H.245 control channel connection between endpoints**



**Figure 12/H.323 – Gatekeeper routed H.245 control**

### 7.3.3 Call Signalling and control protocol revisions

When a call is routed through a gatekeeper, gatekeepers shall use the following rules to determine the H.225.0 or H.245 version number to be indicated in messages originated by an endpoint and routed or forwarded by the gatekeeper:

- a) If the originating endpoint's H.225.0 or H.245 version number is less than or equal to the gatekeeper's version number, and the gatekeeper chooses to proxy the functions of an equal or later version number on behalf of the originating endpoint; the routed messages shall reflect the version number of the gatekeeper. Otherwise they shall reflect the version number of the originating endpoint.
- b) If the originating endpoint's version number is greater than gatekeeper's, the routed messages shall reflect the version number of the gatekeeper.

In all cases, messages sent by the gatekeeper shall use the ASN.1 encoding specified by the H.225.0 or H.245 version to be used by the gatekeeper according to these rules.

## 7.4 Call reference value

All call signalling and RAS messages contain a Call Reference Value (CRV). Refer to Recommendation H.225.0. There is one CRV for the Call Signalling Channel and an independent CRV for the RAS channel. One CRV is used to associate the call signalling messages. This CRV shall be used in all call signalling messages between two entities (endpoint to Gatekeeper, endpoint-to-endpoint, etc.) related to the same call. A second CRV is used to associate the RAS messages. This CRV shall be used in all RAS messages between two entities related to the same call. New CRVs shall be used for new calls. A second call from an endpoint to invite another endpoint into the same conference shall use new CRVs. The CRV is not the same as the Call ID or the Conference ID (CID). The CRV associates call signalling or RAS messages between two entities within the same call, the Call ID associates all messages between all entities within the same call, and the CID associates all messages between all entities within all calls in the same conference.

## 7.5 Call ID

The Call ID is a globally unique non-zero value created by the calling endpoint and passed in various H.225.0 messages. The Call ID identifies the call with which the message is associated. It is used to associate all RAS and Call Signalling messages related to the same call. Unlike CRV, the Call ID does not change within a call. All messages from the calling endpoint to its Gatekeeper, the calling endpoint to the called endpoint, and the called endpoint to its Gatekeeper related to the same call shall contain the same Call ID. The Call ID is encoded as described in Recommendation H.225.0. In reference to Figures 13 through 23 in clause 8, all messages within a figure shall have the same Call ID.

When a version 1 endpoint calls a version 2 endpoint, it is the responsibility of the version 2 endpoint to generate a Call ID prior to sending ARQ to its Gatekeeper.

## 7.6 Conference ID and Conference Goal

The Conference ID (CID) is a unique non-zero value created by the calling endpoint and passed in various H.225.0 messages. The CID identifies the conference with which the message is associated. Therefore, messages from all endpoints within the same conference will have the same CID. The CID is encoded as specified in Recommendation H.225.0.

The **conferenceGoal** indicates the intention of the call. Choices are: **create** – to create a new conference, **join** – to join an existing conference, **invite** – to invite a new endpoint into an existing conference, **capability-negotiation** – negotiate capabilities for a later H.332 conference, and **callIndependentSupplementaryService** – transport of supplementary services APDUs.

## 8 Call signalling procedures

The provision of the communication is made in the following steps:

- Phase A: Call setup (see 8.1).
- Phase B: Initial communication and capability exchange (see 8.2).
- Phase C: Establishment of audiovisual communication (see 8.3).
- Phase D: Call services (see 8.4).
- Phase E: Call termination (see 8.5).

### 8.1 Phase A – Call setup

Call setup takes place using the call control messages defined in Recommendation H.225.0 according to the call control procedures defined below. Requests for bandwidth reservation should take place at the earliest possible phase.

If both the alias address and the Transport Address are specified, preference shall be given to the alias address.

There is no explicit synchronization or locking between two endpoints during the call setup procedure. This implies that endpoint A can send a Setup message to endpoint B at exactly the same time that endpoint B sends a Setup message to endpoint A. It is up to the application to determine if only one call is desired and to take the appropriate action. This action may be for an endpoint to indicate that it is busy whenever it has an outstanding Setup message. If an endpoint can support more than one simultaneous call, it should indicate that it is busy whenever it receives a Setup message from the same endpoint to which it has an outstanding Setup message.

An endpoint shall be capable of sending the Alerting message. Alerting has the meaning that the called party (user) has been alerted of an incoming call. Alerting shall only be originated by the ultimate called endpoint and then only when it has alerted the user. In the case of interworking through a Gateway, the Gateway shall send Alerting when it receives a ring indication from the SCN. If an endpoint can respond to a Setup message with a Connect, Call Proceeding, or Release Complete within 4 seconds, it is not required to send the Alerting message. An endpoint sending the Setup message can expect to receive either an Alerting, Connect, Call Proceeding, or Release Complete message within 4 seconds after successful transmission.

The Connect message should be sent only if it is certain that the H.245 capability exchange will conclude successfully and a minimum level of communications can take place. This is to maintain the consistency of the meaning of the Connect message between packet based networks and circuit switched networks.

### 8.1.1 Basic call setup – Neither endpoint registered

In the scenario shown in Figure 13 neither endpoint is registered to a Gatekeeper. The two endpoints communicate directly. Endpoint 1 (calling endpoint) sends the Setup (1) message to the well-known Call Signalling Channel TSAP Identifier of Endpoint 2. Endpoint 2 responds with the Connect (4) message which contains an H.245 Control Channel Transport Address for use in H.245 signalling.

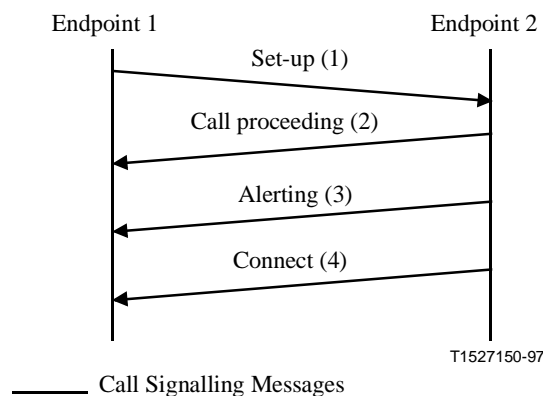
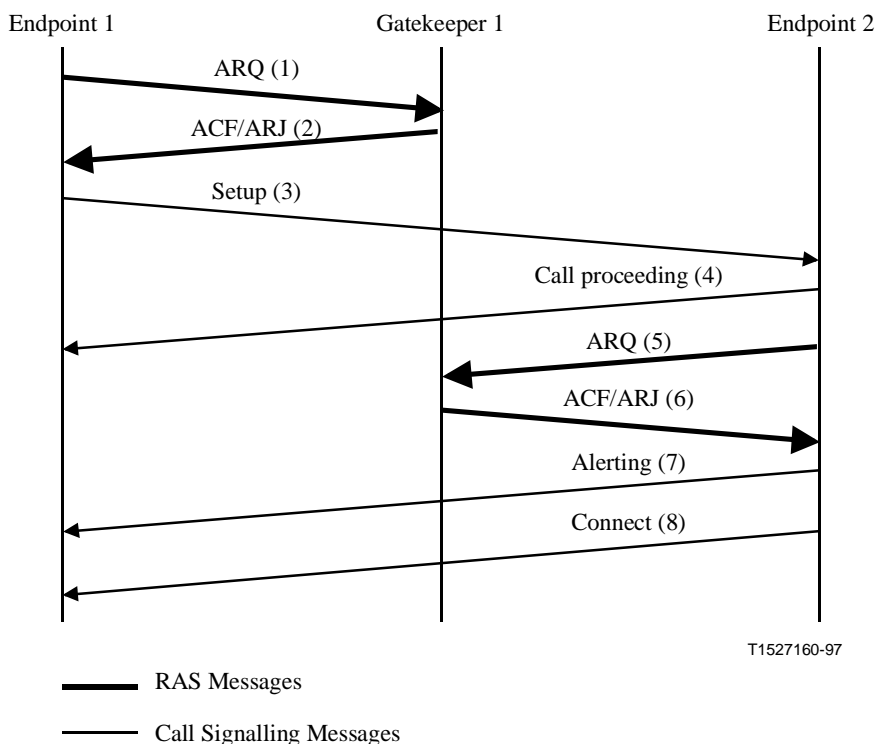


Figure 13/H.323 – Basic call setup, no Gatekeepers

### 8.1.2 Both endpoints registered to the same Gatekeeper

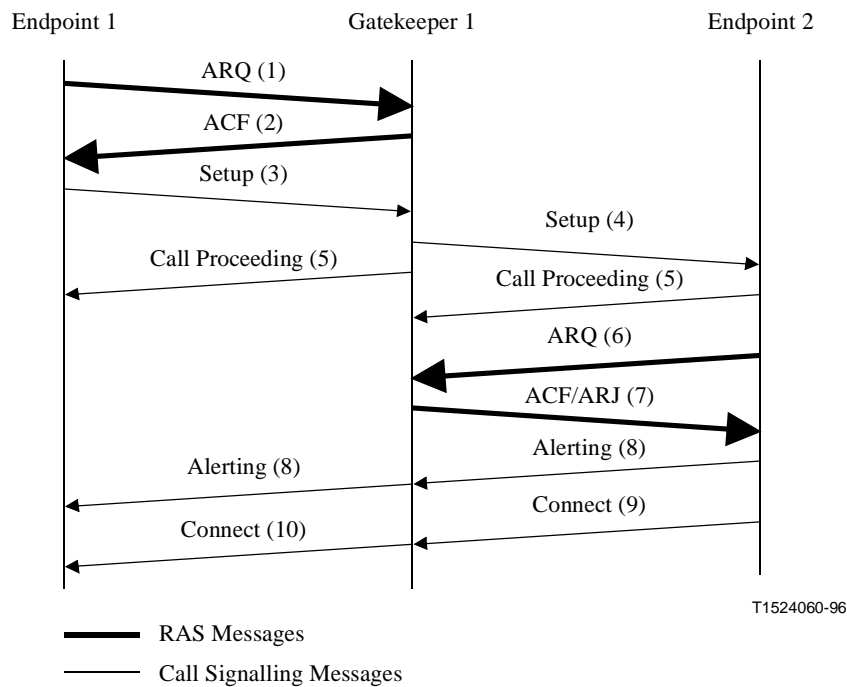
In the scenario shown in Figure 14, both endpoints are registered to the same Gatekeeper, and the Gatekeeper has chosen Direct Call Signalling. Endpoint 1 (calling endpoint) initiates the ARQ (1)/ACF (2) exchange with that Gatekeeper. The Gatekeeper shall return the Call Signalling Channel Transport Address of Endpoint 2 (called endpoint) in the ACF. Endpoint 1 then sends the Setup (3) message to Endpoint 2 using that Transport Address. If Endpoint 2 wishes to accept the call, it initiates an ARQ (5)/ACF (6) exchange with the Gatekeeper. It is possible that an ARJ (6) is

received by Endpoint 2, in which case it sends Release Complete to Endpoint 1. Endpoint 2 responds with the Connect (8) message which contains an H.245 Control Channel Transport Address for use in H.245 signalling.



**Figure 14/H.323 – Both endpoints registered, same Gatekeeper – Direct call signalling**

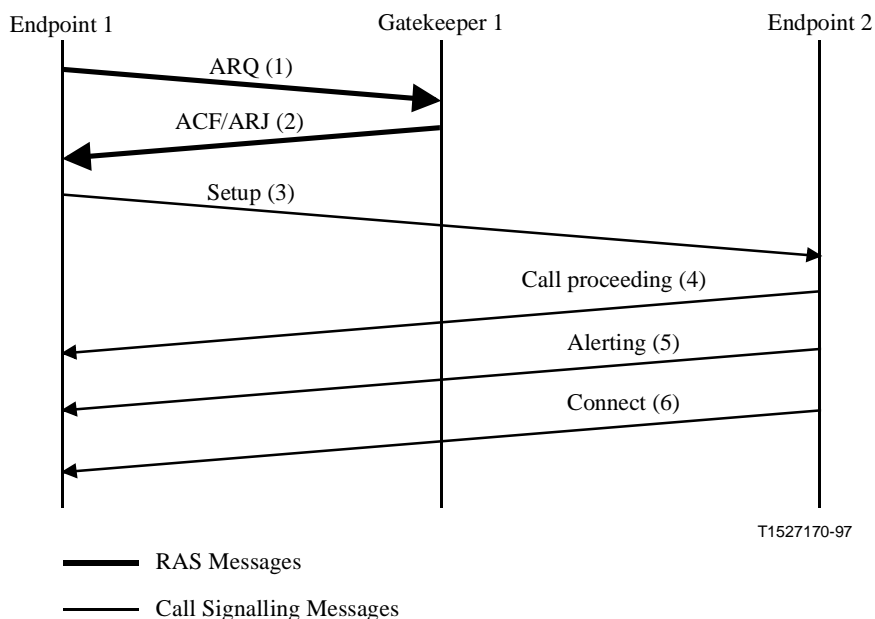
In the scenario shown in Figure 15, both endpoints are registered to the same Gatekeeper, and the Gatekeeper has chosen to route the call signalling. Endpoint 1 (calling endpoint) initiates the ARQ (1)/ACF (2) exchange with that Gatekeeper. The Gatekeeper shall return a Call Signalling Channel Transport Address of itself in the ACF. Endpoint 1 then sends the Setup (3) message using that Transport Address. The Gatekeeper then sends the Setup (4) message to Endpoint 2. If Endpoint 2 wishes to accept the call, it initiates an ARQ (6)/ACF (7) exchange with the Gatekeeper. It is possible that an ARJ (7) is received by Endpoint 2, in which case it sends Release Complete to the Gatekeeper. Endpoint 2 responds with the Connect (9) message which contains an H.245 Control Channel Transport Address for use in H.245 signalling. The Gatekeeper sends the Connect (10) message to Endpoint 1 which may contain the Endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.



**Figure 15/H.323 – Both endpoints registered, same Gatekeeper – Gatekeeper routed call signalling**

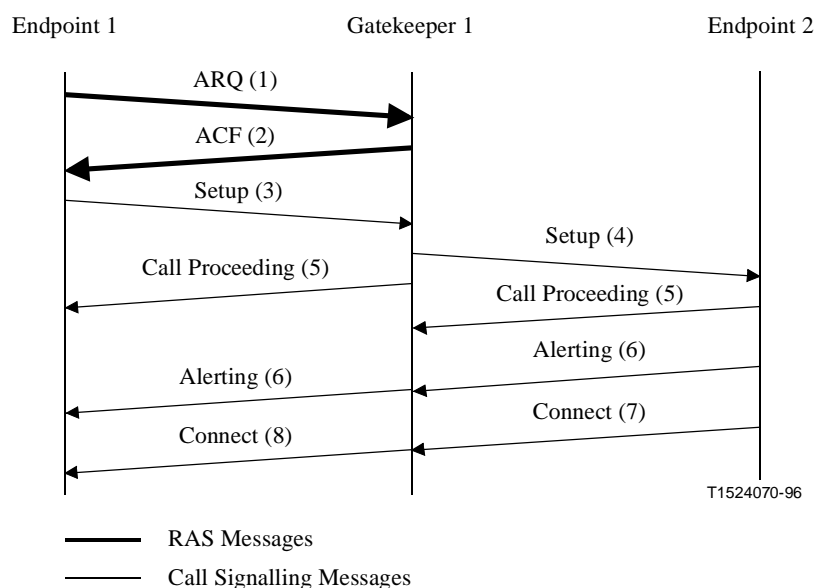
### 8.1.3 Only calling endpoint has Gatekeeper

In the scenario shown in Figure 16, Endpoint 1 (calling endpoint) is registered with a Gatekeeper, Endpoint 2 (called endpoint) is not registered with a Gatekeeper, and the Gatekeeper has chosen Direct Call Signalling. Endpoint 1 initiates the ARQ (1)/ACF (2) exchange with the Gatekeeper. Endpoint 1 then sends the Setup (3) message to Endpoint 2 using the well-known Call Signalling Channel Transport Address. If Endpoint 2 wishes to accept the call, it responds with the Connect (6) message which contains an H.245 Control Channel Transport Address for use in H.245 signalling.



**Figure 16/H.323 – Only calling endpoint registered – Direct call signalling**

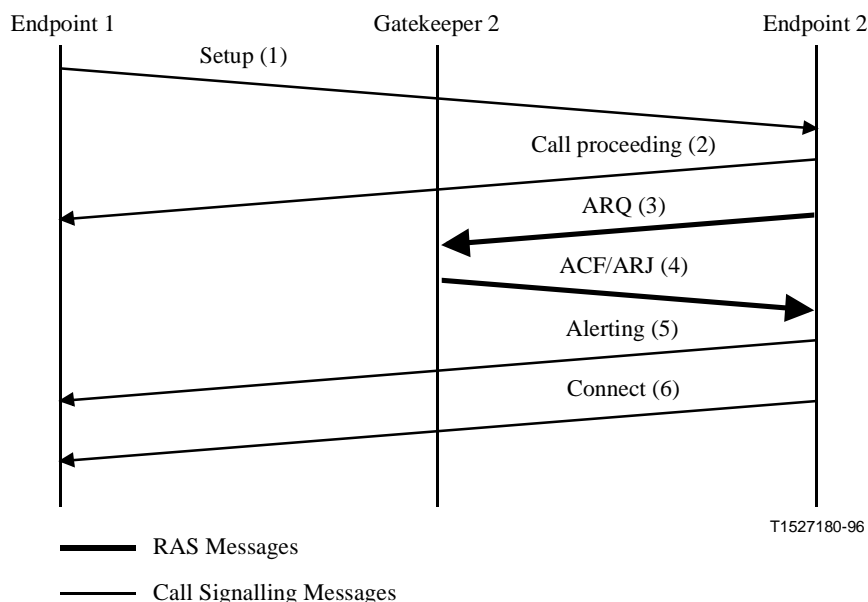
In the scenario shown in Figure 17, Endpoint 1 (calling endpoint) is registered with a Gatekeeper, Endpoint 2 (called endpoint) is not registered with a Gatekeeper, and the Gatekeeper has chosen to route the call signalling. Endpoint 1 (calling endpoint) initiates the ARQ (1)/ACF (2) exchange with that Gatekeeper. The Gatekeeper shall return a Call Signalling Channel Transport Address of itself in the ACF (2). Endpoint 1 then sends the Setup (3) message using that Transport Address. The Gatekeeper then sends the Setup (4) message to the well-known Call Signalling Channel Transport Address of Endpoint 2. If Endpoint 2 wishes to accept the call, it responds with the Connect (7) message which contains an H.245 Control Channel Transport Address for use in H.245 signalling. The Gatekeeper sends the Connect (8) message to Endpoint 1 which may contain the Endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.



**Figure 17/H.323 – Only calling endpoint registered – Gatekeeper routed call signalling**

#### 8.1.4 Only called endpoint has Gatekeeper

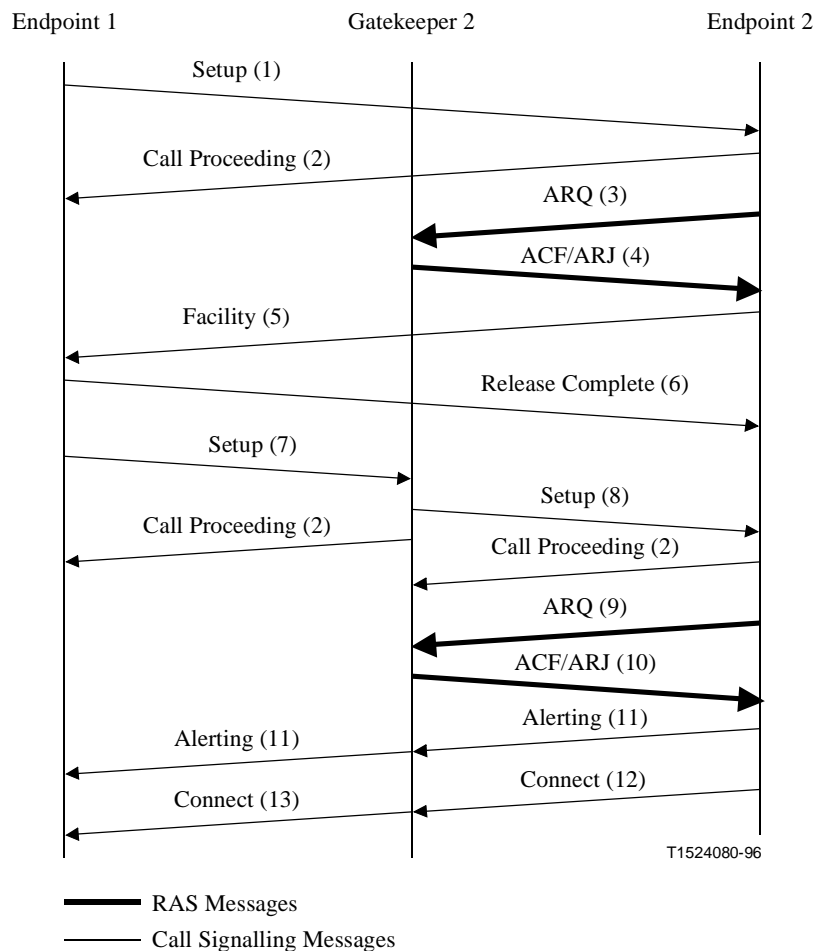
In the scenario shown in Figure 18, Endpoint 1 (calling endpoint) is not registered with a Gatekeeper, Endpoint 2 (called endpoint) is registered with a Gatekeeper, and the Gatekeeper has chosen Direct Call Signalling. Endpoint 1 sends the Setup (1) message to Endpoint 2 using the well-known Call Signalling Channel Transport Address. If Endpoint 2 wishes to accept the call, it initiates an ARQ (3)/ACF (4) exchange with the Gatekeeper. It is possible that an ARJ (4) is received by Endpoint 2, in which case it sends Release Complete to Endpoint 1. Endpoint 2 responds with the Connect (6) message which contains an H.245 Control Channel Transport Address for use in H.245 signalling.



**Figure 18/H.323 – Only called endpoint registered – Direct call signalling**

In the scenario shown in Figure 19, Endpoint 1 (calling endpoint) is not registered with a Gatekeeper, Endpoint 2 (called endpoint) is registered with a Gatekeeper, and the Gatekeeper has chosen to route the call signalling. Endpoint 1 (calling endpoint) sends a Setup (1) message to the well-known Call Signalling Channel Transport Address of Endpoint 2. If Endpoint 2 wishes to accept the call, it initiates the ARQ (3)/ACF (4) exchange with that Gatekeeper. If acceptable, the Gatekeeper shall return a Call Signalling Channel Transport Address of itself in the ARJ (4) with a cause code of **routeCallToGatekeeper**. Endpoint 2 replies to Endpoint 1 with a Facility (5) message containing the Call Signalling Transport Address of its Gatekeeper. Endpoint 1 then sends the Release Complete (6) message to Endpoint 2. Endpoint 1 sends a Setup (7) message to the Gatekeeper's Call Signalling Channel Transport Address. The Gatekeeper sends the Setup (8) message to Endpoint 2. Endpoint 2 initiates the ARQ (9)/ACF (10) exchange with that Gatekeeper. Endpoint 2 then responds with the Connect (12) message which contains its H.245 Control Channel Transport Address for use in H.245 signalling. The Gatekeeper sends the Connect (13) message to Endpoint 1 which may contain the Endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.

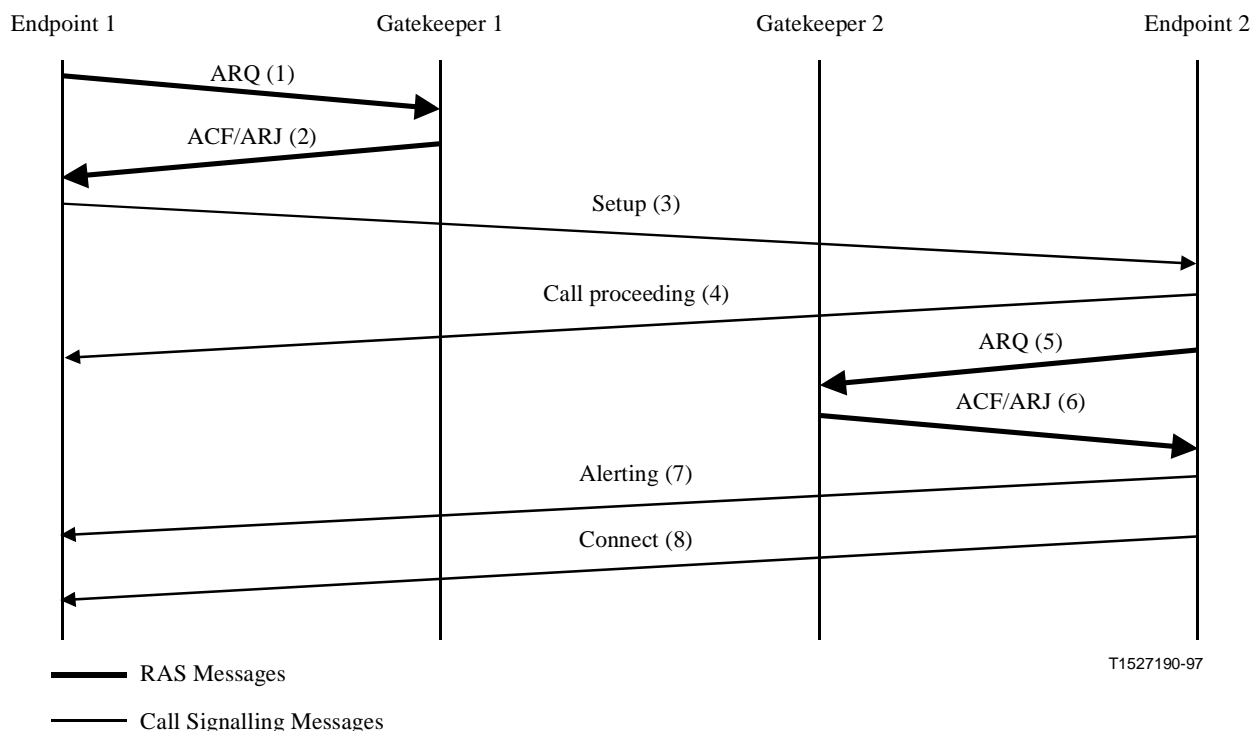




**Figure 19/H.323 – Only called endpoint registered – Gatekeeper routed call signalling**

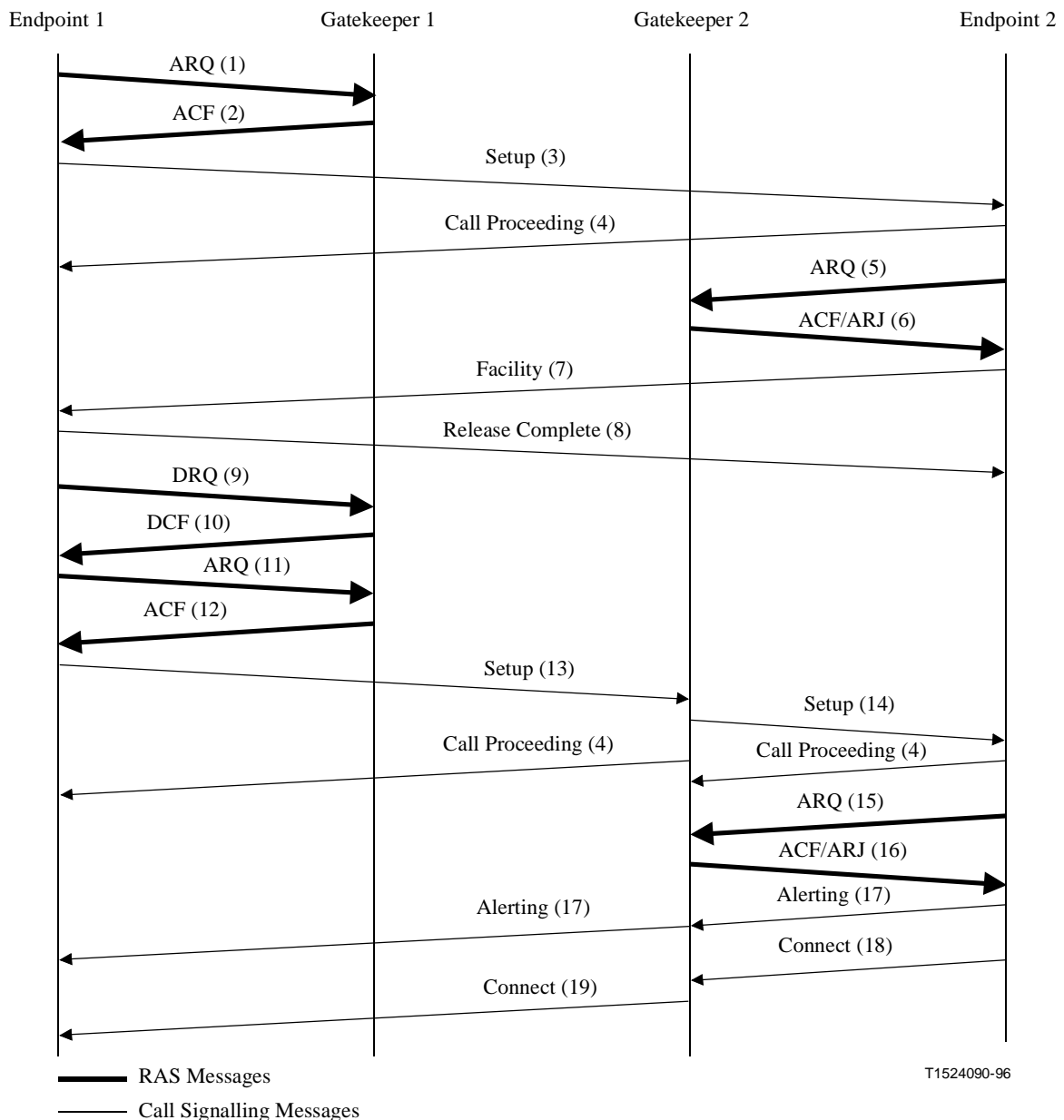
### 8.1.5 Both endpoints registered to different Gatekeepers

In the scenario shown in Figure 20, both endpoints are registered to different Gatekeepers, and both Gatekeepers choose Direct Call Signalling. Endpoint 1 (calling endpoint) initiates the ARQ (1)/ACF (2) exchange with Gatekeeper 1. Gatekeeper 1 may return the Call Signalling Channel Transport Address of Endpoint 2 (called endpoint) in the ACF if Gatekeeper 1 has a method of communicating with Gatekeeper 2. Endpoint 1 then sends the Setup (3) message to either the Transport Address returned by the Gatekeeper (if available) or to the well-known Call Signalling Channel Transport Address of Endpoint 2. If Endpoint 2 wishes to accept the call, it initiates an ARQ (5)/ACF (6) exchange with Gatekeeper 2. It is possible that an ARJ (6) is received by Endpoint 2, in which case it sends Release Complete to Endpoint 1. Endpoint 2 responds with the Connect (8) message which contains an H.245 Control Channel Transport Address for use in H.245 signalling.



**Figure 20/H.323 – Both endpoints registered – Both gatekeepers direct call signalling**

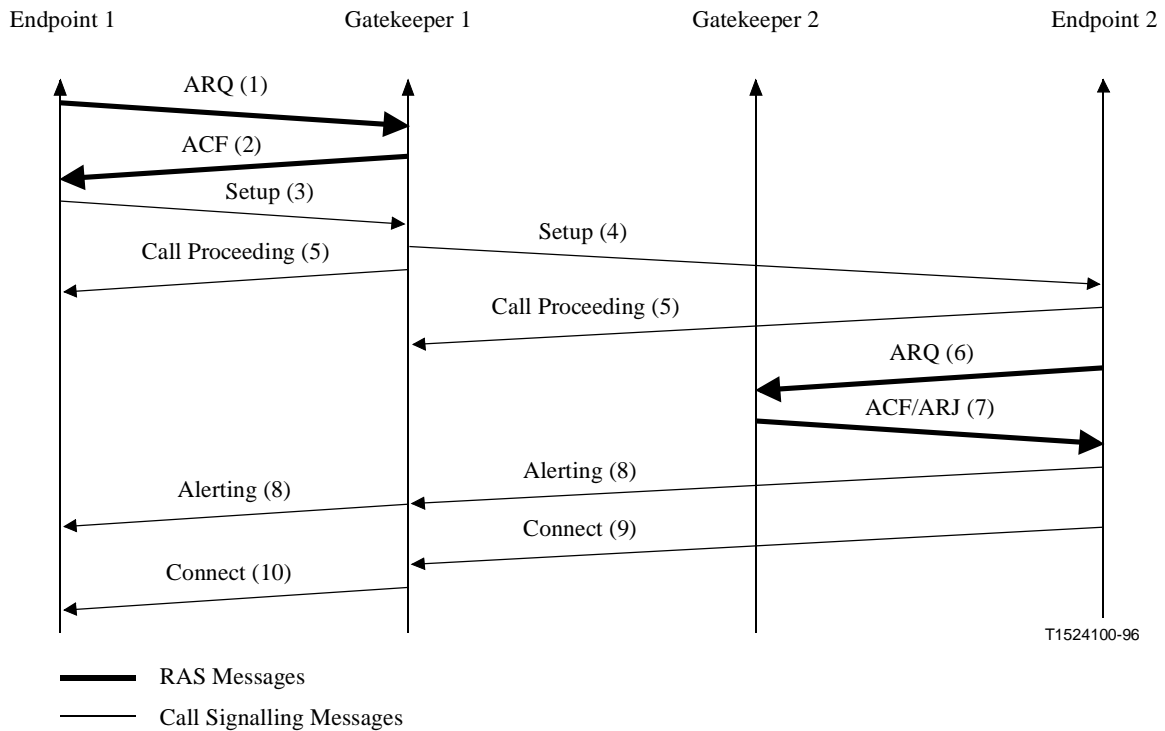
In the scenario shown in Figure 21, both endpoints are registered to different Gatekeepers, the calling endpoint's Gatekeeper chooses Direct Call Signalling, and the called endpoint's Gatekeeper chooses to route the call signalling. Endpoint 1 (calling endpoint) initiates the ARQ (1)/ACF (2) exchange with Gatekeeper 1. Gatekeeper 1 may return the Call Signalling Channel Transport Address of Endpoint 2 (called endpoint) in the ACF (2) if Gatekeeper 1 has a method of communicating with Gatekeeper 2. Endpoint 1 then sends the Setup (3) message to either the Transport Address returned by the Gatekeeper (if available) or to the well-known Call Signalling Channel Transport Address of Endpoint 2. If Endpoint 2 wishes to accept the call, it initiates the ARQ (5)/ACF (6) exchange with Gatekeeper 2. If acceptable, Gatekeeper 2 shall return a Call Signalling Channel Transport Address of itself in the ARJ (6) with a cause code of **routeCallToGatekeeper**. Endpoint 2 replies to endpoint 1 with a Facility (7) message containing the Call Signalling Transport Address of Gatekeeper 2. Endpoint 1 then sends the Release Complete (8) message to Endpoint 2. Endpoint 1 shall send a DRQ (9) to Gatekeeper 1 which responds with DCF (10). Endpoint 1 then initiates a new ARQ (11)/ACF (12) exchange with Gatekeeper 1. Endpoint 1 sends a Setup (13) message to the Gatekeeper's Call Signalling Channel Transport Address. Gatekeeper 2 sends the Set-up (14) message to Endpoint 2. Endpoint 2 initiates the ARQ (15)/ACF (16) exchange with Gatekeeper 2. Endpoint 2 then responds with the Connect (18) message which contains its H.245 Control Channel Transport Address for use in H.245 signalling. Gatekeeper 2 sends the Connect (19) message to Endpoint 1 which may contain the Endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper 2 H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.



**Figure 21/H.323 – Both endpoints registered – Direct/routed call signalling**

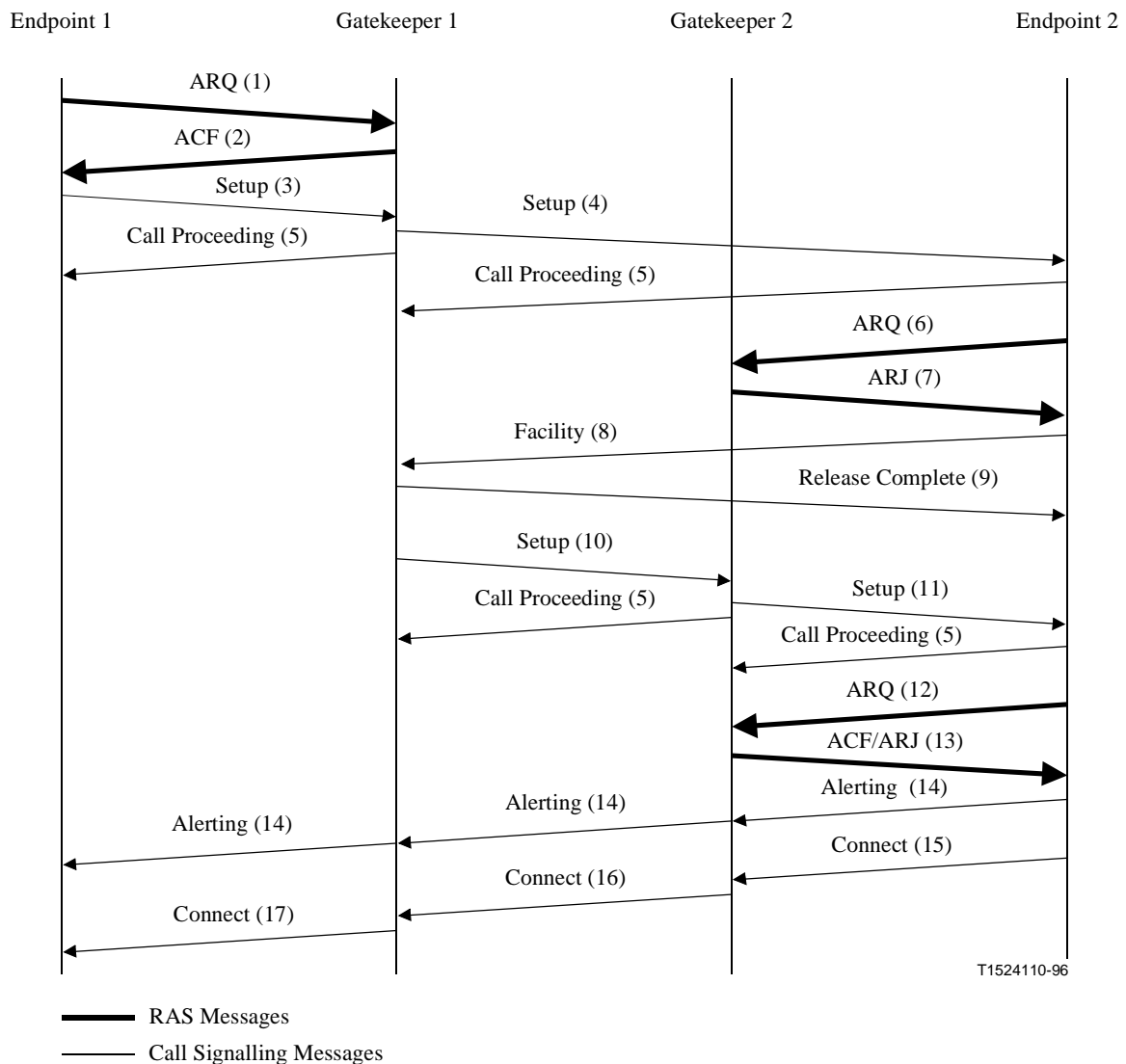
In the scenario shown in Figure 22, both endpoints are registered to different Gatekeepers, the calling endpoint's Gatekeeper chooses to route the call signalling, and the called endpoint's Gatekeeper chooses Direct Call Signalling. Endpoint 1 (calling endpoint) initiates the ARQ (1)/ACF (2) exchange with Gatekeeper 1. Gatekeeper 1 shall return a Call Signalling Channel Transport Address of itself in the ACF (2). Endpoint 1 then sends the Setup (3) message using that Transport Address. Gatekeeper 1 then sends the Setup (4) message containing its Call Signalling Channel Transport Address to the well-known Call Signalling Channel Transport Address of Endpoint 2. If Endpoint 2 wishes to accept the call, it initiates the ARQ (6)/ACF (7) exchange with Gatekeeper 2. It is possible that an ARJ (7) is received by Endpoint 2, in which case it sends Release Complete to Endpoint 1. Endpoint 2 responds to Gatekeeper 1 with the Connect (9) message which contains its H.245 Control Channel Transport Address for use in H.245 signalling. Gatekeeper 1 sends the Connect (10) message to Endpoint 1 which may contain the Endpoint 2 H.245 Control

Channel Transport Address, or a Gatekeeper 1 H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.



**Figure 22/H.323 – Both endpoints registered – Routed/direct call signalling**

In the scenario shown in Figure 23, both endpoints are registered to different Gatekeepers, and both Gatekeepers choose to route the call signalling. Endpoint 1 (calling endpoint) initiates the ARQ (1)/ACF (2) exchange with Gatekeeper 1. Gatekeeper 1 shall return a Call Signalling Channel Transport Address of itself in the ACF (2). Endpoint 1 then sends the Setup (3) message using that Transport Address. Gatekeeper 1 then sends the Setup (4) message to the well-known Call Signalling Channel Transport Address of Endpoint 2. If Endpoint 2 wishes to accept the call, it initiates the ARQ (6)/ACF (7) exchange with Gatekeeper 2. If acceptable, Gatekeeper 2 shall return a Call Signalling Channel Transport Address of itself in the ARJ (7) with a cause code of **routeCallToGatekeeper**. Endpoint 2 replies to Gatekeeper 1 with a Facility (8) message containing the Call Signalling Transport Address of Gatekeeper 2. Gatekeeper 1 then sends the Release Complete (9) message to endpoint 2. Gatekeeper 1 sends a Setup (10) message to Gatekeeper 2's Call Signalling Channel Transport Address. Gatekeeper 2 sends the Setup (11) message to Endpoint 2. Endpoint 2 initiates the ARQ (12)/ACF (13) exchange with Gatekeeper 2. Endpoint 2 then responds to Gatekeeper 2 with the Connect (15) message which contains its H.245 Control Channel Transport Address for use in H.245 signalling. Gatekeeper 2 sends the Connect (16) message to Gatekeeper 1 which may contain the Endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper 2 H.245 Control Channel Transport Address, based on whether the Gatekeeper 2 chooses to route the H.245 Control Channel or not. Gatekeeper 1 sends the Connect (17) message to Endpoint 1 which may contain the H.245 Control Channel Transport Address sent by Gatekeeper 2, or a Gatekeeper 1 H.245 Control Channel Transport Address, based on whether the Gatekeeper 1 chooses to route the H.245 Control Channel or not.



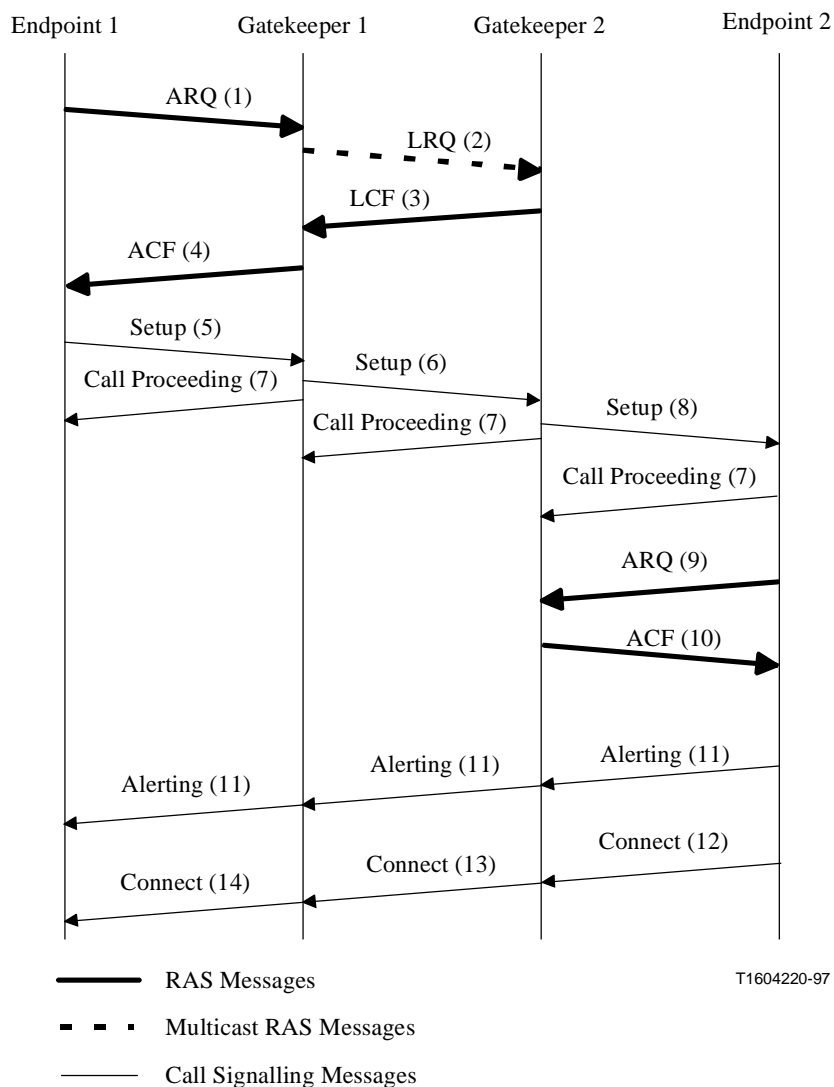
**Figure 23/H.323 – Both endpoints registered – Both Gatekeepers routing call signalling**

### 8.1.6 Optional Called Endpoint Signalling

The procedures defined in 8.1.4 and 8.1.5 show that when a called endpoint is registered to a Gatekeeper, a Setup message is initially sent to the called endpoint from the calling endpoint or the calling endpoint's Gatekeeper. If the called endpoint's Gatekeeper wishes to use the Gatekeeper routed call model, it returns its own Call Signalling Channel Transport Address in the ARJ. The called endpoint then uses the Facility message to redirect the call to the called endpoint's Gatekeeper's Call Signalling Transport Address. These procedures assume that the calling endpoint or calling endpoint's Gatekeeper only knows the called endpoints Call Signalling Channel Transport Address. This address may have been received in an LCF sent in response to an LRQ requesting the address of the called endpoint or it may be known through out-of-band methods.

If the called endpoint's Gatekeeper desires a Gatekeeper routed call model, it may return its own Call Signalling Transport Address in the LCF. This will allow the calling endpoint or calling endpoints Gatekeeper to send the Setup message directly to the called endpoints Gatekeeper, thus eliminating the redirection process.

An example of this scenario is shown in Figure 24. In this example, both endpoints are registered to different Gatekeepers, and both Gatekeepers choose to route the call signalling (similar to the case in Figure 23). Endpoint 1 (calling endpoint) sends an ARQ (1) to Gatekeeper 1. Gatekeeper 1 multicasts an LRQ (2) to locate called Endpoint 2. Gatekeeper 2 returns an LCF (3) with the Call Signalling Channel Transport Address of itself. Thus, Gatekeeper 1 will subsequently send a Setup (6) message to Gatekeeper 2's Call Signalling Channel Transport Address and Gatekeeper 2 will send a Setup (8) message to Endpoint 2. Endpoint 2 initiates the ARQ (9)/ACF (10) exchange with Gatekeeper 2. Endpoint 2 then responds to Gatekeeper 2 with the Connect (12) message which contains its H.245 Control Channel Transport Address for use in H.245 signalling. Gatekeeper 2 sends the Connect (13) message to Gatekeeper 1 which may contain the Endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper 2 H.245 Control Channel Transport Address, based on whether the Gatekeeper 2 chooses to route the H.245 Control Channel or not. Gatekeeper 1 sends the Connect (14) message to Endpoint 1 which may contain the H.245 Control Channel Transport Address sent by Gatekeeper 2, or a Gatekeeper 1 H.245 Control Channel Transport Address, based on whether the Gatekeeper 1 chooses to route the H.245 Control Channel or not.



**Figure 24/H.323 – Optional Called Endpoint Signalling**

### 8.1.7 Fast Connect Procedure

H.323 endpoints may establish media channels in a call using either the procedures defined in Recommendation H.245 or the "Fast Connect" procedure described in this subclause. The Fast Connect procedure allows the endpoints to establish a basic point-to-point call with as few as one round-trip message exchange, enabling immediate media stream delivery upon call connection.

The calling endpoint initiates the Fast Connect procedure by sending a Setup message containing the **fastStart** element to the called endpoint. The **fastStart** element consists of a sequence of **OpenLogicalChannel** structures describing media channels which the calling endpoint proposes to send and receive, including all of the parameters necessary to immediately open and begin transferring media on the channels. Details of the content and usage of the **fastStart** element are discussed below.

The called endpoint may refuse to use the Fast Connect procedure, either because it does not implement it or because it intends to invoke features that require use of the procedures defined in Recommendation H.245. Refusal of the Fast Connect procedure is accomplished by not returning the **fastStart** element in any Q.931 message up to and including the Connect message. Note that an endpoint may omit **fastStart** elements in a message prior to Connect, but then later return **fastStart** element in the Connect message thereby accepting the Fast Connect procedure. Refusing the Fast Connect procedure (or not initiating it) requires that H.245 procedures be used for capabilities exchange and opening of media channels.

When the called endpoint desires to proceed with the Fast Connect procedure, it sends a Q.931 message (Call Proceeding, Progress, Alerting, or Connect) containing a **fastStart** element selecting from amongst the **openLogicalChannel** proposals offered by the calling endpoint. Channels thus accepted are considered opened as though the usual H.245 **openLogicalChannel** and **openLogicalChannelAck** procedure had been followed. The called endpoint shall not include a **fastStart** element in any Q.931 message sent after the Connect message, and shall not include **fastStart** in any Q.931 message unless the Setup message contained a **fastStart** element.

The called endpoint may begin transmitting media (according to the channels opened) immediately after sending a Q.931 message containing **fastStart**. The calling endpoint must therefore be prepared to receive media on *any* of the receive channels it proposed in the Setup message, since it is possible for the media to be received prior to the Q.931 message indicating precisely which channels will be used. Once a Q.931 message containing **fastStart** is received by the calling endpoint, the calling endpoint may discontinue attempting to receive media on the channels for which proposals were not accepted by the called endpoint. Note that national requirements may prohibit called endpoints from transmitting media, or limit the nature of the content of the media stream, prior to transmission of a Connect message; it is the responsibility of the endpoint to comply with applicable requirements. If the calling endpoint sets the **mediaWaitForConnect** element to TRUE in the Setup message, then the called endpoint shall not send any media until after the Connect message is sent.

The calling endpoint may begin transmitting media (according to the channels opened) immediately upon receiving a Q.931 message containing **fastStart**. Thus, the called endpoint must be prepared to immediately receive media on the channels it accepted in the Q.931 message containing **fastStart**. Note that national requirements may prohibit calling endpoints from transmitting media prior to receipt of a Connect message; it is the responsibility of the endpoint to comply with applicable requirements.

#### 8.1.7.1 Proposal, selection and opening of media channels

The calling endpoint may propose multiple media channels, or multiple alternative sets of characteristics for each media channel, by encoding multiple **OpenLogicalChannel** structures within the **fastStart** element of the Setup message. Each **OpenLogicalChannel** structure within the **fastStart** element describes exactly one unidirectional media channel.

In the Setup message, each **OpenLogicalChannel** is a *proposal* to establish a media channel. **OpenLogicalChannel** proposals are included in the **fastStart** element in order of preference, with the most preferred alternatives listed first in the **fastStart** sequence; proposals to open audio channels shall be listed prior to channels for any other media types. In the Q.931 message containing **fastStart** sent in response to Setup, each **OpenLogicalChannel** is an *acceptance* of a proposed media channel and indicates the channels that are established and can immediately be used for media transmission.

In an **openLogicalChannel** which proposes a channel for transmission from the calling endpoint to the called endpoint, the **forwardLogicalChannelParameters** element shall contain parameters specifying the characteristics of the proposed channel, and the **reverseLogicalChannelParameters** element shall be omitted. Each such **OpenLogicalChannel** structure shall have a unique **forwardLogicalChannelNumber** value. Alternative proposals for the same transmit channel shall contain the same **sessionID** value in **H2250LogicalChannelParameters**. The **mediaChannel** element shall be omitted in the proposal; it will be provided by the called endpoint should the proposal be accepted. The other **H2250LogicalChannelParameters** and **dataType** shall be set to correctly describe the transmit capabilities of the calling endpoint associated with this proposed channel. The calling endpoint may choose not to propose any channels for transmission from the calling endpoint to the called endpoint, such as if it desires to use H.245 procedures later to establish such channels.

In an **openLogicalChannel** which proposes a channel for transmission from the called endpoint to the calling endpoint, the **reverseLogicalChannelParameters** element shall be included and contain parameters specifying the characteristics of the proposed channel. The **forwardLogicalChannelParameters** element must also be included (because it is not optional), with the **dataType** element set to **nullData**, **multiplexParameters** set to **none**, and all optional elements omitted. Alternative proposals for the same receive channel shall contain the same **sessionID** value in **H2250LogicalChannelParameters**. All alternative **OpenLogicalChannel** structures, that propose a channel for transmission from the called endpoint to the calling endpoint, shall contain the same **sessionID** and the same **mediaChannel** value. The other **H2250LogicalChannelParameters** and **dataType** within **reverseLogicalChannelParameters** shall be set to correctly describe the receive capabilities of the calling endpoint associated with this proposed channel. The calling endpoint may choose not to propose any channels for transmission from the called endpoint to the calling endpoint, such as if it desires to use H.245 procedures later to establish such channels.

In the Setup message, each **openLogicalChannel** which proposes a channel for transmission from the called endpoint to the calling endpoint, shall contain **mediaControlChannel** element (indicating the RTCP channel going in the same direction) into the **H2250LogicalChannelParameters** element of the **reverseLogicalChannelParameters** structure. All **mediaControlChannel** elements inserted by the calling endpoint for the same **sessionID** for both directions shall have the same value.

Upon receipt of a Setup message containing **fastStart**, determining that it is willing to proceed with the Fast Connect procedure, and reaching the point in the connection at which it is ready to begin media transmission, the called endpoint shall choose from amongst the proposed **OpenLogicalChannel** structures containing **reverseLogicalChannelParameters** elements for each media type it wants to transmit, and from amongst the proposed **OpenLogicalChannel** structures specifying **forwardLogicalChannelParameters** (and omitting **reverseLogicalChannelParameters**) for each media type it wants to receive. If alternative proposals are presented, only one **OpenLogicalChannel** structure shall be selected from amongst each alternative set; alternatives within a set have the same **sessionID**. The called endpoint accepts a proposed channel by returning the corresponding **OpenLogicalChannel** structure in any Q.931 message sent in response to Setup, up to and including Connect. The called endpoint may choose not to open media flow in a particular direction or of a particular media type by not including a corresponding **OpenLogicalChannel** structure in the **fastStart** element of the Q.931 response.

When accepting a proposed channel for transmission from called endpoint to calling endpoint, the called endpoint shall return the corresponding **OpenLogicalChannel** structure to the



calling endpoint, inserting a unique **forwardLogicalChannelNumber** into the **forwardLogicalChannelParameters** structure and a valid **mediaControlChannel** element (indicating the reverse RTCP channel) into the **H2250LogicalChannelParameters** element of the **reverseLogicalChannelParameters** structure. All **mediaControlChannel** elements inserted by the called endpoint for the same **sessionID** for both directions shall have the same value. The called endpoint may begin transmitting media on the accepted channel according to the parameters specified in **reverseLogicalChannelParameters** immediately after sending the Q.931 response containing **fastStart**, unless **mediaWaitForConnect** was set to TRUE in which case it must wait until after sending the Connect message.

When accepting a proposed channel for transmission from the calling endpoint to the called endpoint, the called endpoint shall return the corresponding **OpenLogicalChannel** structure to the calling endpoint. The called endpoint shall insert valid **mediaChannel** and **mediaControlChannel** fields (indicating the RTCP channel going in the same direction) into the **h2250LogicalChannelParameters** element of the **forwardLogicalChannelParameters** structure. The called endpoint shall then prepare to immediately receive media flow according to the parameters specified in **forwardLogicalChannelParameters**. The calling endpoint may begin transmitting media on the accepted and opened channels upon receipt of the Q.931 response containing **fastStart**, and may release any resources allocated to reception on proposed channels that were not accepted.

#### **8.1.7.2 Switching to H.245 procedures**

After establishment of a call using the Fast Connect procedure, either endpoint may determine that it is necessary to invoke call features that require the use of H.245 procedures. Either endpoint may initiate the use of H.245 procedures at any point during the call, using tunnelling as described in 8.2.1 (if **h245Tunnelling** remains enabled) or a separate H.245 connection. The process for switching to a separate H.245 connection is described in 8.2.3.

When a call is established using the Fast Connect procedure, both endpoints shall keep the Q.931 Call Signalling Channel open until either the call is terminated or a separate H.245 connection is established.

When H.245 procedures are activated, all mandatory procedures of H.245 that normally occur upon initiation of an H.245 connection shall be completed prior to initiation of any additional H.245 procedures. The media channels that were established in the Fast Connect procedure are "inherited" as though they had been opened using normal H.245 **openLogicalChannel** and **openLogicalChannelAck** procedures. In order for such "inheritance" to succeed, media sessions opened during the Fast Connect procedure shall use only well-known **sessionID** values defined in Recommendation H.245.

#### **8.1.7.3 Terminating a call**

If a call connected using the Fast Connect procedure continues to completion without initiation of H.245 procedures, then the call may be terminated by either endpoint sending a Q.931 Release Complete message. If H.245 procedures are initiated during the call, then the call is terminated as described in 8.5.

If a separate H.245 connection has not been established and the Q.931 Call Signalling Channel is terminated, the call shall also be terminated.

### **8.1.8 Call setup via gateways**

#### **8.1.8.1 Gateway in-bound call setup**

When an external terminal calls a network endpoint via the Gateway, call setup between the Gateway and the network endpoint proceeds the same as the endpoint-to-endpoint call setup. The Gateway may need to issue Call Proceeding messages to the external terminal while establishing the call on the network.

A Gateway which cannot directly route an incoming SCN call to an H.323 endpoint shall be able to accept two-stage dialing. For Gateways to H.320 networks (also H.321, H.322 and H.310 in H.321 mode), the Gateway shall accept SBE numbers from the H.320 terminal. Optionally, Gateways to H.320 networks may support the TCS-4 and IIS BAS codes to retrieve the H.323 dialing information after a H.320 call has been established. For Gateways to H.310 native mode and H.324 networks, the Gateway shall accept H.245 **userInputIndication** messages from the H.324 terminal. In these two cases, support of DTMF is optional. For Gateways to speech-only terminals, the Gateway shall accept DTMF numbers from the speech-only terminal. These numbers will indicate a second stage dialing number to access the individual endpoint on the network.

#### **8.1.8.2 Gateway out-bound call setup**

When a network endpoint calls an external terminal via the Gateway, call setup between the network endpoint and the Gateway proceeds the same as the endpoint-to-endpoint call setup. The Gateway will receive the destination E.164 or **partyNumber** address in the Setup message. It will then use this address to place the out-bound call. The Gateway may issue Call Proceeding messages to the network endpoint while establishing the outgoing call.

A Gateway should send Call Proceeding message after it receives the Setup message (or after it receives ACF) if it expects more than 4 seconds to elapse before it can respond with Alerting, Connect, or Release Complete.

The Progress Indicator information element is used to indicate that inter-networking is occurring. The Gateway shall issue a Progress indicator information element within the Alerting, Call Proceeding or Connect messages. This information may also be sent in a Progress message.

The network endpoint shall send all E.164 or **partyNumber** addresses that it is calling in the Setup message. For example, a six B-channel call on the ISDN will require six E.164 or **partyNumber** addresses in the Setup message. The Gateway shall respond to the Setup message with a Connect or Release Complete message as well as Alerting, Call Proceeding, or Progress messages. Failure of the SCN call shall be reported to the network endpoint in the Release Complete message. The use of multiple CRV values and multiple Setup messages is for further study. Addition of channels on the SCN during a call is for further study.

A network endpoint which is registered with a Gatekeeper should request sufficient call bandwidth in the ARQ message for the aggregate of all SCN calls. If sufficient call bandwidth was not requested in the ARQ message, the procedures of 8.4.1, Bandwidth Changes, shall be followed in order to obtain additional call bandwidth.

The Gateway may advance to Phase B after placing the first call on the SCN. Additional calls for the additional SCN E.164 or **partyNumber** numbers may be placed after the capability exchange with the Gateway and establishment of audio communications with the SCN endpoint.

#### **8.1.9 Call setup with an MCU**

For Centralized Multipoint Conferences, all endpoints exchange call signalling with the MCU. Call setup between an endpoint and the MCU proceeds the same as the endpoint-to-endpoint call setup scenarios of 8.1.1 through 8.1.5. The MCU may be the called endpoint or the calling endpoint.

In a Centralized Multipoint Conference, the H.245 Control Channel is opened between the endpoints and the MC within the MCU. The audio, video, and data channels are opened between the endpoints and the MP within the MCU. In a Decentralized Multipoint Conference, the H.245 Control Channel is open between the endpoint and the MC (there may be many such H.245 Control Channels, one for each call). The Audio and Video Channels should be multicast to all endpoints in the conference. The Data Channel shall be opened with the Data MP.

In an ad hoc Multipoint Conference where the endpoints do not contain an MC and the Gatekeeper would like to provide an Ad Hoc multipoint service for the endpoints, the H.245 Control Channel

may be routed through the Gatekeeper. Initially, the H.245 Control Channel would be routed between the endpoints through the Gatekeeper. When the conference switches to multipoint, the Gatekeeper may connect the endpoints to an MC associated with the Gatekeeper.

In an ad hoc Multipoint Conference where one or both of the endpoints contains an MC, the normal call setup procedures defined in 8.1.1 through 8.1.5 are used. These procedures may apply even if an endpoint which contains an MC is actually a MCU. The master-slave determination procedure is used to determine which MC will be the Active MC for the conference.

#### 8.1.10 Call forwarding

An endpoint wishing to forward a call to another endpoint may issue a Facility message indicating the address of the new endpoint. The endpoint receiving this Facility indication should send a Release Complete and then restart the Phase A procedures with the new endpoint.

#### 8.1.11 Broadcast call setup

Call setup for loosely controlled Broadcast and Broadcast Panel conferences shall follow the procedures defined in Recommendation H.332.

#### 8.1.12 Overlapped Sending

H.323 entities can optionally support overlap sending. If a Gatekeeper is present, and overlap sending is being used, endpoints should send an ARQ message to the Gatekeeper each time some new addressing information is input. The endpoint shall place the total cumulative addressing information into the **destinationInfo** field each time an ARQ message is sent. If there is insufficient addressing information in the ARQ, the Gatekeeper should respond with an ARJ with the **reason** set to **incompleteAddress**. This indicates that the endpoint should send another ARQ when more addressing information is available. When a Gatekeeper has sufficient addressing information to assign a suitable **destCallSignalAddress**, it shall return an ACF. Note that this does not necessarily mean that the addressing information is complete. If the Gatekeeper sends an ARJ with **AdmissionRejectReason** set to something other than **incompleteAddress**, the call setup process shall be aborted.

When an endpoint has a suitable **destCallSignalAddress**, it shall send a Setup message with the **canOverlapSend** field assigned according to whether it is capable of supporting the overlap sending procedures. If a remote entity receives a Setup message with an incomplete address and the **canOverlapSend** field set to TRUE, it should initiate overlap sending procedures by returning the Setup Acknowledge message. Additional addressing information should be sent using Information messages. If the address is incomplete and the **canOverlapSend** field set to FALSE, the remote entity should send Release Complete. Note that Gateways should not transfer Setup Acknowledge messages from the SCN to H.323 endpoints that have not indicated that they can support overlap sending procedures as the desired result may not be achieved.

#### 8.1.13 Call setup to conference alias

Alias addresses (see 7.1.3) may be used to represent a conference at an MC. The procedures in the preceding subclauses apply, except as noted here.

##### 8.1.13.1 Joining to a Conference Alias, with no Gatekeeper

Endpoint 1 (calling endpoint) sends the Setup (1) message (see Figure 13) to the well-known Call Signalling Channel TSAP Identifier of Endpoint 2 (the MC). The Setup message includes the following fields:

<b>destinationAddress</b>	= <b>conferenceAlias</b>
<b>destCallSignalAddress</b>	= <b>MC(U) transport address</b>
<b>conferenceID</b>	= <b>0 (since the CID is unknown)</b>
<b>conferenceGoal</b>	= <b>join</b>

Endpoint 2 responds with the Connect (4) message, which contains:

<b>h245Address</b>	= Transport Address for H.245 signalling
<b>conferenceID</b>	= CID for the conference

### 8.1.13.2 Joining to a Conference Alias, with Gatekeeper

Endpoint 1 (calling endpoint) initiates the ARQ (1)/ACF (2) exchange (reference Figure 14) with the Gatekeeper. The ARQ contains:

<b>destinationInfo</b>	= conferenceAlias
<b>callIdentifier</b>	= some value N
<b>conferenceID</b>	= 0 (since the CID is unknown)

The Gatekeeper shall return the Call Signalling Channel Transport Address of Endpoint 2 (called endpoint, containing the MC) in the ACF. Endpoint 1 then sends the Setup (3) message to Endpoint 2 using that Transport Address and the following fields:

<b>destinationAddress</b>	= conferenceAlias
<b>destCallSignalAddress</b>	= address supplied by ACF
<b>conferenceID</b>	= 0
<b>conferenceGoal</b>	= join

Ultimately, Endpoint 2 returns a Connect message with following fields:

<b>h245Address</b>	= Transport Address for H.245 signalling
<b>conferenceID</b>	= CID for the conference

Endpoint 1 completes the call by informing its Gatekeeper of the correct CID. Endpoint 1 sends an IRR to the Gatekeeper with the following fields:

<b>callIdentifier</b>	= same value N as used in the first ARQ
<b>conferenceID</b>	= original CID from endpoint 1
<b>substituteConferenceIDs</b>	= CID from endpoint 2

### 8.1.13.3 Create or Invite with a Conference Alias

Endpoint 1 (calling endpoint) may send a Setup message to Endpoint 2. The Setup message includes the following fields:

<b>destinationAddress</b>	= conferenceAlias
<b>destCallSignalAddress</b>	= MC(U) transport address
<b>conferenceID</b>	= CID of the conference
<b>conferenceGoal</b>	= create or invite

Endpoint 2 responds with the Connect message, which contains:

<b>h245Address</b>	= Transport Address for H.245 signalling
<b>conferenceID</b>	= CID for the conference

### 8.1.13.4 Consideration for Version 1 Endpoints

When an H.323 entity (endpoint or MCU) receives a Setup message from a Version 1 entity and the **destinationAddress** matches one of its conferences aliases, then it shall ignore the **conferenceGoal** and treat the Setup request as a join request.

When a Gatekeeper receives an ARQ, from a Version 1 entity and the **destinationInfo** matches one of its conferences aliases, then it shall ignore the **conferenceID** field. Likewise, when an H.323 entity receives a Setup message from a Version 1 entity and the **destinationAddress** matches one of its conferences aliases, then it shall ignore the **conferenceID**.

These provisions allow a Version 1 endpoint to call a conference Alias.

## 8.2 Phase B – Initial communication and capability exchange

Once both sides have exchanged call setup messages from Phase A, the endpoints shall, if they plan to use H.245, establish the H.245 Control Channel. The procedures of Recommendation H.245 are used over the H.245 Control Channel for the capability exchange and to open the media channels.

NOTE – Optionally, the H.245 Control Channel may be set up by the called endpoint on receipt of Setup, and by the calling endpoint on receipt of Alerting or Call Proceeding. In the event that Connect does not arrive, or an endpoint sends Release Complete, the H.245 Control Channel shall be closed.

Endpoints shall support the capabilities exchange procedure of H.245 as described in 6.2.8.1.

Endpoint system capabilities are exchanged by transmission of the H.245 **terminalCapabilitySet** message. This capability message shall be the first H.245 message sent. If prior to successful completion of terminal capability exchange, any other procedure fails, (i.e. rejected, not understood, not supported) then the initiating endpoint should initiate and successfully complete terminal capability exchange before attempting any other procedure. An endpoint which receives a **terminalCapabilitySet** message from a peer prior to initiating capabilities exchange shall respond as required by 6.2.8.1, and should initiate and successfully complete capabilities exchange with that peer prior to initiating any other procedure.

Endpoints shall support the master-slave determination procedure of H.245 as described in 6.2.8.4. In cases where both endpoints in a call have MC capability, the master-slave determination is used for determining which MC will be the Active MC for the conference. The Active MC may then send the **mcLocationIndication** message. The procedure also provides master-slave determination for opening bidirectional channels for data.

Master-slave determination shall be advanced (by sending either **MasterSlaveDetermination** or **MasterSlaveDeterminationAck** as appropriate) in the first H.245 message after Terminal Capability Exchange has been initiated.

If the initial capability exchange or master-slave determination procedures fail, these should be retried at least two additional times before the endpoint abandons the connection attempt and proceeds to Phase E.

Following successful completion of the requirements of Phase B, the endpoints shall proceed directly to the desired operating mode, normally Phase C.

### 8.2.1 Encapsulation of H.245 messages within Q.931 messages

In order to conserve resources, synchronize call signalling and control, and reduce call setup time, it may be desirable to convey H.245 messages within the Q.931 Call Signalling Channel instead of establishing a separate H.245 channel. This process, known as "encapsulation" or "tunnelling" of H.245 messages, is accomplished by utilizing the **h245Control** element of **h323\_uu\_pdu** on the Call Signalling Channel, copying an encoded H.245 message as an octet string.

When tunnelling is active, one or more H.245 messages can be encapsulated in any Q.931 message. If tunnelling is being utilized and there is no need for transmission of a Q.931 message at the time an H.245 message must be transmitted, then a Facility message shall be sent with **h323-message-body** set to **empty**.

A calling endpoint capable of and willing to use H.245 encapsulation shall set the **h245Tunnelling** element to TRUE in the Setup message and any subsequent Q.931 messages it sends so long as it

desires tunnelling to remain active. A called endpoint capable of and willing to use H.245 encapsulation shall set the **h245Tunnelling** element to TRUE in the first Q.931 message sent in response to Setup and in every subsequent Q.931 message it sends so long as it desires Tunnelling to remain active. The called endpoint shall not set **h245Tunnelling** to TRUE in any Q.931 response (and tunnelling remains disabled) unless it was TRUE in the Setup message to which it is responding. If **h245Tunnelling** is not set to TRUE in any Q.931 message, then tunnelling is disabled from that point for the duration of the call and a separate H.245 connection shall be established when and if H.245 procedures are invoked.

The calling endpoint may include tunnelled H.245 messages in the Setup message; it must also set the **h245Tunnelling** element to TRUE. If the called endpoint does not set **h245Tunnelling** to TRUE in the first Q.931 message sent in response to Setup, then the calling endpoint shall assume that the H.245 messages it had encapsulated in Setup were ignored by the called endpoint and repeat them, as necessary, after the separate H.245 channel is established. The called endpoint, if it sets **h245Tunnelling** to TRUE, may also include encapsulated H.245 messages in the first and subsequent Q.931 messages.

The calling endpoint shall *not* include both a **fastStart** element and encapsulated H.245 messages in **h245Control** in the same Setup message, since the presence of the encapsulated H.245 messages would override the Fast Connect procedure. A calling endpoint may, however, include both a **fastStart** element and set **h245Tunnelling** to TRUE within the same Setup message; likewise, a called endpoint may include **fastStart** and set **h245Tunnelling** to TRUE within the same Q.931 response. In this case, the Fast Connect procedures are followed, and the H.245 connection remains "unestablished" until actual transmission of the first tunnelled H.245 message or opening of the separate H.245 connection. The sending of encapsulated H.245 messages or the initiation of the separate H.245 connection by either endpoint prior to the sending of a Q.931 message containing **fastStart** by the called endpoint terminates the Fast Connect procedures.

When H.245 encapsulation is being used, both endpoints shall keep the Q.931 Call Signalling Channel open until either the call is terminated or a separate H.245 connection is established.

When an endpoint receives an **h245control** element encapsulating more than one H.245 PDU, the encapsulated H.245 PDUs shall be processed (i.e. provided to higher layers) sequentially by order of increasing offset from the beginning of the H.225.0 message.

### 8.2.2 Tunnelling through intermediate signalling entities

Entities in the signalling path such as gatekeepers may perform functions such as divert on no-reply or other advanced call control that results in representing to an endpoint a Q.931 call state that is different from the actual call state at the other endpoint. Such intermediate entities shall ensure that H.245 messages encapsulated in Q.931 messages are forwarded to the other endpoint even if the Q.931 message in which the H.245 message is encapsulated would be consumed and not forwarded to the other endpoint. This is accomplished by transferring the encapsulated H.245 message into a Facility message with the **h323-message-body** set to **empty**. For example, if a gatekeeper has already sent a Connect message to a calling endpoint and later receives a Connect message from a called endpoint that contains an encapsulated H.245 message, it must forward the H.245 message using a Facility message.

### 8.2.3 Switching to a separate H.245 connection

When H.245 encapsulation or Fast Connect is being used, either endpoint may choose to switch to using the separate H.245 connection at any time. In order to facilitate initiation of the separate H.245 connection by either endpoint, each endpoint may include **h245Address** in any Q.931 message it sends during the call. If, at the time an endpoint deems it necessary to initiate the separate H.245 connection, it finds that it has not yet received the **h245Address** of the other endpoint, the endpoint shall transmit a Facility message with a **FacilityReason** of **startH245** and provide its H.245 address in the **h245Address** element. An endpoint receiving a Facility message with a **facilityReason** of

**startH245** which has not already independently initiated the separate H.245 channel shall open the H.245 channel using the **h245Address** specified. Use of the separate H.245 connection is initiated by opening the H.245 TCP connection, and accepted by acknowledgement of the H.245 TCP connection.

If tunnelling was being used, the endpoint initiating the separate H.245 connection shall not send any further tunnelled H.245 messages on the Call Signalling Channel, and shall send no H.245 messages on the separate H.245 connection until the establishment of the TCP connection is acknowledged. The endpoint acknowledging opening of the separate H.245 connection shall not send any further tunnelled H.245 messages on the Call Signalling Channel after acknowledging the opening of the separate H.245 connection. Because of the possibility that H.245 messages have already been sent and are in transit when the separate H.245 channel is initiated, endpoints shall continue to receive and correctly process tunnelled H.245 messages until a Q.931 message is received with the **h245Tunnelling** flag set to FALSE; responses to such "late" tunnelled H.245 messages or acknowledgement of such messages shall be sent on the separate H.245 connection after it is established. Once a separate H.245 connection has been established, it is not possible to switch back to using tunnelling.

In the event that both endpoints simultaneously initiate the separate H.245 connection, the endpoint with the numerically smaller **h245Address** shall close the TCP connection it opened and use the connection opened by the other endpoint. For purposes of comparing the numeric values of **h245Address**, each octet of the address shall be individually compared beginning with the first octet of the OCTET STRING and continuing through the OCTET STRING left to right until unequal numeric octet values are found. Comparison shall first be performed on the network-layer address element of **h245Address**, and, if found to be equal, then on the transport (port) address element.

### 8.3 Phase C – Establishment of audiovisual communication

Following the exchange of capabilities and master-slave determination, the procedures of Recommendation H.245 shall then be used to open logical channels for the various information streams. The audio and video streams, which are transmitted in the logical channels setup in H.245, are transported over dynamic TSAP Identifiers using an unreliable protocol (see Recommendation H.225.0). Data communications which is transmitted in the logical channels setup in H.245, are transported using a reliable protocol (see Recommendation H.225.0).

The **openLogicalChannelAck** message returns, or the **reverseLogicalChannelParameters** of the **openLogicalChannel** request contains, the Transport Address that the receiving endpoint has assigned to that logical channel. The transmitting channel shall send the information stream associated with the logical channel to that Transport Address.

Following the opening of logical channels for audio and video, one **h2250MaximumSkewIndication** message shall be sent by the transmitter for each associated audio and video pair.

#### 8.3.1 Mode changes

During a session, the procedures for changing channel structure, capability, receive mode, etc., shall be carried out as defined in Recommendation H.245. Appendix V/H.245 contains a procedure for changing modes on a logical channel which may minimize the disruption of the audio.

#### 8.3.2 Exchange of video by mutual agreement

The indication **videoIndicateReadyToActivate** is defined in Recommendation H.245. Its use is optional, but when used the procedure shall be as follows:

Endpoint 1 has been set so that video is not transmitted unless, and until, Endpoint 2 has also indicated readiness to transmit video. Endpoint 1 shall send the indication **videoIndicateReadyToActivate** when the initial capability exchange has been completed, but shall not

transmit a video signal until it has received either **videoIndicateReadyToActivate** or incoming video from Endpoint 2.

An endpoint which has not been set in this optional way is not obliged to wait until receipt of **videoIndicateReadyToActivate** or video before initiating its video transmission.

### 8.3.3 Media stream address distribution

In unicast, the endpoint shall open logical channels to the MCU or other endpoint. Addresses are passed in the **openLogicalChannel** and **openLogicalChannelAck**.

In multicast, the multicast addresses are assigned by the MC and distributed to the endpoints in the **communicationModeCommand**. It is the responsibility of the MC to allocate and assign unique multicast addresses. The endpoint shall signal an **openLogicalChannel** to the MC with the assigned multicast address. The MC shall forward the **openLogicalChannel** to each receiving endpoint. In cases where media from multiple endpoints are transmitted on a single session (e.g. single multicast address), the MC shall open a logical channel to each endpoint receiving media from an endpoint in the conference.

In cases where an endpoint joins a conference after the initial **communicationModeCommand** has been transmitted, it is the responsibility of the MC to send an updated **communicationModeCommand** to the new endpoint and to open the appropriate logical channels for media sourced from the new endpoint. In cases where an endpoint leaves the conference after the initial **communicationModeCommand** has been transmitted, it is the responsibility of the MC to close the appropriate logical channels which were being sourced from the endpoint which left the conference.

In multi-unicast, the endpoint must open logical channels to each of the other endpoints. The **openLogicalChannel** is sent to the MC and shall contain the terminal number of the endpoint for which the channel is intended. The endpoint can match a **openLogicalChannelAck** by the **forwardLogicalChannelNumber**.

### 8.3.4 Correlation of media streams in multipoint conferences

The following method shall be used to associate a logical channel with an RTP stream within a multipoint conference. The media stream source endpoint sends the **openLogicalChannel** message to the MC. In cases where the source would like to indicate a destination for the **openLogicalChannel**, the source endpoint should place the **terminalLabel** of the destination endpoint in the **destination** field of the **h2250LogicalChannelParameters**. The source endpoint shall also place its own **terminalLabel** in the **source** field of **h2250LogicalChannelParameters**. Note that in the multicast model, the absence of a **destination** indicates that the stream is applicable to all endpoints.

If a source endpoint has been assigned a **TerminalLabel** by an MC, the source endpoint shall use an SSRC that contains the lowest byte of its **TerminalLabel** as the lowest byte of its SSRC.

The destination endpoint may associate the logical channel number with the RTP stream source by comparing the **OpenLogicalChannel.h2250LogicalChannelParameters.source** field with the lowest byte of the SSRC in the RTP header.

It is possible for SSRC collisions when an H.323 endpoint is in an H.332 conference. The endpoint detecting the collision shall follow the procedures in RTP for SSRC collision resolution.

### 8.3.5 Communication Mode Command Procedures

The H.245 **communicationModeCommand** is sent by an H.323 MC to specify the communication mode for each media type: unicast or multicast. This command may cause a switch between a centralized and decentralized conference and therefore may involve closing all existing logical channels and opening new ones.

The **communicationModeCommand** specifies all the sessions in the conference. For each session, the following data are specified: the RTP session identifier, the associated RTP session ID if applicable,



a terminal label if applicable, a description of the session, the **dataType** of the sessions (e.g. G.711), and a unicast or multicast address for the media and media control channels as appropriate for the conference configuration and type.

The **communicationModeCommand** conveys the transmit modes which conference endpoints are to use in a conference. The command does not convey receive modes, as they are specified by **openLogicalChannel** commands which are sent from the MC to the endpoints.

It is presumed that the **communicationModeCommand** is defining the modes of a conference and is therefore sent after the **multipointConference** indication which notifies an endpoint that it must comply with the commands of the MC. Endpoints should wait for a **communicationModeCommand** before opening logical channels when they have received a **multipointConference** indication.

Endpoints receiving a **communicationModeCommand** use the **terminalLabel** field of each table entry to determine if the entry is applicable for its own processing. Entries which do not contain a **terminalLabel** apply to all endpoints in the conference. Entries which contain **terminalLabels** are commands to specific endpoints which match the **terminalLabel** in the entry. For example, when audio streams from all endpoints are placed on one multicast address (one session), the table entry for the audio mode, media address, and media control address will not contain a **terminalLabel**. When the table entry commands an endpoint to send its video to a multicast address, the MC will include that endpoint's **terminalLabel**.

The **communicationModeCommand** can be used to instruct endpoints in a conference (or a point-to-point call) to change modes by indicating a new mode for a **mediaChannel** that is already in use. It can also be used to tell an endpoint to transmit the media stream to a new address by indicating the mode currently in use, but with new **mediaChannel**. Similarly, an endpoint that receives a **communicationModeCommand** indicating the mode currently in use and no **mediaChannel** should close the appropriate channel and the attempt to reopen using the **openLogicalChannel-openLogicalChannelAck** sequence, where the **openLogicalChannelAck** contains the address to which the endpoint will send the media.

Appendix I contains examples of the **communicationModeTable** entries for various cases.

## 8.4 Phase D – Call services

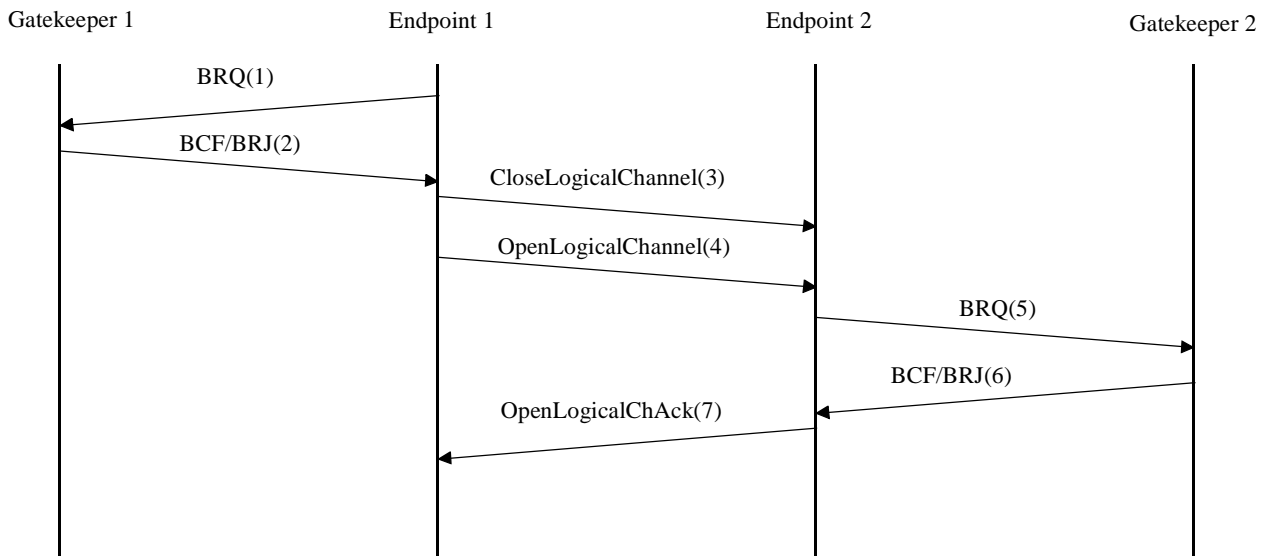
### 8.4.1 Bandwidth changes

Call bandwidth is initially established and approved by the Gatekeeper during the admissions exchange. An endpoint shall assure that the aggregate for all transmitted and received audio and video channels, excluding any RTP headers, RTP payload headers, network headers, and other overhead, is within this bandwidth. Data and control channels are not included in this limit.

At any time during a conference, the endpoints or Gatekeeper may request an increase or decrease in the call bandwidth. An endpoint may change the bit rate of a logical channel without requesting a bandwidth change from the Gatekeeper if the aggregate bit rate of all transmitted and received channels does not exceed the current call bandwidth. If the change will result in an aggregate bit rate that exceeds the current call bandwidth, the endpoint shall request a change in the call bandwidth from its Gatekeeper and await confirmation prior to actually increasing any bit rate. A bandwidth change request is recommended when an endpoint will use a reduced bandwidth for an extended period of time, thus freeing up bandwidth for other calls.

An endpoint wishing to change its call bandwidth sends a Bandwidth Change Request (BRQ) message (1) to the Gatekeeper. The Gatekeeper determines if the request is acceptable. The criteria for this determination is outside the scope of this Recommendation. If the Gatekeeper determines that the request is not acceptable, it returns a Bandwidth Change Reject (BRJ) message (2) to endpoint. If the Gatekeeper determines that the request is acceptable, it returns a Bandwidth Change Confirm (BCF) message (2).

If Endpoint 1 wishes to increase its transmitted bit rate on a logical channel, it first determines if the call bandwidth will be exceeded. See Figure 25. If it will, Endpoint 1 shall request a bandwidth change (1 and 2) from Gatekeeper 1. When the call bandwidth is sufficient to support the change, Endpoint 1 sends a **closeLogicalChannel** (3) message to close the logical channel. It then reopens the logical channel using the **openLogicalChannel** (4) specifying the new bit rate. If the receiving endpoint wishes to accept the channel with the new bit rate, it must first assure that its call bandwidth is not exceeded by the change. If it is, the endpoint shall request a call bandwidth change (5 and 6) with its Gatekeeper. When the call bandwidth is sufficient to support the channel, the endpoint replies with an **openLogicalChannelAck** (7); otherwise, it responds with an **openLogicalChannelReject** indicating unacceptable bit rate.

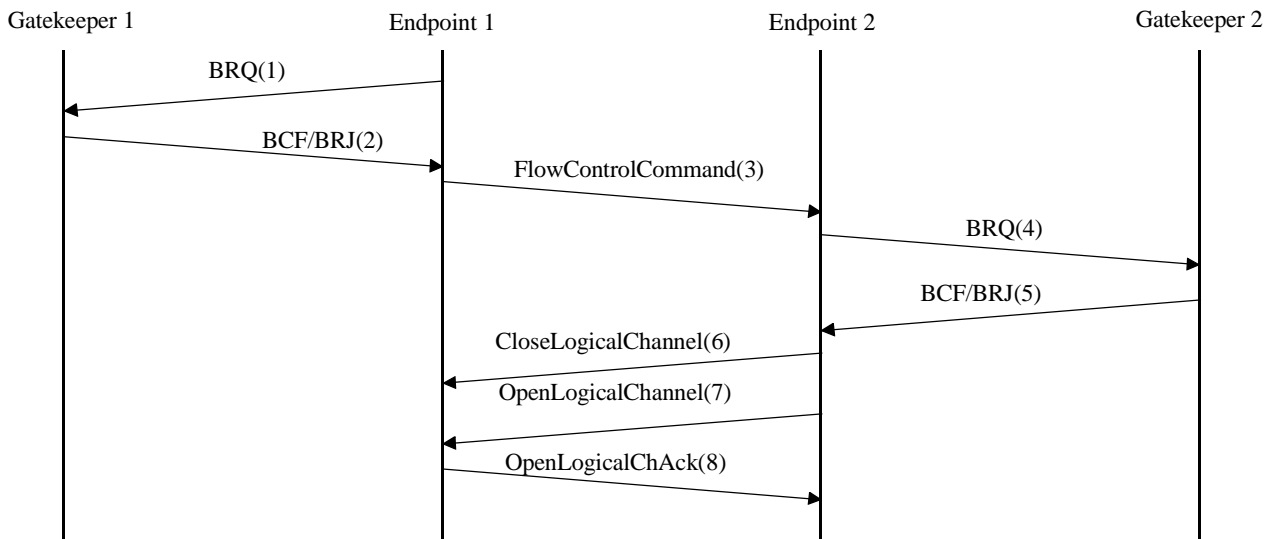


T1521430-96

NOTE – Gatekeeper 1 and Gatekeeper 2 may be the same Gatekeeper.

**Figure 25/H.323 – Bandwidth change request – Transmitter change**

If Endpoint 1 wishes to increase its transmitted bit rate on a logical channel from Endpoint 2, which it previously flow controlled to a lower bit rate, Endpoint 1 first determines if the call bandwidth will be exceeded. See Figure 26. If it will, Endpoint 1 shall request a bandwidth change from Gatekeeper 1. When the call bandwidth is sufficient to support the change, Endpoint 1 sends a **flowControlCommand** (3) to indicate the new upper limit on bit rate for the channel. If Endpoint 2 decides to increase the bit rate on the channel, it must first assure that its call bandwidth is not exceeded by the change. If it is, Endpoint 2 shall request a call bandwidth change (4 and 5) with its Gatekeeper. When the call bandwidth is sufficient to support the channel, Endpoint 2 will send the **closeLogicalChannel** (6) message to close the logical channel. It then reopens the logical channel using the **openLogicalChannel** (7) specifying the new bit rate. Endpoint 1 should then accept the channel with the new bit rate, and it replies with an **openLogicalChannelAck** (8).



T1521440-96

NOTE – Gatekeeper 1 and Gatekeeper 2 may be the same Gatekeeper.

**Figure 26/H.323 – Bandwidth change request – Receiver change**

A Gatekeeper wishing to change the transmitted bit rate of Endpoint 1 sends a BRQ message to Endpoint 1. If the request is for a decrease in bit rate, Endpoint 1 shall always comply by reducing its aggregate bit rate and returning a BCF. Endpoint 1 may initiate the appropriate H.245 signalling to inform Endpoint 2 that bit rates have changed. This will allow Endpoint 2 to inform its Gatekeeper of the change. If the request is for an increase, the endpoint may increase its bit rate when desired.

#### 8.4.2 Status

In order for the Gatekeeper to determine if an endpoint is turned off, or has otherwise entered a failure mode, the Gatekeeper may use the Information Request (IRQ)/Information Request Response (IRR) message sequence (see Recommendation H.225.0) to poll the endpoints at an interval decided by the manufacturer. The polling interval shall be greater than 10 s. This message may also be used by a diagnostic device as described in 11.2.

The Gatekeeper may want an endpoint to periodically send an unrequested IRR message. The Gatekeeper may indicate this to the endpoint by specifying the rate that this IRR is sent within the irrFrequency field of the Admission Confirm (ACF) message. An endpoint receiving this irrFrequency rate shall send an IRR message at that rate for the duration of the call. While this rate is in effect, the Gatekeeper may still send IRQ messages to the endpoint which shall respond as described above.

An endpoint may want some of the unrequested IRRs to be delivered reliably. The Gatekeeper can enable this by using the **willRespondToIRR** field in the RCF or ACF that it can acknowledge unrequested IRRs. In this case, the endpoint may explicitly request the Gatekeeper to send an acknowledgment for the IRR. The Gatekeeper shall respond to such an IRR message by sending either an acknowledgment (IACK) or a negative acknowledgment (INAK). If the Gatekeeper did not announce that it will acknowledge IRRs, or if the endpoint did not request such an acknowledgment, no response shall follow the IRR.

During the duration of a call, an endpoint or Gatekeeper may periodically request call status from another endpoint. The requesting endpoint or Gatekeeper issues a Status Enquiry message. The Endpoint receiving the Status Enquiry message shall respond with a Status message indicating the current call state. This procedure may be used by the Gatekeeper in order to periodically check if a call is still active. Endpoints shall be able to accept any valid state values received in the Status

message, including those which it may not be capable of entering. Note that this is a H.225.0 message sent on the Call Signalling Channel, and should not be confused with IRR which is a RAS message sent on the RAS Channel.

The Gatekeeper may want to receive copies of certain H.225.0 call signalling PDUs when they are received or sent by an endpoint. An endpoint indicates its capability to send these PDUs by setting the **willSupplyUIEs** in the ARQ or RRQ message sent to the Gatekeeper. The Gatekeeper indicates the list of PDU types it wishes to receive copies of, in the **uuiesRequested** field in the ACF or RCF. It also indicates if it wants copies when the PDUs are sent or received. An endpoint indicating this capability and receiving this list, shall send an IRR to the Gatekeeper each time it receives/sends the type of PDU requested.

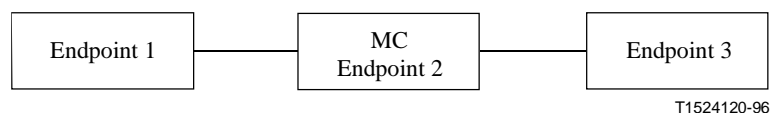
### 8.4.3 Ad hoc conference expansion

The following procedures are optional for terminals and Gateways and mandatory for MCs.

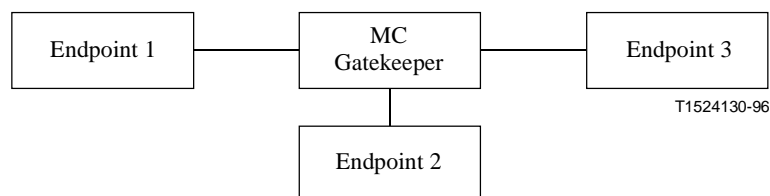
When a user places a call, the intent of the call is often not known to the calling endpoint. The user may wish to simply create a conference for itself and the called endpoint, the user may wish to join some conference at the called entity, or the user may wish to get a list of conferences that the called entity can provide. Using the procedures of this subclause the conferences can be expanded from point-to-point calls into Ad Hoc Multipoint conferences.

An Ad Hoc Multipoint conference is one that can be expanded from a point-to-point conference involving an MC to a multipoint conference. First, a point-to-point conference is created between two endpoints (Endpoint 1 and Endpoint 2). At least one endpoint, or the Gatekeeper, must contain an MC. Once the point-to-point conference has been created, the conference may be expanded to multipoint conference in two different ways. The first way is when any endpoint in the conference invites another endpoint (Endpoint 3) into the conference by calling that endpoint through the MC. The second way is for an endpoint (Endpoint 3) to join an existing conference by calling an endpoint in the conference.

Ad Hoc Conference expansion can take place when using either the Direct Call Signalling model or the Gatekeeper Routed Call Signalling model. The H.245 Control Channel topology for the Direct Call Signalling model appears as:



The H.245 Control Channel topology for the Gatekeeper Routed Call Signalling model appears as:



In either case an MC must be present in the conference at the time of expansion to any number greater than 2 endpoints. Note that in the Gatekeeper routed model, the MC may be located in the Gatekeeper and/or one of the endpoints.

The procedures required to create a point-to-point conference and then expand the conference through invite and join, for each call model, is covered in the following subclauses. Procedures for the calling endpoint to discover a list of conferences that the called entity can provide are also covered.

It should be noted that the call is ended by a failure of the entity that is providing the MC.

#### 8.4.3.1 Direct Endpoint Call Signalling – Conference Create

Endpoint 1 creates a conference with Endpoint 2 as follows:

- A1) Endpoint 1 sends a Setup message to Endpoint 2 containing a globally unique CID = N and **conferenceGoal = create** according to the procedure in 8.1.
- A2) Endpoint 2 has the following options:
  - A2a) If it wants to join the conference, it sends a Connect message with CID = N to Endpoint 1. In this case it is either:
    - 1) not participating in another conference; or
    - 2) it is participating in another conference, it is capable of participating in multiple conferences at the same time, and the received CID = N does not match the CID of any of the conferences in which it is currently participating.
  - A2b) If it is in another conference with CID = M and can participate in only one conference at a time it either:
    - 1) rejects the call by sending Release Complete indicating in-conference; or
    - 2) it can request Endpoint 1 to join the conference with CID = M by sending a Facility message indicating **routeCallToMC** with the Call Signalling Channel Transport Address of the endpoint containing the MC and CID = M of the conference. The handling of the Facility message by Endpoint 1 is described in 8.4.3.7.
  - A2c) If it does not wish to join this conference, it rejects the call by sending Release Complete indicating that the destination is busy.
  - A2d) If Endpoint 2 is an MC(U) that hosts multiple conferences and wishes to provide Endpoint 1 with a choice of conferences to join, it can send a Facility message indicating **conferenceListChoice** and a list of conferences that Endpoint 1 may choose from. The list of conferences is sent as part of the Facility-UUIE. For backward compatibility, with version 1 endpoints, conference lists are only provided if the **protocolIdentifier** in Endpoint 1's Setup message indicates that it is version 2 or above.

Upon receipt of this **conferenceListChoice** Facility message, Endpoint 1 may join a conference from the list of conferences by sending a new Setup message to the MC(U) on the Call Signalling Channel which contains the selected CID and which has **conferenceGoal = join**. If Endpoint 1 chooses not to join any of the listed conferences, it shall send a Release Complete message to the MC(U).
- A3) If Endpoint 2 enters the conference, Endpoint 1 uses the Transport Address of the Control Channel provided in the Connect message to open the Control Channel with Endpoint 2.
- A4) The H.245 messages are then exchanged as described below:
  - A4a) **terminalCapabilitySet** messages are exchanged between the endpoints to determine the version number of the H.245 used in order to parse the remaining received messages correctly.
  - A4b) Using H.245 master-slave determination procedure, it is determined that Endpoint 2 is the master. In the Gatekeeper-Routed model, the master could be in an MC collocated with the Gatekeeper. If the master has an MC, it becomes the Active MC. It may then send the **mcLocationIndication** to the other endpoint(s). The MC may be active in the conference now, or when the user initiates the multipoint conference function, at the choice of the manufacturer.

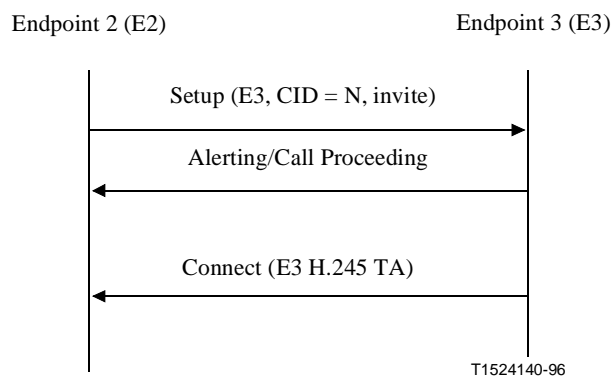
- A4c) The master may send the **terminalNumberAssign** message to the endpoints. The endpoints shall use the 8-bit terminal number, and not use the 8-bit MCU number, from the 16-bit number assigned as the low 8 bits of the SSRC field in the RTP header. These low 8 bits in SSRC then identify the streams from a particular endpoint.
- A4d) Since the capabilities of the receiver are known from the **terminalCapabilitySet** message, the transmitter opens the logical channels. It shall send one **h2250MaximumSkewIndication** for each pair of audio and video transmitted.

#### 8.4.3.2 Direct Endpoint Call Signalling – Conference Invite

There are two cases of the conference invite. First, the endpoint which contains the Active MC wishes to invite another endpoint into the conference. Second, an endpoint which does not contain the Active MC wishes to invite another endpoint into the conference.

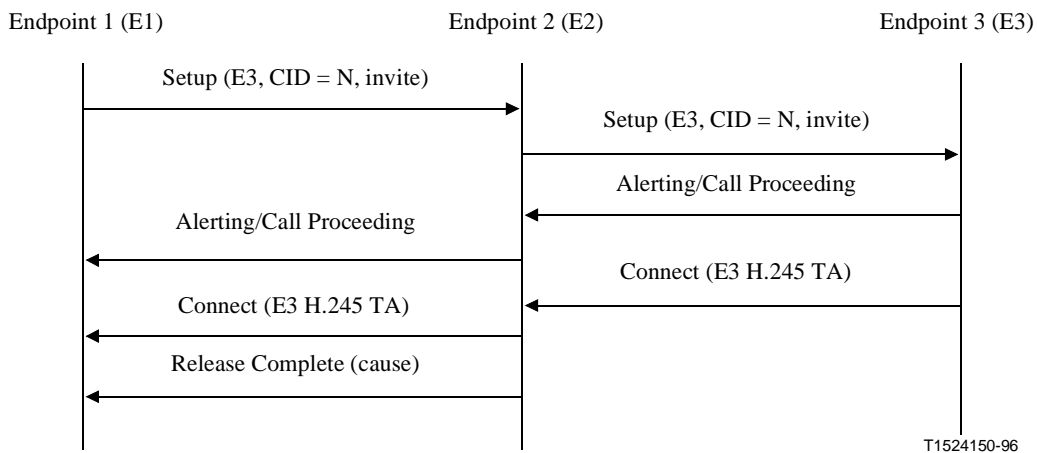
- 1) After a point-to-point conference has been established using procedures A1) to A4), an endpoint (Endpoint 2) containing the Active MC wishing to add another endpoint to the conference shall use the following procedure:
  - B1) Endpoint 2 sends a Setup message to Endpoint 3 with CID = N and **conferenceGoal = invite** according to the procedures in 8.1. See Figure 27.
  - B2) Endpoint 3 has the following options:
    - B2a) If it wishes to accept the invitation to join the conference, it sends a Connect message with CID = N to Endpoint 2.
    - B2b) If it wishes to reject the invitation to join the conference, it sends a Release Complete message to Endpoint 2 indicating that the destination is busy.
    - B2c) If it is in another conference with CID = M, it can request Endpoint 2 to join the conference with CID = M by sending a Facility message indicating **routeCallToMC** with the Call Signalling Channel Transport Address of the endpoint containing the MC and CID = M of the conference. The handling of the Facility message by Endpoint 2 is described in 8.4.3.7.
    - B2d) If the received CID matches the CID of a conference that Endpoint 3 is currently participating in, it shall reject the call by sending Release Complete indicating that it is already in the conference.
  - B3) If Endpoint 3 accepts the invitation, Endpoint 2 uses the Transport Address of the Control Channel provided in the Connect message to open the Control Channel with Endpoint 3.
  - B4) The H.245 messages are then exchanged as described below:
    - C1) **terminalCapabilitySet** messages are exchanged between the MC and Endpoint 3.
    - C2) Using H.245 master-slave determination procedure, it is determined that Endpoint 2 is already the Active MC. The MC may then send the **mcLocationIndication** to the Endpoint 3.
    - C3) The MC shall send **multipointConference** at this time to all the three endpoints.
    - C4) The MC may send the **terminalNumberAssign** message to Endpoint 3. If received, the endpoints shall use the 8-bit terminal number, and not use the 8-bit MCU number, from the 16-bit number assigned as the low 8 bits of the SSRC field in the RTP header. These low 8 bits in SSRC then identify the streams from a particular endpoint.
    - C5) An endpoint can get the list of the other endpoints in the conference by sending the **terminalListRequest** message to the MC. The MC responds with the **terminalListResponse**.

- C6) Whenever a new endpoint joins the conference, the MC sends the **terminalNumberAssign** message to Endpoint 4 and **terminalJoinedConference** message to Endpoints 1, 2 and 3.
- C7) Whenever an endpoint leaves the conference, the MC sends **terminalLeftConference** to the remaining endpoints.
- C8) The MC shall send the **communicationModeCommand** to all the endpoints in the conference.
- C9) Endpoint 1 and Endpoint 2 will close their logical channels that were created during the point-to-point conference if they are inconsistent with the information contained in the **communicationModeCommand**.
- C10) The logical channels can now be opened between the MC and the endpoints.



**Figure 27/H.323 – MC invite signalling**

- 2) After a point-to-point conference has been established using procedures A1) to A4) in 8.4.3.1, an endpoint (Endpoint 1) that does not contain the Active MC wishing to add another endpoint to the conference, shall use the following procedure:
  - B1) Endpoint 1 sends a Setup message to the MC (Endpoint 2) with a new CRV indicating a call to Endpoint 3 by providing the Transport Address of Endpoint 3, CID = N, and **conferenceGoal = invite**. See Figure 28.
  - B2) Endpoint 2 sends a Setup message to Endpoint 3 with CID = N and **conferenceGoal = invite** according to the procedures in 8.1.
  - B3) During call signalling with Endpoint 3, Endpoint 2 shall pass Call Signalling messages received from Endpoint 3, including Connect, to Endpoint 1 (the original inviter).
  - B4) Endpoint 3 has the same options, described previously, of either accepting or rejecting the invitation.
  - B5) At some time after the completion of the call setup procedure between Endpoint 2 and Endpoint 3, Endpoint 2 shall send a Release Complete message to Endpoint 1.
  - B6) If Endpoint 3 accepts the invitation, Endpoint 2 uses the Transport Address of the Control Channel provided in the Connect message to open the Control Channel with Endpoint 3.
  - B7) The H.245 messages are then exchanged as previously described in procedures C1) to C10).



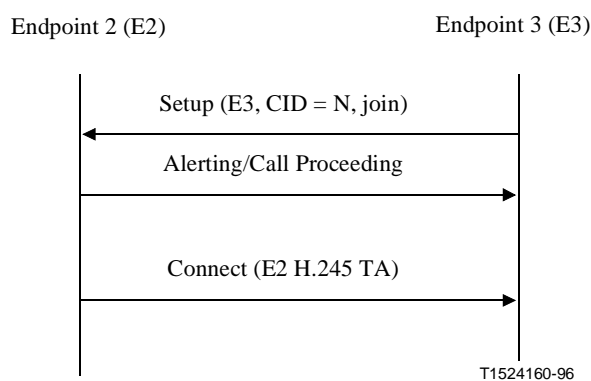
**Figure 28/H.323 – Non-MC invite signalling**

### 8.4.3.3 Direct Endpoint Call Signalling – Conference Join

There are two cases of the conference join. First, an endpoint calls the endpoint which contains the Active MC. Second, an endpoint calls an endpoint which is not the Active MC.

After a point-to-point conference has been established using procedures A1) to A4), an endpoint (Endpoint 3) wishing to join a conference may attempt to connect with the endpoint containing the Active MC in the conference. In this case, the following procedure shall be used:

- B1) Endpoint 3 sends a Setup message to Endpoint 2 with CID = N and **conferenceGoal = join** according to the procedures in 8.1. See Figure 29.
- B2) If the CID matches the CID of an active conference in the MC, Endpoint 2 (MC) has the following options:
  - B2a) If it decides that Endpoint 3 should be allowed to join the conference, it sends the Connect message with CID = N.
  - B2b) If it decides that Endpoint 3 should not be allowed to join the conference, it sends the Release Complete message indicating that the destination is busy.
- B3) If the CID does not match the CID of an active conference in the MC, Endpoint 2 shall send Release Complete indicating a bad CID.
- B4) If Endpoint 2 allows the join, Endpoint 2 opens the Control Channel with Endpoint 3.
- B5) The H.245 messages are then exchanged as previously described in procedures C1) to C10).

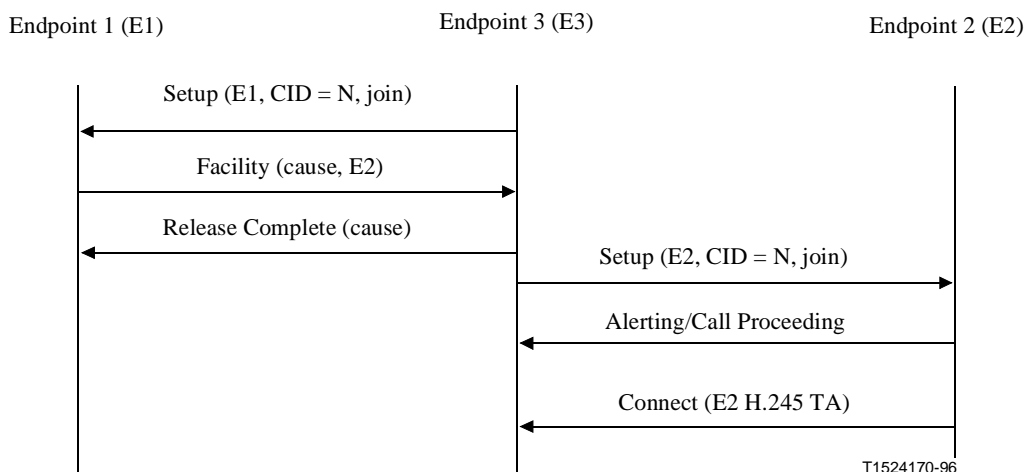


**Figure 29/H.323 – MC join signalling**



After a point-to-point conference has been established using procedures A1) to A4), an endpoint (Endpoint 3) wishing to join a conference may attempt to connect with an endpoint that does not contain the Active MC in the conference. In this case, the following procedure shall be used:

- B1) Endpoint 3 sends a Setup message to Endpoint 1 with CID = N and **conferenceGoal = join** according to the procedures in 8.1. See Figure 30.
- B2) Endpoint 1 returns a Facility message indicating **routeCallToMC** with the Call Signalling Channel Transport Address of Endpoint 2 (containing the Active MC) and the CID = N of the conference.
- B3) Endpoint 3 then sends a Setup message to Endpoint 2 (MC) with CID = N and **conferenceGoal = join** as described in the previous conference join procedure.



**Figure 30/H.323 – Non-MC join signalling**

#### 8.4.3.4 Gatekeeper Routed Call Signalling – Conference Create

In cases where the Gatekeeper routes the Call Signalling Channel and the H.245 Control Channel, the Gatekeeper may contain (or have access to) an MC or MCU. Procedures A1) to A4) are used to establish the point-to-point call.

If the MC(U) hosts multiple conferences and wishes to provide Endpoint 1 with a choice of conferences to join, it can send a Facility message indicating **conferenceListChoice** and a list of conferences that Endpoint 1 may choose from. The list of conferences is sent as part of the Facility-UUIE. For backward compatibility, with version 1 endpoints, conference lists are only provided if the **protocolIdentifier** in Endpoint 1's Setup message indicates that it is version 2 or above.

Upon receipt of this **conferenceListChoice** Facility message, Endpoint 1 may join a conference from the list of conferences by sending a new Setup message to the MC(U) on the Call Signalling Channel which contains the selected CID and which has **conferenceGoal = join**. If Endpoint 1 chooses not to join any of the listed conferences, it shall send a Release Complete message to the MC(U).

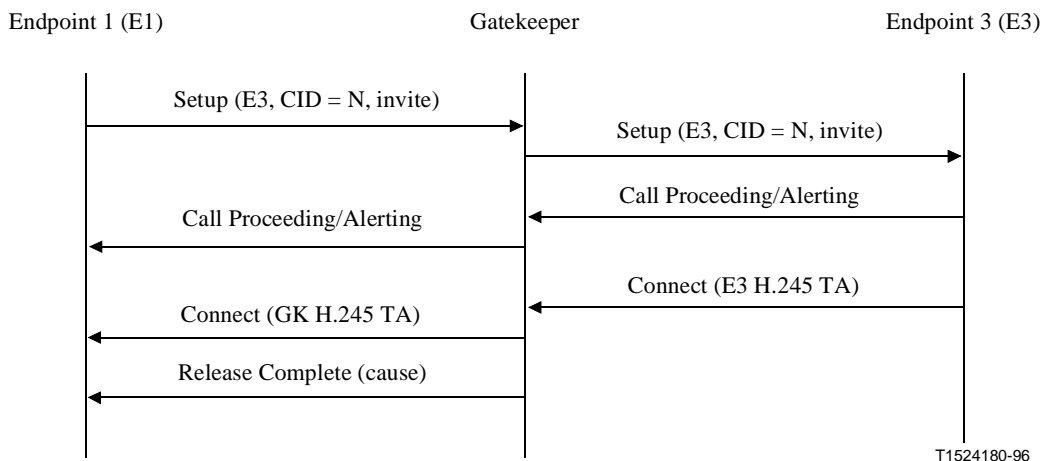
During master-slave determination [A4b], if the Gatekeeper's **terminalType** is greater than the **terminalType** received in the **masterSlaveDetermination** message, the Gatekeeper may attempt to become master for the call. In this case, the Gatekeeper shall immediately send a **masterSlaveDeterminationAck** message to the source of the Master-Slave Determination message indicating that it is a slave, and the Gatekeeper performs Master-Slave Determination with the destination entity as defined in section 6.2.8.4. If the Gatekeeper wins that Master-Slave Determination, the MC associated with the Gatekeeper shall be the Active MC. If the Gatekeepers

**terminalType** is not greater than the **terminalType** of the endpoint or the Gatekeeper decides not to replace the endpoint's **terminalType** with its own, the Gatekeeper shall not modify the **terminalType** value and it shall transparently relay all messages of that Master-Slave Determination procedure.

#### 8.4.3.5 Gatekeeper Routed Call Signalling – Conference Invite

After a point-to-point conference has been established using procedures A1) to A4) as modified above, an endpoint (Endpoint 1 or 2) that does not contain the Active MC wishing to add another endpoint to the conference shall use the following procedure:

- B1) Endpoint 1 sends a Setup message through the Gatekeeper directed to Endpoint 3 with a new CRV, CID = N and **conferenceGoal = invite**. See Figure 31.
- B2) The Gatekeeper (MC) sends a Setup message to Endpoint 3 with CID = N and **conferenceGoal = invite** according to the procedures in 8.1.
- B3) During call signalling with Endpoint 3, the Gatekeeper shall pass Call Signalling messages received from Endpoint 3, including Connect, to Endpoint 1 (the original inviter).
- B4) Endpoint 3 has the same options, described previously, of either accepting or rejecting the invitation.
- B5) At some time after the completion of the call setup procedure between the Gatekeeper and Endpoint 3, the Gatekeeper shall send a Release Complete message to Endpoint 1.
- B6) If Endpoint 3 accepts the invitation, the Gatekeeper uses the Transport Address of the Control Channel provided in the Connect message to open the Control Channel with Endpoint 3.
- B7) The H.245 messages are then exchanged as previously described in procedures C1) to C10) with the Gatekeeper taking part in all master-slave determination procedures as the Active MC (C2). At this time, the Control Channels from the endpoints should be connected to the MC, and the MC should be in control of the conference.



**Figure 31/H.323 – Gatekeeper routed invite signalling**

### 8.4.3.6 Gatekeeper Routed Call Model – Conference Join

After a point-to-point conference has been established using procedures A1) to A4) as modified above, an endpoint (Endpoint 3), wishing to join a conference may attempt to connect with an endpoint that does not contain the Active MC in the conference. In this case, the following procedure shall be used:

- B1) Endpoint 3 sends a Setup message through the Gatekeeper directed to Endpoint 1 with CID = N and **conferenceGoal = join** according to the procedures in 8.1. See Figure 32.
- B2) If the CID matches the CID of an active conference in the MC, the Gatekeeper (MC) has the following options:
  - B2a) If it decides that Endpoint 3 should be allowed to join the conference, it sends the Connect message with CID = N to Endpoint 3.
  - B2b) If it decides that Endpoint 3 should not be allowed to join the conference, it sends the Release Complete message indicating that the destination is busy.
  - B2c) The Gatekeeper may forward the Setup message to Endpoint 1. Endpoint 1 may respond with a Facility message indicating **routeCallToMC** or it may respond with a Release Complete.
- B3) If the CID does not match the CID of an active conference in the MC, the Gatekeeper shall send Release Complete indicating a bad CID.
- B4) If the Gatekeeper allows the join, the Gatekeeper uses the Transport Address of the Control Channel provided in the Setup message to open the Control Channel with Endpoint 3.
- B5) The H.245 messages are then exchanged as previously described in procedures C1) to C10) with the Gatekeeper taking part in all master-slave determination procedures as the Active MC (C2). At this time, the Control Channels from the endpoints should be connected to the MC, and the MC should be in control of the conference.

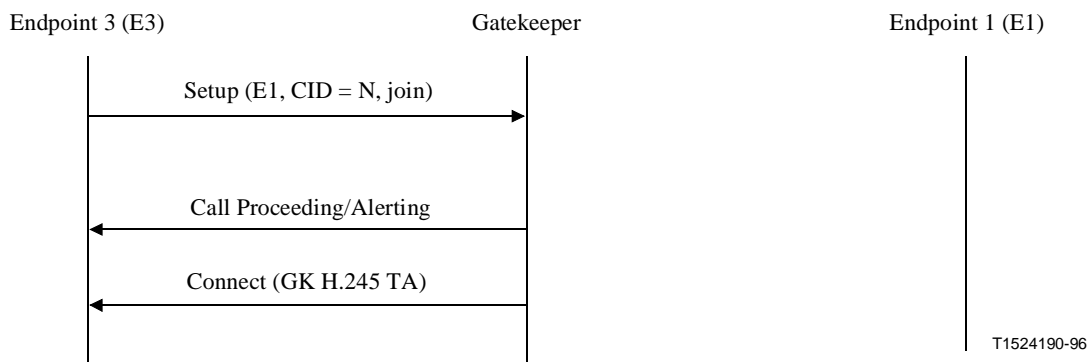


Figure 32/H.323 – Gatekeeper routed join signalling

### 8.4.3.7 Handling of the Facility Message

Upon receiving a Facility message indicating **routeCallToMC** with the Call Signalling Channel Transport Address of the endpoint containing the MC and CID of a conference, an endpoint may release the current call and attempt to join the indicated conference according to the procedures in 8.4.3.3 or in 8.4.3.6.

An endpoint may receive such a Facility message either as a direct reply to its Setup message or during the active phase of a call.

#### 8.4.3.8 Conference out of Consultation

This subclause defines the procedures for an endpoint (endpoint A) requesting an ad hoc conference with two or more other endpoints (remote endpoints B, C, etc.) with which endpoint A already has active calls. This typically applies – but is not limited to – ad-hoc conference being requested out of a consultation condition.

NOTE 1 – "Consultation condition" refers to a situation where endpoint A has an active call with endpoint C (consultation call) while having one or more endpoints on hold, i.e. held call(s). An endpoint may be put on hold by using the procedures of H.450.4 [36], H.323 subclause 8.4.6, or by local procedures.

Endpoint A has the capability of "merging" the independent calls to multiple endpoints into a single conference either at the endpoint A (as described in scenario 1 below) or by forming the conference on a separate MCU (as described in scenario 2 below).

NOTE 2 – Procedures in this subclause relate only to the calls at an endpoint that are to be joined into a conference out of consultation. An endpoint may have additional calls that do not participate in the conference and to which this subclause will not apply.

##### 8.4.3.8.1 Scenario 1: Conference provided by endpoint

If endpoint A has the capabilities, it may "merge" the held call and the consulted call into a conference resulting in a three-way conversation between A, B and C. For this scenario endpoint A must have an MC. Both the centralized and the de-centralized conferencing models are possible. If the centralized model is to be used (i.e. if the terminal provides the media mixing/switching), endpoint A shall have an MP.

An endpoint with MC and MP is actually an MCU and should use **terminalType** 170, 180 or 190 as appropriate for master-slave determination.

The following scenarios are possible:

- 1a) If endpoint A is the master of both calls to B and C, it may simply retrieve the held call into the conference with C and declare itself as the Active MC on both calls through master-slave negotiation.
- 1b) If endpoint A is a slave on one or more of the calls but no call on which it is the slave has an Active MC, endpoint A should reinitiate master-slave determination on all calls in which it is slave, using the **terminalType** 240, as specified in Table 1 for an Active MC. If it ends this procedure as master on all calls, it should act as in 1a) above; if it is slave in one or more calls, endpoint A should act as directed in 1c) below.
- 1c) If one or more of the calls in which endpoint A is participating is already a call in which endpoint A is not the Active MC, procedures for cascading MCUs shall be followed.

Once a conference is established within endpoint A, a further endpoint D – that is being consulted by endpoint A – may be invited into the existing conference as described in 8.4.3.2 and 8.4.3.5.

##### 8.4.3.8.2 Scenario 2: Conference provided by MCU

If endpoint A has access to an MCU, the following procedure may be used to accomplish conference out of consultation:

- 2a) Endpoint A establishes a new call to the MCU using a Setup message with **conferenceGoal** = **create** and **CID** = N.
- 2b) Endpoint A drops its call with endpoint C using a Release Complete message with **reason** set to **replaceWithConferenceInvite** including argument **CID** = N.
- 2c) Endpoint A sends a Setup message to the MCU with **conferenceGoal** = **invite**, **CID** = N, and sufficient information for the MCU to make a call to endpoint C (see also 8.4.3.2).

- 2d) Steps 2b) and 2c) shall be repeated with "endpoint C" replaced by "endpoint B". Note that there is no requirement to retrieve the call to B from hold before inviting it to the conference.
- 2e) For exchange of H.245 conference related messages refer to 8.4.3.2 steps C1-C10.

Alternative mechanisms to steps 2b), 2c) and 2d) are:

- 1) H.450.2 [35] Call Transfer (with endpoint A acting as "transferring" endpoint, endpoints B and C acting as "transferred" endpoints and the MC/MCU acting as the "transferred-to" endpoint. The Facility message containing **callTransferInitiate Invoke APDU** shall also contain element **CID = N**.
- 2) H.225.0 "Facility re-route to MC" mechanism (sending an H.225.0 Facility message to endpoints B and C containing **CID = N**, **facilityReason = routeCallToMC** and the address of the MCU) if H.450.2 is not supported.

These alternative mechanisms are recommended if the remote endpoint is located within the SCN.

An endpoint (e.g. endpoint A) may split from the conference (e.g. by putting its call to the MCU on hold). Endpoint A may then consult with a further endpoint D that may subsequently be invited to the existing conference by using the procedures as described in 2b) and 2c) above with "endpoint C" replaced by "endpoint D". Alternative mechanisms as described above by means of using H.450.2 Call Transfer or H.225.0 "Facility re-route to MC" may be used instead.

#### 8.4.4 Supplementary services

Support for Supplementary Services is optional. The H.450-series of Recommendations describe a method of providing Supplementary Services in the H.323 environment.

#### 8.4.5 Multipoint cascading

In order to cascade MCs, a call must be established between the entities containing the MCs. This call is established according to the procedures defined in 8.1 and 8.4.3. Once the call is established, and the H.245 Control Channel is opened, the Active MC (determines according to the master/slave procedures in 6.2.8.4) may activate the MC in a connected entity. This is done by using the H.245 **remoteMC** message. The following results shall occur in response to the **remoteMC** message:

Calling entity	Called entity	Conference goal	RemoteMC Sender	RemoteMC Selection	Result
Active MC	Inactive MC	<b>create</b>	Calling entity	<b>masterActivate</b>	Called MC accepts request and becomes the master MC
Active MC	Inactive MC	<b>invite</b>	Calling entity	<b>slaveActivate</b>	Called MC accepts request and becomes a slave MC
Active MC	Inactive MC	<b>join</b>	N/A	N/A	Not allowed
Inactive MC	Active MC	<b>create</b>	N/A	N/A	Not allowed
Inactive MC	Active MC	<b>invite</b>	N/A	N/A	Not allowed
Inactive MC	Active MC	<b>join</b>	Called entity	<b>slaveActivate</b>	Calling MC accepts request and becomes a slave MC

Once the cascaded conference is established, either the master or slave MCs may invite other endpoints into the conference. There shall only be one master MC in a conference. A slave MC shall only be cascaded to a master MC. Slave MCs shall not be cascaded to other slave MCs. This allows only dumb-bell or star cascaded configurations.

The slave MC shall identify the cascaded conference using the CID established by the master when the conference was created.

The slave MC shall accept and act upon **communicationsModeCommand** messages from the master MC. The slave MC shall forward these messages to its locally connected endpoints. The slave MC may receive **requestMode** messages from its locally connected endpoints. It should forward these to the master MC. The slave MC shall not send **communicationsModeCommand** messages to the master MC.

The master MC should follow the procedures in 8.4.3.2, C3) through C10), in order to establish a common operating mode with the slave MC. Based on this information, each MC is responsible for opening logical channels for media distribution between its locally connected endpoints and endpoints designated by the master MC.

In addition to inviting new endpoints into the conference, an MC which supports multiple conferences may directly move endpoints into another conference without tearing down the existing connection. If this is done, the MC should send the **substituteCID** message to these endpoints. Endpoints which receive a **substituteCID** message during a call shall continue to use the conference ID (CID) used in the previous RAS messages (e.g. ARQ, BRQ, etc.) when conversing with its Gatekeeper for the duration of that particular call.

Terminal numbering and chair control functions may follow the procedures defined in Recommendation H.243. The use of T.120 for controlling MC cascading is for further study. The use of T.120 in cascaded connections is described in the T.120-series of Recommendations.

When a master sends a **remoteMC** Request with the selection **deActivate**, the slave MC should remove all endpoints from the conference.

#### **8.4.6 Third party initiated pause and re-routing**

For the purpose of this subclause, an empty capability set is defined as a **terminalCapabilitySet** message that contains only a sequence number and a protocol identifier.

To allow Gatekeepers to re-route connections from endpoints that do not support supplementary services, endpoints shall respond to the reception of an empty capability set as defined in this subclause. This feature allows "network" elements such as PBXs, call centers, and IVR systems to re-route connections independently of supplementary services and facilitates pre-connect announcements. It can also be used to delay H.245 media establishment when features such as Gatekeeper based user location are being used. It is also highly recommended that Version 1 endpoints support this feature.

On reception of an empty capability set, an endpoint shall enter a "transmitter side paused" state. On entering this state, the endpoint shall stop transmitting on established logical channels, and shall close all logical channels that it previously opened, including bidirectional logical channels. It shall close these channels in the usual way, that is, by sending the **closeLogicalChannel** message. The endpoint shall not request the remote endpoint to close logical channels, either uni-directional or bidirectional, that the remote endpoint opened. The endpoint shall send the **terminalCapabilitySetAck** message in the usual way: the message may be sent before stopping transmission, and so shall not be interpreted as an indication that transmission has stopped.

While in the "transmitter side paused" state, an endpoint shall not initiate the opening of any logical channels, but shall accept the opening and closing of logical channels from the remote end based on the usual rules and shall continue to receive media on open logical channels opened by the remote endpoint. This allows endpoints to receive announcements (e.g. pre-connect call progress) where the announcing entity does not wish to receive media from the endpoint. A **terminalCapabilitySet** message may be sent whenever an endpoint's capabilities change, including when the endpoint is in the "transmitter side paused" state. This allows communication to be established between two endpoints that initially do not declare any capabilities.

An endpoint shall leave the "transmitter side paused" state on reception of any **terminalCapabilitySet** message, other than an empty capability set. On leaving this state, an endpoint shall reset its H.245

state to that which it was in just after the H.245 transport connection was made at call establishment time (i.e. the beginning of phase B), but shall preserve state information relating to any logical channels that are open. This puts the endpoint in a known H.245 state after the pause. This allows an endpoint to be connected to a different endpoint when it is released from the paused state.

After leaving the "transmitter side paused" state, an endpoint shall proceed with normal H.245 procedures: it shall take part in master/slave determination signalling, and may proceed with normal open logical channel signalling procedures. When an MC leaves the "transmitter side paused" state, it shall act as if a new endpoint has entered the conference.

Unless its capabilities have changed, an endpoint need not resend a capability set as the gatekeeper will have supplied this to the remote endpoint to remove any paused state in the remote endpoint. This option of not sending a capability set enables faster reconnection. If the first **terminalCapabilitySet** message sent by an endpoint after leaving the "transmitter side paused" state differs from the capability set that the gatekeeper provided to the remote endpoint, the gatekeeper shall signal the remote endpoint to remove capabilities which were not indicated by the initiating endpoint.

NOTE 1 – An endpoint should take care with the capabilities it sends at this time. In particular, an endpoint shall send all capabilities it wants to advertise and not a small addition to previously signalled capabilities. In addition, if the endpoint has so many capabilities that it requires more than one **terminalCapabilitySet** to signal them, there may be a window of time when the gatekeeper has removed the capabilities described in second and subsequent **terminalCapabilitySet** messages.

NOTE 2 – A non-empty capability set shall not be sent to an endpoint until all of its transmit and receive logical channels have been closed. A switching entity should also send an H.450 redirection indication Facility message if the endpoint is being re-routed.

## 8.5 Phase E – Call termination

Either endpoint may terminate a call by the following procedure:

- 1) It should discontinue transmission of video at the end of a complete picture, and then close all logical channels for video.
- 2) It should discontinue transmission of data and then close all logical channels for data.
- 3) It should discontinue transmission of audio and then close all logical channels for audio.
- 4) It shall transmit the H.245 **endSessionCommand** message in the H.245 Control Channel, indicating to the far end that it wishes to disconnect the call and then discontinue H.245 message transmission.
- 5) It shall then wait to receive the **endSessionCommand** message from the other endpoint and then shall close the H.245 Control Channel.
- 6) If the Call Signalling Channel is open, a Release Complete message shall be sent and the channel closed.
- 7) It shall clear the call by using the procedures defined below.

An endpoint receiving **endSessionCommand** without first having transmitted it shall carry out steps 1) to 7) above, except that in step 5), it shall not wait for the **endSessionCommand** from the first endpoint.

Terminating a call may not terminate a conference; a conference may be explicitly terminated using an H.245 message (**dropConference**). In this case, the endpoints shall wait for the MC to terminate the calls as described above.

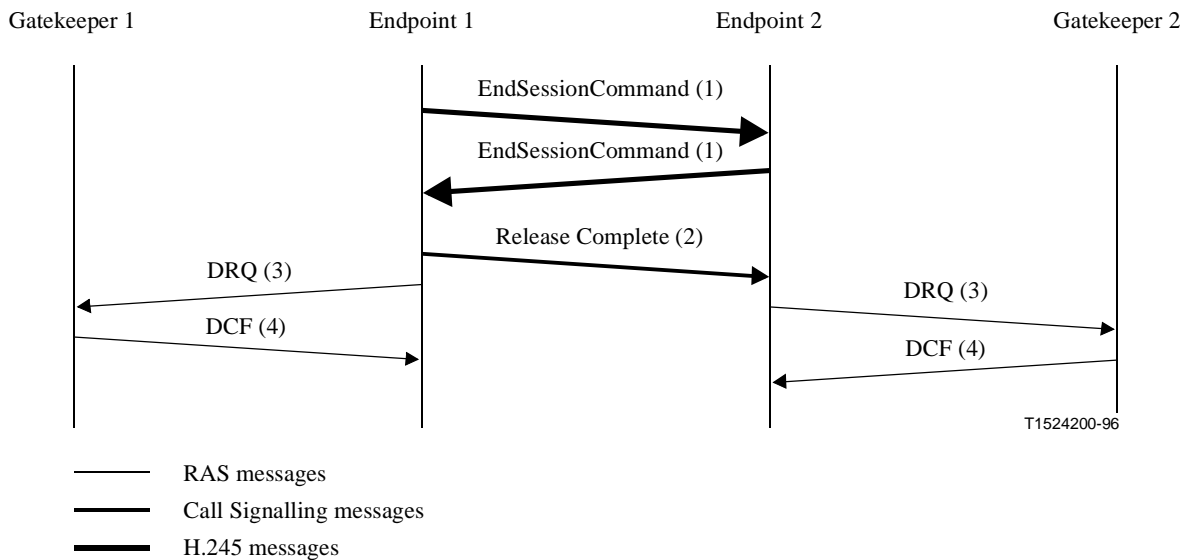
### 8.5.1 Call clearing without a Gatekeeper

In networks that do not contain a Gatekeeper, after steps 1) to 6) above, the call is terminated. No further action is required.

### 8.5.2 Call clearing with a Gatekeeper

In networks that contain a Gatekeeper, the Gatekeeper needs to know about the release of bandwidth. After performing steps 1) to 6) above, each endpoint shall transmit an H.225.0 Disengage Request (DRQ) message (3) to its Gatekeeper. The Gatekeeper shall respond with a Disengage Confirm (DCF) message (4). After sending the DRQ message, the endpoints shall not send further unsolicited IRR messages to the Gatekeeper. See Figure 33. At this point, the call is terminated. Figure 33 shows the direct call model; a similar procedure is followed for the Gatekeeper routed model.

The DRQ and DCF messages shall be sent on the RAS Channel.



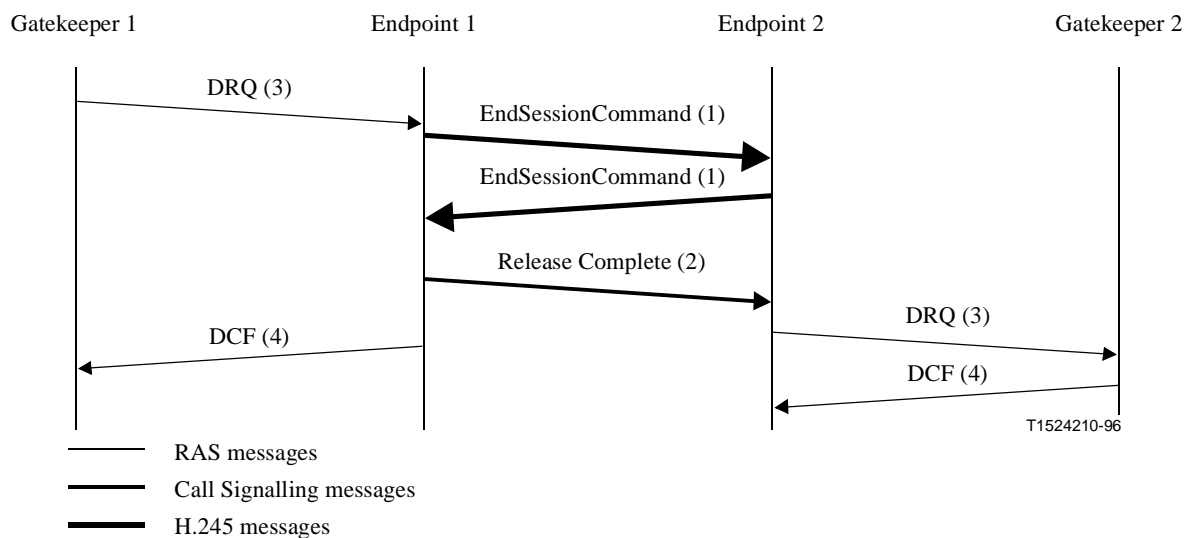
**Figure 33/H.323 – Endpoint initiated call clearing**

### 8.5.3 Call clearing by Gatekeeper

The Gatekeeper may terminate a call by sending a DRQ to an endpoint. See Figure 34. The endpoint shall immediately follow steps 1) through 6) from above and then reply to the Gatekeeper with DCF. The other endpoint, upon receiving **endSessionCommand**, shall follow the procedure described above. Figure 34 shows the direct call model; a similar procedure is followed for the Gatekeeper routed model.

If the conference is a multipoint conference, the Gatekeeper should send a DRQ to each endpoint in the conference, in order to close the entire conference.





**Figure 34/H.323 – Gatekeeper initiated call clearing**

## 8.6 Protocol failure handling

The underlying reliable protocol of the H.245 Control Channel uses appropriate effort to deliver or receive data on the channel before reporting a protocol failure. Therefore, if a protocol failure is reported on the channel, the H.245 Control Channel, and all associated logical channels shall be closed. This shall be done following the procedures of Phase E, as if the other endpoint had issued the H.245 **endSessionCommand**. This includes transmission of the DRQ message to the Gatekeeper and termination of the Call Signalling Channel. In the case where the failure is detected by the MC in a multipoint conference, the MC shall send **terminalLeftConference** messages to the remaining terminals. It is up to the implementation whether or not to try to re-establish the call without user intervention. In any case, this would appear to the other endpoint (and the Gatekeeper) as a new call.

The Call Signalling Channel also uses an underlying reliable protocol. Depending on the routing of the Call Signalling Channel, either the Gatekeeper or an endpoint may detect the protocol failure. If the Gatekeeper detects the failure, it shall attempt to re-establish the Call Control Channel. This implies that the endpoint shall always have the ability to establish a channel on its Call Signalling Channel Transport Address. Failure of the Call Signalling channel shall not change the Q.931 call state. After re-establishment of the Call Signalling Channel, the Gatekeeper may send a Status message to request the call state of the endpoint to assure that they are in synchronization.

If the endpoint detects the failure, the endpoint may choose to terminate the call as described in Phase E, or it may attempt to re-establish the Call Signalling Channel as described above.

If, during a call, an endpoint wants to determine if the other endpoint is still functioning and connected, it may send the H.245 **roundTripDelayRequest**. Since H.245 Control Channel is carried on a reliable channel, this will result in a response from the other endpoint or an error from the transport interface. In the latter case, the procedures described above shall be used. An endpoint in a multipoint conference may use the same mechanism; however, it will learn only whether it still has a connection to the MC. Note that it is possible for an endpoint to have an error-free connection with the MC but still be receiving no audio or video from the rest of the terminals in the conference.

## 9 Interoperation with other terminal types

Interoperation with other terminals shall be accomplished through the Gateway. See 6.3 and Recommendation H.246.

### 9.1 Speech-only terminals

Interoperation with speech-only terminals (telephony) over the ISDN or GSTN can be provided by:

- 1) using a H.323-ISDN speech Gateway;
- 2) using a H.323-GSTN speech Gateway.

The Gateway should consider the following issues:

- Audio code conversion:
  - ISDN: if desired, since ISDN uses G.711.
  - GSTN: from analogue to G.711.
- Bit stream conversion:
  - ISDN: H.225.0 to/from unframed.
  - GSTN: generate H.225.0.
- Control conversion (generate H.245).
- Call Control Signalling conversion.
- DTMF tone conversion to/from H.245 **userInputIndication** message.

### 9.2 Visual telephone terminals over the ISDN (H.320)

Interoperation with visual telephone terminals over the ISDN (H.320) can be provided by:

- using a H.323-H.320 Gateway.

The Gateway should consider the following issues:

- Video format conversion. (If desired, H.261 is mandatory for both terminal types.)
- Audio code conversion. (If desired, G.711 is mandatory for both terminal types.)
- Data protocol conversion.
- Bit stream conversion. (H.225.0 to/from H.221.)
- Control conversion. (H.245 to/from H.242.)
- Call Control Signalling conversion.
- SBE Number conversion to/from H.245 **userInputIndication** message.

### 9.3 Visual telephone terminals over GSTN (H.324)

Interoperation with visual telephone terminals over the GSTN (H.324) can be provided by two methods:

- 1) using a H.323-H.324 Gateway;
- 2) using a H.323-H.320 Gateway, assuming that there exists an H.320-H.324 Gateway in the circuit switched network.

The Gateway should consider the following issues:

- Video format conversion. (If desired, H.261 is mandatory for both terminal types.)
- Data protocol conversion.
- Audio code conversion. (G.711 is mandatory for H.323 terminal, G.723.1 is mandatory for H.324 terminal.)

- Bit stream conversion. (H.225.0 to/from H.223.)
- Call Control Signalling conversion.

#### **9.4 Visual telephone terminals over mobile radio (H.324/M)**

For further study.

#### **9.5 Visual telephone terminals over ATM (H.321 and H.310 RAST)**

Interoperation with visual telephone terminals over ATM networks (H.321 and H.310 RAST terminals operating in H.320/H.321 interworking mode) can be provided by two methods:

- 1) using a H.323-H.321 Gateway;
- 2) using a H.323-H.320 Gateway, assuming that there exists an I.580 ISDN/ATM Interworking Unit in the network.

The Gateway should consider the following issues:

- Video format conversion. (If desired, H.261 is mandatory for both terminal types.)
- Data protocol conversion.
- Audio code conversion. (If desired, G.711 is mandatory for both terminal types.)
- Bit stream conversion. (H.225.0 to/from H.221.)
- Control conversion. (H.245 to/from H.242.)
- Call Control Signalling conversion.

#### **9.6 Visual telephone terminals over guaranteed quality of service LANs (H.322)**

Interoperation with visual telephone terminals over Guaranteed Quality of Service LANs (H.322) can be provided by:

- using a H.323-H.320 Gateway, assuming that there exists a GQOS LAN-ISDN Gateway in the network.

The Gateway should consider the following issues:

- Video format conversion. (If desired, H.261 is mandatory for both terminal types.)
- Data protocol conversion.
- Audio code conversion. (If desired, G.711 is mandatory for both terminal types.)
- Bit stream conversion. (H.225.0 to/from H.221.)
- Control conversion. (H.245 to/from H.242.)
- Call Control Signalling conversion.

#### **9.7 Simultaneous voice and data terminals over GSTN (V.70)**

Interoperation with Simultaneous Voice and Data Terminals over GSTN (V.70) can be provided by:

- using a H.323-V.70 Gateway.

The Gateway should consider the following issues:

- Audio code conversion. (G.711 to/from Annex A/G.729.)
- Data protocol conversion.
- Bit stream conversion. (H.225.0 to/from V.76/V.75.)
- Control conversion. (Both terminals use H.245.)
- Call Control Signalling conversion.

## 9.8 T.120 terminals on the packet based network

An H.323 terminal that has T.120 capability should be capable of being configured as a T.120-only terminal which listens and transmits on the standard T.120 well-known TSAP Identifier. This will allow the T.120 capable H.323 terminal to participate in T.120-only conferences.

A T.120-only terminal on the network shall be able to participate in the T.120 portion of multipoint H.323 conferences. See 6.2.7.1.

## 9.9 Gateway for H.323 Media Transport Over ATM

It is possible to transport H.323 media streams originating from non-ATM IP networks over an ATM network using H.323-to-H.323 gateways. This mechanism is described in AF-SAA-0124.000 [33].

## 10 Optional enhancements

### 10.1 Encryption

Authentication and security for H.323 systems is optional; however, if it is provided, it shall be provided in accordance with Recommendation H.235.

### 10.2 Multipoint operation

#### 10.2.1 H.243 Control and Indication

H.245 contains multipoint control and indication messages carried forward from H.243. These messages may be used to provide certain multipoint capabilities (such as chair control) by following the procedures defined in Recommendation H.243.

NOTE – Clause 15/H.243 contains guidance for the implementation of these capabilities using the T.120-series of Recommendations.

## 11 Maintenance

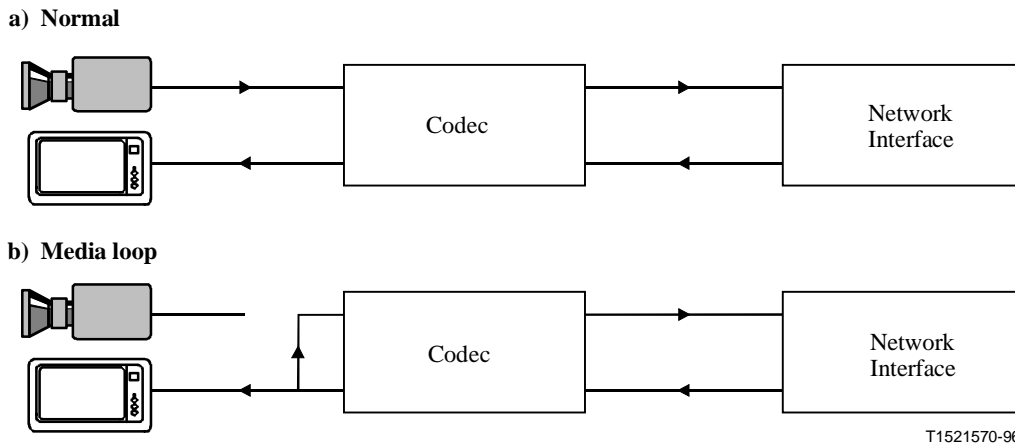
### 11.1 Loopbacks for maintenance purposes

Some loopback functions are defined in Recommendation H.245 to allow verification of some functional aspects of the terminal, to ensure correct operation of the system and satisfactory quality of the service to the remote party.

The **systemLoop** request and **logicalChannelLoop** request shall not be used. The **mediaLoop** request is optional. An endpoint that receives the **maintenanceLoopOffCommand** shall turn off all loopbacks currently in effect.

For the purpose of loopbacks, two modes are defined:

- a) Normal operation mode: No loopback. Indicated in **a)** of Figure 35. This shall be the default mode, and the mode entered when the **maintenanceLoopOffCommand** is received.
- b) Media loop mode: Loopback of media stream at the analogue I/O interface. Upon receiving the **mediaLoop** request as defined in Recommendation H.245, loopback of the content of the selected logical channel shall be activated as close as possible to the analogue interface of the video/audio codec towards the video/audio codec, so that decoded and re-coded media content is looped, as indicated in **b)** of Figure 35. This loopback is optional. It should be used only when a single logical channel containing the same media type is opened in each direction. Operation when multiple channels are opened in the return direction is undefined.



**Figure 35/H.323 – Loopback**

A Gateway to H.324, which receives an H.245 **systemLoop** request, H.245 **logicalChannelLoop** request, or a Gateway to H.320, H.321, or H.322, which receives an H.230 Dig-Loop command from an SCN endpoint may perform the appropriate loopback function within the Gateway. The Gateway shall not pass these requests to the network endpoint. A Gateway to H.324, receiving H.245 **mediaLoop** from an SCN endpoint shall pass the request to the network endpoint. A Gateway to H.320, H.321, or H.322, receiving H.230 Vid-loop or A $\mu$ -loop command from an SCN endpoint shall convert it to the appropriate H.245 **mediaLoop** request and send it to the network endpoint.

A Gateway to H.320, H.321, or H.322, which receives an H.245 **mediaLoop** request from a network endpoint shall convert it to the appropriate H.230 Vid-loop or A $\mu$ -loop command and send it to the SCN endpoint.

A Gateway to H.324 may send an H.245 **systemLoop** request or H.245 **logicalChannelLoop** request to the SCN endpoint. A Gateway to H.320, H.321, or H.322 may send an H.230 Dig-Loop command to the SCN endpoint. If a network endpoint is in a call to the SCN endpoint, the audio and video sent to the network endpoint may be the looped back audio or video, pre-recorded audio or video message indicating the loopback condition, or no audio or video.

## 11.2 Monitoring methods

All terminals shall support the Information Request/Information Request Response (IRQ/IRR) message of Recommendation H.225.0. The Information Request Response message contains the TSAP Identifier of all channels currently active on the call, including T.120 and H.245 control, as well as audio and video. This information can be used by third party maintenance devices to monitor H.323 conferences to verify system operation.

## ANNEX A

### H.245 messages used by H.323 endpoints

The following rules apply to the use of H.245 messages by H.323 endpoints:

- An endpoint shall not malfunction or otherwise be adversely affected by receiving H.245 messages that it does not recognize. An endpoint receiving an unrecognized request, response, or command shall return "function not supported". (This is not required for indications.)

- The following abbreviations are used in Tables A.1 to A.12:  
M Mandatory;  
O Optional;  
F Forbidden to transmit.
- A message marked as mandatory for the receiving endpoint indicates that the endpoint shall accept the message and take the appropriate action. A message marked as mandatory for the transmitting endpoint indicates that the endpoint shall generate the message under the appropriate circumstances.

**Table A.1/H.323 – Master-slave determination messages**

Message	Receiving endpoint status	Transmitting endpoint status
Determination	M	M
Determination Acknowledge	M	M
Determination Reject	M	M
Determination Release	M	M

**Table A.2/H.323 – Terminal capability messages**

Message	Receiving endpoint status	Transmitting endpoint status
Capability Set	M	M
Capability Set Acknowledge	M	M
Capability Set Reject	M	M
Capability Set Release	M	M

**Table A.3/H.323 – Logical channel signalling messages**

Message	Receiving endpoint status	Transmitting endpoint status
Open Logical Channel	M	M
Open Logical Channel Acknowledge	M	M
Open Logical Channel Reject	M	M
Open Logical Channel Confirm	M	M
Close Logical Channel	M	M
Close Logical Channel Acknowledge	M	M
Request Channel Close	M	O
Request Channel Close Acknowledge	O	O
Request Channel Close Reject	O	M
Request Channel Close Release	O	M

**Table A.4/H.323 – Multiplex table signalling messages**

Message	Status
Multiplex Entry Send	F
Multiplex Entry Send Acknowledge	F
Multiplex Entry Send Reject	F
Multiplex Entry Send Release	F

**Table A.5/H.323 – Request multiplex table signalling messages**

Message	Status
Request Multiplex Entry	F
Request Multiplex Entry Acknowledge	F
Request Multiplex Entry Reject	F
Request Multiplex Entry Release	F

**Table A.6/H.323 – Request mode messages**

Message	Receiving endpoint status	Transmitting endpoint status
Request Mode	M	O
Request Mode Acknowledge	M	O
Request Mode Reject	O	M
Request Mode Release	O	M

**Table A.7/H.323 – Round trip delay messages**

Message	Receiving endpoint status	Transmitting endpoint status
Round Trip Delay Request	M	O
Round Trip Delay Response	O	M

**Table A.8/H.323 – Maintenance loop messages**

Message	Receiving endpoint status	Transmitting endpoint status
Maintenance Loop Request		
System Loop	F	F
Media Loop	O (Note)	O (Note)
Logical Channel Loop	F	F
Maintenance Loop Acknowledge	O	O
Maintenance Loop Reject	O	M
Maintenance Loop Command Off	M	O
NOTE – Mandatory in Gateways.		

**Table A.9/H.323 – Conference Request and Response Messages**

Message	Receiving endpoint status	Transmitting endpoint status
Terminal List Request	O	O
Drop Terminal	O	O
Make Me Chair	O	O
Cancel Make Me Chair	O	O
Enter H.243 Password	O	O
Enter H.243 Terminal Id	O	O
Enter H.243 Conference ID	O	O
Request Terminal ID	O	O
Terminal ID Response	O	O
MC Terminal ID Response	O	O
Enter Extension Address	O	O
Enter Address Response	O	O
Terminal List Response	O	O
Make Me Chair Response	O	O
Conference ID Response	O	O
Password Response	O	O

**Table A.10/H.323 – Commands**

Message	Receiving endpoint status	Transmitting endpoint status
Send Terminal Capability Set	M	M
Encryption	O	O
Flow Control	M	O
End Session	M	M
<b>Miscellaneous Commands</b>		
Equalize Delay	O	O
Zero Delay	O	O
Multipoint Mode Command	M	O
Cancel Multipoint Mode Command	M	O
Video Freeze Picture	M	O
Video Fast Update Picture	M	O
Video Fast Update GOB	M	O
Video Fast Update MB	M	O
Video Temporal Spatial Trade Off	O	O
Video Send Sync Every GOB	O	O
Video Send Sync Every GOB Cancel	O	O



**Table A.10/H.323 – Commands (concluded)**

Message	Receiving endpoint status	Transmitting endpoint status
Terminal ID Request	O	O
Video Command Reject	O	O
Make Me Chair Response	O	O
<b>Conference Commands</b>		
Broadcast My Logical Channel Me	O	O
Cancel Broadcast My Logical Channel Me	O	O
Make Terminal Broadcaster	O	O
Cancel Make Terminal Broadcaster	O	O
Send This Source	O	O
Cancel Send This Source	O	O
Drop Conference	O	O

**Table A.11/H.323 – Conference mode commands**

Message	Receiving endpoint status	Transmitting endpoint status
Communication Mode Command	M	O
Communication Mode Request	O	O
Communication Mode Response	O	O

**Table A.12/H.323 – Indications**

Message	Receiving endpoint status	Transmitting endpoint status
Function Not Understood	M	M
Function Not Supported	M	M
<b>Miscellaneous Indication</b>		
Logical Channel Active	O	O
Logical Channel Inactive	O	O
Multipoint Conference	M	O
Cancel Multipoint Conference	M	O
Multipoint Zero Comm	O	O
Cancel Multipoint Zero Comm	O	O
Multipoint Secondary Status	O	O
Cancel Multipoint Secondary Status	O	O
Video Indicate Ready to Activate	O	O
Video Temporal Spatial Trade Off	O	O
Video Not Decoded MBs	O	O

**Table A.12/H.323 – Indications (concluded)**

Message	Receiving endpoint status	Transmitting endpoint status
<b>Conference Indications</b>		
SBE Number	O	O
Terminal Number Assign	M	O
Terminal Joined Conference	O	O
Terminal Left Conference	O	O
Seen By At Least One Other	O	O
Cancel Seen By At Least One Other	O	O
Seen By All	O	O
Cancel Seen By All	O	O
Terminal You Are Seeing	O	O
Request For Floor	O	O
Vendor Indications	O	O
MC Location Indication	M	O
Jitter Indication	O	O
H.223 Skew Indication	F	F
H2250MaximumSkewIndication	O	M
New ATM Virtual Channel Indication	F	F
User Input	M (for 0-9, *and #)	M (for 0-9, *and #)

Non-standard commands, requests, etc., are allowed.

## ANNEX B

### Procedures for layered video codecs

#### B.1 Scope

This annex describes enhancements within the framework of the H.323 specification, to incorporate layered video codecs. The described procedure is scalable for multipoint conferences.

#### B.2 Introduction

Layered video coding is a technique that allows the video information to be transmitted in multiple data streams in order to achieve scalability. These may provide bandwidth scalability, temporal scalability, SNR scalability, and/or spatial scalability. Annex O/H.263 describes the use of layered coding within H.263. Conferences can take advantage of this feature to service connected endpoints that have different capabilities, using one bitstream. This will allow more efficient use of network bandwidth.

### B.3 Scalability methods

Scalability of a video stream refers to the generation of a stream that may only be decoded in part due to limitations of available resources. Scalability may be desired to overcome limitations of available computing power, or to accommodate bandwidth limitations.

There are three types of scaling: Temporal, Signal-to-Noise Ratio (SNR), and Spatial that are available in Recommendation H.263. Other video codecs may have similar layering capability. All of these methods can be used separately, or together to create a multi-layer scalable bit stream. The resolution, frame rate, and quality of the image can only increase by adding scaling layers. The base layer can be used to guarantee a minimum level of image quality. Endpoints can then use additional layers to add image quality by increasing frame rate, display frame size, or accuracy of decoded images. Allowing multiple scaling methods in a conference can add resource efficiency, especially when endpoints participating have varying processing and bandwidth capabilities. This is especially true for multipoint and loosely-coupled conferences.

### B.4 Call establishment

H.323 call establishment takes place following the same procedures described in clause 8. The layered coding capability will be signalled using the H.245 capabilities exchange methods. Codepoints within H.245 exist which clearly identify what layering methods are supported by the endpoints. The endpoints shall use these capabilities in order to signal the exact layering methods they support.

The use of simultaneous capabilities methods in H.245 shall be used to indicate which layering methods will be used together to create the video layers when they are going to be sent in two or more logical channels. It is also possible to send two or more layers in single logical channels. The exact video layers that will be used are signalled during the **openLogicalChannel** in the same manner that is currently used to indicate what video **dataType** will be used, except that the endpoint shall indicate dependencies between the base layer logical channel and the enhancement layer logical channels.

### B.5 Use of RTP sessions and codec layers

It is desired to allow separate RTP sessions for the different qualities of video that are available. The base layer should be considered the primary video session, and its level considered the minimum quality of video that is available in the conference. Enhancement layers can be sent on separate RTP sessions. The **forward/reverseLogicalChannelDependency** parameter, added to H.245 **openLogicalChannel** command, shall be used to indicate how the video layers are organized. This is outlined in the following subclauses. RTP Timestamps must be the same in the base and all dependent enhancement layers corresponding to a frame to allow reassembly and proper display.

#### B.5.1 Associate base to audio for lip synchronization

The base video session should be associated with the audio session corresponding with the audio track of the video, for lip synch purposes. This is done in the same manner that existing non-layered video sessions are associated with their corresponding audio. This is done using the **associatedSessionID** and the **sessionID** parameters located in the **H2250LogicalChannelParameters**. The enhancement layers may also be associated with the audio or with the base layer using the **associatedSessionID**. Coding dependency shall be indicated using the **forwardLogicalChannelDependency** and the **reverseLogicalChannelDependency** parameter in the **openLogicalChannel** command as explained below.

## B.5.2 Enhancement layer dependency

Enhancement layer dependency can create many complex cases using multiple layers that contain multiple enhancement frame types. Dependency between layers shall be indicated using the **forward/reverseLogicalChannelDependency** parameter, added to H.245 **OpenLogicalChannel** command. Dependency is used to indicate that the data sent on the logical channel cannot be used without the contents of the logical channel it is dependent on. Enhancement layers, by definition, must be differentially coded from the video layer they are enhancing, and are therefore dependent on that video layer for meaningful decoding. If an enhancement layer is sent on a separate logical channel, it shall indicate the layer it was differentially coded from in the **forward/reverseLogicalChannelDependency** parameter.

Since the **forward/reverseLogicalChannelDependency** parameter allows the indication of a single logical channel, the logical channels need to be opened in order of dependence starting with the base layer. An endpoint shall have either sent or received the **openLogicalChannelAck** for any logical channel that is used in a **forward/reverseLogicalChannelDependency** parameter. An endpoint shall send an **openLogicalChannel** for a dependent logical channel, only after the logical channel on which it is dependent is opened and acknowledged. Logical channels that have common dependency may be opened in parallel. Enhancement layers must be indicated to be dependent on the highest layer that is required for proper decoding.

Assuming that separate RTP sessions are used for each layer, an example can be built as shown in Figure B.1.

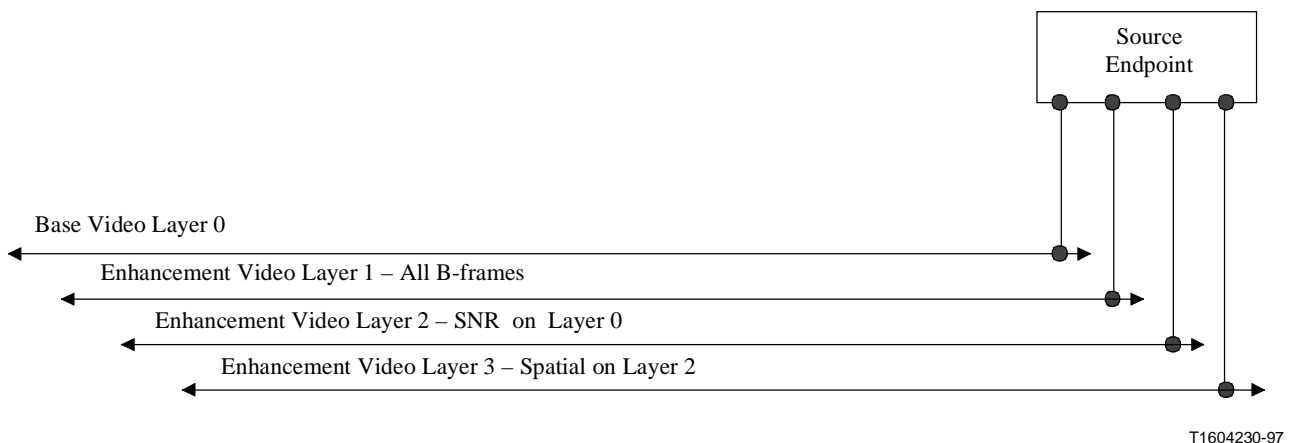


Figure B.1/H.323 – Model with layered video

In this example, layered video is created that has four layers:

- 1) The base video, not dependent on any other layer. This is associated with its corresponding audio.
- 2) Enhancement level one consisting of B-frames, dependent on the base video. This is indicated to be dependent on the base video session, Layer 0.
- 3) Enhancement level two that is SNR enhancement of the base video, dependent only on the base video, Layer 0. This is indicated to be dependent on the base video session.
- 4) Enhancement level three that consists of spatial enhancement of enhancement level two, dependent on Layer 2, which implies the base is also required. This is indicated to be dependent on the video in Layer 2.

In this example, the base video logical channel must be opened first. The **openLogicalChannel** for enhancement Layers 1 and 2 may be sent in parallel, only after receiving the **openLogicalChannelAck**

for the base video logical channel. The **openLogicalChannel** for enhancement Layer 3 can only be sent after the **openLogicalChannelAck** has been received or sent for the logical channel used for enhancement Layer 2.

## **B.6 Possible layering models**

There are many possible methods for layering of the video and organization of the corresponding RTP sessions. The reason that the layers may need to be separated is that they are used for either decoder power scaling, or for bandwidth usage scaling. It may be desirable to separate all non-B-frames into separate layers that can be discarded if they cannot be used. An important feature of the layered codec is that at any time an endpoint may discard any or all enhancement layers, without affecting the quality of the base video, in order to provide decoder power scaling.

In a similar manner, the layers may need to be organized into bandwidth usage levels that correspond to the bandwidths reported by the endpoints that are connected to the conference. This would allow the conference to accommodate multipoint conferences that have endpoints using connection methods that may limit the available bandwidth, and create a layer that gives them the best possible video at that bandwidth. The endpoint may add or subtract layers as its available bandwidth varies up and down.

### **B.6.1 Multiple logical channels and RTP sessions for a layered stream**

If bandwidth scaling is the goal of using layering, each layer should flow on a separate logical channel with a separate RTP session. This means that what is a single video source will now have to be coordinated amongst multiple logical channels and RTP sessions.

If the goal of layering is processor-power scaling, the enhancement layers can be sent, with the base video on a single logical channel and RTP session.

If the goal is a mixture of bandwidth and processor-power scaling, then groups of enhancement layers, sent in logical channels on a group basis can be sent. The choice of layers and grouping is a choice based on system need. The method used to make these choices is an implementation issue and outside the scope of this Recommendation.

### **B.6.2 Impact of one layer per logical channel and per RTP session**

The impact of using a single logical channel and RTP session for each layer is that the encoder and decoder are burdened with having to split and reassemble the video stream according to the chosen layering model. This model is signalled to the receiving side so that it can properly interpret the layer information. It is signalled using H.245 capabilities, with a capability per logical channel that, when combined with the dependencies, will sufficiently describe the layering model. Possible layering models are signalled during capabilities exchange, using the simultaneous capabilities feature of Recommendation H.245.

Strict timing consideration will need to be used to insure that the layers are properly synchronized. For H.323, this will be handled in the RTP payload format.

## **B.7 Impact on multipoint conferences**

The most likely envisaged usage of video layering is in multipoint conferences. In H.323, this can be performed by a centralized MCU, used for audio mixing and video switching, or using a decentralized model, with each endpoint responsible for video switching and audio mixing. In either case, the MC should perform the function of reporting what the layering model is for the conference. This is done using the **communicationModeCommand**.

In order for an endpoint to receive a video layer, a logical channel containing that layer must be opened. The decision to open a logical channel can be made by either the MC or the endpoint sending an **openLogicalChannel**. If an MC or endpoint decides not to open a logical channel, it must

reject the **openLogicalChannel** when it is offered. The MC or endpoint can only offer a logical channel that corresponds to a **dataType** that is supported by the receiving endpoint.

When implementing support for layer codecs, an MC can take two approaches. If the MC does not make any decisions as to what logical channels will be opened, it can be called the "MC Impartial" model. In this model the MC offers all media to all endpoints without regard to any reported QOS. When the MC makes the decision to strictly enforce QOS, it is called the "MC Decision" model. These models are explained further below.

### B.7.1 MC Impartial model

The MC Impartial model does not depend on the QOS capability set additions, and as such may allow for a simpler MC implementation. In this case, the endpoint must judge whether it has sufficient bandwidth to accept logical channels offered by the MC. If it will exceed the transmission capabilities of the endpoint or the underlying network, then the endpoint may reject the logical channel. This method will require the endpoint to have knowledge of the network bandwidth available. The MC should indicate all available media in the **communicationModeCommand**.

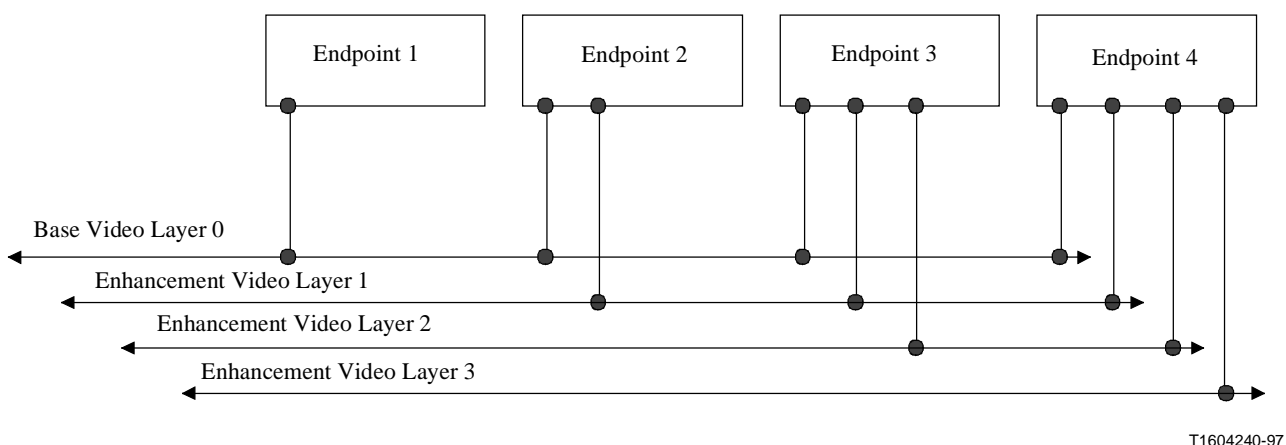
### B.7.2 MC Decision model

The MC Decision model depends upon the addition of Quality of Service (QOS) capabilities to the Terminal Capability Set. This has been previously proposed and is work in progress. The MC can then examine the QOS capabilities of the endpoints and only offer logical channels that are within the QOS of the endpoint. The endpoint will need to determine its available QOS at the start of the conference and indicate this using the QOS capabilities defined by work in progress.

In the MC Decision model, the MC may send a **communicationModeCommand** to an endpoint that only shows the sessions within the endpoint's QOS capabilities. In this way, the MC can strictly enforce bandwidth usage.

### B.7.3 Multipoint conference containing endpoints on different bandwidths

In the model where the multipoint conference contains endpoints that have different bandwidth capabilities, the layering will need to be tuned to match these bandwidth levels. This can be done by using two possible models. One is illustrated in Figure B.2.



**Figure B.2/H.323 – Endpoints attached to one or more layers according to bandwidth**

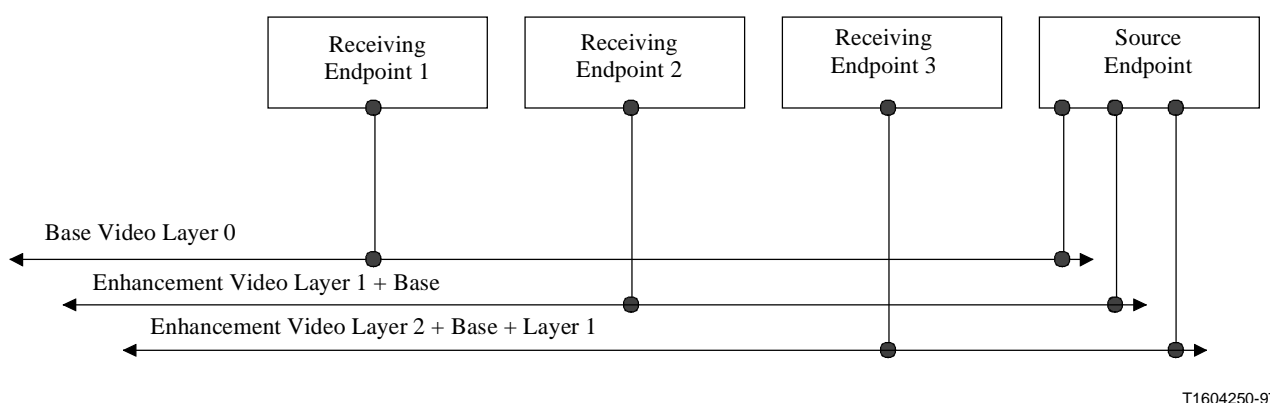
In this case, the endpoints are attached to the base layer of video and the enhancement layers up to the total bandwidth desired. Each enhancement layer is on a separate logical channel. The endpoints are burdened with recombination of the layers to create the video stream. The sending endpoint must have capability for the combined bandwidth of all streams it sources. In this case, each endpoint may

have communicated a different set of capabilities. The MC will examine the capabilities and QoS and create a layering model that is likely to provide the best use of the endpoints capabilities and bandwidth. This layering is indicated in the **communicationModeCommand** by the indication of **sessionDependency** in the **communicationModeTableEntry**. The **sessionDependency** field is set by the MC to indicate when a session is dependent on another session for meaningful decoding of its data. This information will be translated into **logicalChannelNumbers** when opening a dependent logical channel, according to the actual logical channels that are opened.

In the above case, using the MC Decision model, the MC will then offer the endpoints the logical channels that correspond to the layers that match the endpoint's capabilities. The MC will offer Endpoint 1 only the logical channel corresponding to the Base Video Layer. Endpoint 2 will be offered the logical channels corresponding to base video and enhancement video Layer 1. Endpoint three is offered three logical channels corresponding to the base video and two enhancements layers, and Endpoint 4 is offered all video logical channels.

In the MC impartial case, the MC will offer all logical channels, to all endpoints, that are within their **dataType** capabilities. The endpoints will refuse any logical channel that will cause them to exceed their bandwidth capabilities.

A second layering model is shown in Figure B.3. In this model each logical channel contains a totally independent video stream.



T1604250-97

**Figure B.3/H.323 – Endpoints attached to single Layer according to bandwidth**

In this case, the endpoint shall connect only to the logical channel that corresponds to the bandwidth it has available. This stream has all layers that build the video stream to the bandwidth of the logical channel. This method eliminates the burden from the endpoints to recombine the video, but burdens the sender with producing several video streams. This is a less efficient use of network resources, since enhancement layers include all lower layers.

In order to perform proper lip synch, any session containing base video should be associated with the audio session corresponding to its audio track, using the **associatedSessionID** in the **H2250LogicalChannelParameters**. In the example shown in Figure B.2, the base video session should be associated with the audio session for lip synch. In the example shown in Figure B.3, all three video sessions should be associated with the audio session for lip synch, since all three contain base video.

## **B.8 Use of network QOS for layered video streams**

Several important characteristics of the nature of layered coding usage should be considered when using network QOS for delivery of layered coded video streams. An enhancement layer cannot be decoded properly without receiving the layers on which it is dependent. Enhancement video layers may be discarded without affecting the decoding of the layer on which they are dependent.

If available, network QOS may be used to help guarantee that a video stream will be delivered by the network. Since layered video may be delivered using multiple streams, delivered on separate network connections, different QOS can be used on each video layer. QOS used on layered video streams should be specified when the logical channel is opened.

It is important that a dependent video layer has the information on which they are dependent at the time the dependent layer is to be decoded. This leads to general rules regarding use of QOS:

- 1) Dependent layers that are delivered using network QOS should have the layer they are dependent on, also delivered using QOS.
- 2) The base layer should be delivered using network QOS, if any other video layers in the conference are to be delivered using QOS.
- 3) The nearer the video layer is to the base layer, the stronger the delivery guarantees should be.

## **ANNEX C**

### **H.323 on ATM**

#### **C.1 Introduction**

This is an optional enhancement allowing H.323 endpoints to establish QOS-based media streams on ATM networks using AAL 5.

#### **C.2 Scope**

This annex specifies an improved method of using H.323 on AAL 5. H.323 can always be used on ATM by making use of an IP over ATM method. However, this is less efficient than using AAL 5 Virtual Channels (VCs) directly for the transport of the audio and video streams of H.323. When the media streams flow directly on AAL 5, they can benefit from a QOS-based ATM VC.

This annex retains the use of a packet network protocol for H.245 and H.225.0 communications to ensure interoperability with H.323 endpoints that are using a packet network protocol for all streams (whether over ATM or other media). Interoperability with legacy H.323 endpoints is achieved, without the use of a Gateway, by first requiring the basic mode of operation, in which an endpoint sends media streams on a datagram service using a packet network protocol, for example UDP/IP over ATM. In basic mode, unless a packet network protocol infrastructure has been upgraded, QOS may not be available from the network.

##### **C.2.1 Point-to-point conferencing**

This annex specifies a method of point-to-point communication between two H.323 endpoints using AAL 5 VCs for the media streams. The protocol necessary for entering into this mode is specified, as are information elements to be used in ATM signalling.

##### **C.2.2 MCU-based multipoint**

It follows that multipoint MCU-based communications can occur among several H.323 endpoints using AAL 5 VCs for the media streams. Currently no support is specified for the H.323 Decentralized Multipoint using ATM point-to-multipoint capability. This is left for further study.



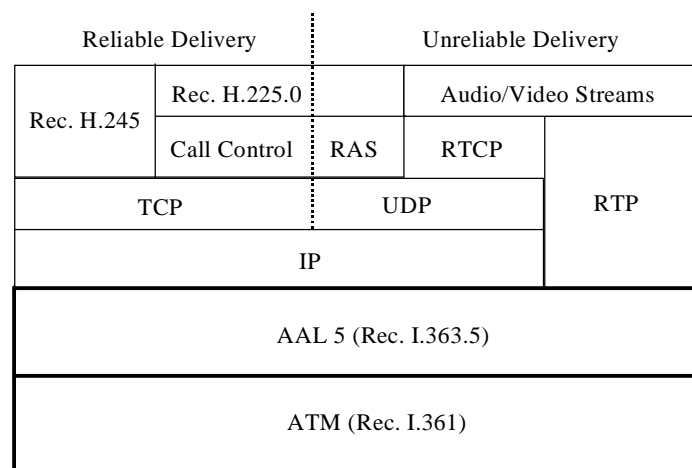
### C.2.3 H.323 interoperability with endpoints using IP

Interoperability is guaranteed with an endpoint using IP for the entire H.323 connection. This annex defines methods that allow an endpoint to detect if support is present for the option of using AAL 5 directly. An endpoint conforming to this annex must accept that the audio and video streams may occur on either AAL 5 VCs or UDP/IP ports.

### C.3 Architecture

The basic protocol architecture of the system is shown in Figure C.1. It uses IP on ATM for delivery of the H.225.0 and H.245 messages and for the RTCP part of the audio and video streams. It uses AAL 5 directly for the RTP part of the audio and video streams.

NOTE – The H.323 media streams, compressed into variable length packets according to Recommendation H.225.0, are easily mapped to AAL 5. It would be difficult to map them to AAL 1, and this alternative has no clear benefit.



T1604260-97

Figure C.1/H.323 – Architecture for H.323 on ATM-AAL 5

#### C.3.1 Overview of system

The system architecture is designed to make use of H.323, and its component protocols, as they are presently specified. It is further designed to use commonly available services of AAL 5 on ATM.

#### C.3.2 Interoperation with other ITU-T H-Series endpoints

Interoperation with other H-series endpoints shall be done through the use of Gateway devices as described in H.323. Gateway vendors will need to support the methods described in this annex, if they wish to support the direct use of AAL 5 VCs by H.323 endpoints.

It should be noted that interoperation with other IP-based H.323 endpoints does not require a Gateway.

#### C.3.3 H.225.0 on IP over ATM

H.225.0 communication requires TCP/IP and UDP/IP using one of the available methods for IP over ATM. No preference is expressed here for which method of IP over ATM to use. If two endpoints on the same network segment use different IP over ATM methods, they must rely on IP routers to forward their packets.

The endpoint shall listen on the well-known TCP ports identified in Recommendation H.225.0. If the endpoint is being used on a network with a Gatekeeper, the endpoint should use the methods described in H.225.0 to discover and register with the Gatekeeper. This requires the support of UDP multicast. If multicast is not available on the network, the endpoint may be pre-configured with the Gatekeeper(s) address(es).

The methods outlined in Recommendation H.225.0, combined with an IP over ATM method, shall be used to establish the H.245 control channel on TCP/IP.

### C.3.4 H.245 on TCP/IP over ATM

Once the reliable H.245 control channel has been established using methods described in Recommendation H.225.0, additional channels for audio, video, and data are established based on the outcome of the H.245 capability exchange using H.245 open logical channel procedures.

### C.3.5 Addressing for A/V streams

H.323 has the capability for the audio and video streams to be established to a different address than the H.245 control channels. This is fortunate since a TCP/IP channel is established to an IP address, and the audio and video, optionally, are to be sent on RTP over AAL 5 directly to an ATM address.

H.323 also has the capability for the RTCP stream to be addressed separately from the RTP stream. The RTCP stream shall continue to be addressed to an IP address, even though the RTP stream is addressed to an ATM address.

### C.3.6 Transport Capabilities added to TransportCapability Set

For operation of H.323 on AAL 5, an addition to the **TransportCapability** set is made in H.245. This includes transport level capabilities such as support for ATM Transfer Capability (DBR, SBR1, SBR2, SBR3, ABT/DT, ABT/IT, ABR) as defined in Recommendation I.371. Terminals that do not send this new capability parameter shall not make use of the new methods described in this annex. The **TransportCapability** information can be sent as part of the Terminal Capability set exchange in the capability exchange phase. It is also included in the **openLogicalChannel**.

### C.3.7 Elements of ATM signalling

#### C.3.7.1 ATM address

The ATM address for an RTP stream shall be given in the **mediaChannel** subfield of **H2250LogicalChannelParameters** of the H.245 **openLogicalChannelAck** message (or the **openLogicalChannel** in the case of Fast Connect). The **mediaChannel** subfield **UnicastAddress** or **MulticastAddress** shall be filled with the 20-octet NSAP-style ATM End System Address.

The use of E.164 for the address is handled by embedding it as the IDP part (AFI = 0x45) of an NSAP address. In this case, an international E.164 number is required.

#### C.3.7.2 Port Number

The **portNumber** field of the **openLogicalChannel** message is conveyed in the GIT information element as per [34]. The format of the GIT information element is specified in C.4.1.1. This enables the receiving side to associate the ATM VC with the proper RTP logical channel.

For backward compatibility with H.323 version 2 endpoints, H.323 version 3 (and later) endpoints shall also be able to use the B-HLI, according to H.323 version 2 Annex C, for conveying the **portNumber** field of **openLogicalChannel**. An H.323 version 3 (or later) endpoint shall use the B-HLI only if it has prior knowledge that the terminating endpoint is H.323 version 2. In cases where the H.323 version of the terminating endpoint is not known, such as establishing a call using Fast Connect, the endpoints shall first attempt to establish the ATM VC using the GIT information element for carrying the **portNumber**. If the connection fails the calling endpoint shall re-attempt call

setup using B-HLI instead of GIT. If the VC setup with B-HLI also fails, the terminal shall assume that ATM connectivity is not available and shall fall back to using RTP/UDP/IP for media channels. The format of the B-HLI information element is specified in C.4.1.2.

### C.3.8 A/V streams on RTP on AAL 5

Servicing the **openLogicalChannel** primitive in H.245 triggers the connection establishment. The audio and video streams are then set up to the destination ATM address. The size of the Maximum Transmission Unit (MTU) shall be signalled in the AAL Parameters information element. The MTU choice may effect system efficiency because of AAL 5 packetization. The packetization rules for AAL 5 are contained in Recommendation I.363.5. If the non-AAL 5 default of 1536 octets is used, the MTU is packetized in 33 ATM cells and the last AAL 5 cell contains only padding and the AAL 5 number. The address field in the **mediaChannel** should be used to determine whether an ATM VC or a UDP port should be opened.

In the event that the ATM VC setup fails, the endpoint shall retry using RTP/RTCP and the higher layer transport protocol such as UDP.

RTP header compression can optionally be used, as described in section 2 of AF-SAA-0124.000 [33], in which case it must be negotiated using the **mediaTransportType**.

#### C.3.8.1 Unidirectional logical channels

H.323 has no concept of the reverse direction of a unidirectional logical channel. However, an important characteristic of point-to-point ATM VCs is that they are inherently bidirectional. The use of both directions of an ATM VC is therefore desirable. Otherwise, the audio and video streams will each need to be sent on two different VC's, one for each direction.

Endpoints conforming to this annex are encouraged to open their media streams as bidirectional logical channels. This reduces the number of AAL 5 VCs to two in typical situations, one VC each for audio and for video.

#### C.3.8.2 Bidirectional logical channels

If the bidirectional usage is indicated, the receiving endpoint shall send an **openLogicalChannelAck** (or the **openLogicalChannel** in the case of Fast Connect) and then it must watch for an ATM VC to be opened by the other endpoint. When ATM VC is completed, it may then use the reverse direction for the media type indicated in the **openLogicalChannel** command. The endpoint that initiates the **openLogicalChannel** command is the endpoint that shall open the ATM VC.

If QOS is to be used, it shall be limited to the **H2250Capability** declared by the other endpoint. The chosen QOS is signalled as part of the establishment of an ATM VC.

If both endpoints have uncompleted **openLogicalChannel** commands for the same media session, these are resolved using the master/slave methods described in Recommendation H.245.

#### C.3.8.3 Maximum transmission unit size

The maximum MTU for AAL 5 is 65 535 octets. As part of **H2250Capability**, the MTU size can be specified in the capabilities exchange during H.245 setup. The forward and backward maximum MTU size shall be equal and will be taken from the smallest of the local and remote values specified in the capabilities exchange.

The MTU size is signalled as the AAL 5 maximum CPCS-PDU size for an ATM VC.

#### C.3.8.4 RTCP on IP over ATM

It is mandatory to open the logical channel for RTCP traffic on a UDP/IP port, using IP over ATM. RTCP is not permitted to ride directly on an AAL 5 VC.

### C.3.9 QOS considerations (Optional)

#### C.3.9.1 QOS classes defined in Recommendation I.356

Recommendation I.356 defines four QOS classes, Class 1 (stringent class), Class 2 (tolerant class), Class 3 (bi-level class), and U class. Table C.1 summarizes the differences among the QOS classes.

**Table C.1/H.323 – Provisional QOS class definitions and network performance objectives**

	CTD	2-pt. CDV	CLR (0+1)	CLR (0)	CER	CMR	SECBR
Default	None	None	None	None	$4 \times 10^{-6}$	1/day	$10^{-4}$
Class 1 (stringent)	400 ms	3 ms	$3 \times 10^{-7}$	None	Default	Default	Default
Class 2 (tolerant)	U	U	$10^{-3}$	None	Default	Default	Default
Class 3 (bi-level)	U	U	U	$10^{-5}$	Default	Default	Default
U class	U	U	U	U	U	U	U

CTD: Cell Transfer Delay; CDV: Cell Delay Variation; CLR: Cell Loss Ratio; CER: Cell Error Ratio; CMR: Cell Misinsertion Rate; SECBR: Severely Errored Cell Block Ratio; U: Unspecified/ unbounded.

#### C.3.9.2 ATM transfer capability defined in Recommendations I.371 and I.371.1

ATM Transfer Capability (ATC), defined in Recommendations I.371 and I.371.1 as a set of ATM layer parameters and procedures, is intended to support an ATM layer service model and a range of associated QOS classes. Open-loop control ATCs (DBR and SBR) and closed-loop controlled ATCs (ABT and ABR) are specified in Recommendations I.371 and I.371.1. SBR is subdivided into SBR1, SBR2 and SBR3, depending on how to handle CLP = 0/1 cells. ABT is subdivided into ABT/DT and ABT/IT depending on the use of negotiation regarding the block cell rate. Table C.2 summarizes the association of ATCs with QOS classes.

**Table C.2/H.323 – Association of ATCs with QOS classes (from Table 3/I.356)**

ATM Transfer Capabilities (ATC)	DBR, SBR1, ABT/DT, ABT/IT	DBR, SBR1, ABT/DT, ABT/IT	SBR2, SBR3, ABR	Any ATC
Applicable QOS class	Class 1 (stringent)	Class 2 (tolerant)	Class 3 (bi-level)	U class

DBR: Deterministic Bit Rate; SBR1: Statistical Bit Rate configuration 1; ABT/DT: ATM Block Transfer/Delayed Transmission; ABT/IT: ATM Block Transfer/Immediate Transmission; SBR2: Statistical Bit Rate configuration 2; SBR3: Statistical Bit Rate configuration 3; ABR: Available Bit Rate.

#### C.3.9.3 Broadband transfer capability defined in Recommendation Q.2961.2

Broadband Transfer Capability (BTC) codes (DBR, BTC5, BTC9, BTC10 and SBR1) in Broadband bearer capability information element are defined in Recommendation Q.2961.2, and valid combinations of bearer class, broadband transfer capability and ATM traffic descriptor parameters are specified in Annex A/Q.2961.2. In the Setup message, the user can specify the BTC according to the traffic he/she generates and the intended use of network services. In Table A.1/Q.2961.2, 3 valid combinations are listed for bearer class BCOB-A, 8 combinations for BCOB-C, and 13 combinations for BCOB-X or FR.

### C.3.9.4 Opening of Virtual Channels

The endpoint that originated the accepted **openLogicalChannel** is responsible for opening the ATM VC. Support for QOS in the ATM VC is signalled at the time it is established. If successful, the ATM network provides a guaranteed QOS for the lifetime of the opened VC. QOS is specified in terms of Q.2931 Information Elements (IEs), including ATM Traffic Descriptor and Broadband Bearer Capability.

### C.3.9.5 Use of DBR

The most likely available ATM traffic type is a constant bit rate using DBR. The use of DBR is signalled as part of the ATM broadband Bearer Capability IE (Bearer class = "BCOB-A"). Use of other ATM traffic type, such as SBR with end-to-end timing required [Bearer class = "BCOB-X" and BTC field = "SBR1 (0010011)"], is also possible.

### C.3.9.6 Setting the proper cell rate

It is important to set the proper cell rate parameters in the ATM Traffic Descriptor information element. The peak cell rate can be derived from the H.245 capabilities exchange parameters and the RTP payload format packet size. For video, the **maxBitRate** field can be used from the **H261VideoCapability** or the **H263VideoCapability** to determine the ATM Cell rate. For audio, the audio capability chosen implies the bit rate to be used. For example, the use of **g711Ulaw64k** suggests the use of a 64 kbit/s audio channel, while the use of **g728** indicates the use of a 16 kbit/s channel. The RTP payload format indicates the packet size. For each packet, the subsequent AAL packet overhead and any needed padding to meet the AAL packetization rules, must be added. This results in an overhead bit rate that is associated with the size of the packet and the way this packet is encapsulated in the AAL, and the frequency of this overhead from this encapsulation.

The bit rate of the data to be sent, and the packetization of the data according to the AAL packetization rules determine the cell rate. The packetization will determine the actual number of cells that must be sent for a given data stream at a given bit rate. The choice of MTU can affect the packetization as explained in C.3.8.

## C.4 Protocol section

### C.4.1 ATM signalling information elements

#### C.4.1.1 Generic Information Transport

IE Parameter	Value	Notes
Identifier related standard/application (octet 5)	0000 1011	Recommendation H.323
Identifier Type (octet 6)	0000 1011	H.245 <b>portNumber</b>
Identifier length (octet 6.1)	0000 0010	2 octets
Identifier value (octets 6.2-6.3)	H.245 <b>portNumber</b>	16-bit binary coded forward H.245 <b>portNumber</b>

H.323 version 3 (or later) endpoints shall set the IE action indicator of the GIT information element to "clear call", according to 4.5.1/Q.2931. In this case, if the terminating endpoint does not support GIT information element coding it will reject the call with the cause value 100 for invalid information element content according to 5.7.2/Q.2931. If the ATM VC setup attempt is rejected because the terminating endpoint does not understand GIT it will reject the VC call setup with cause number 99 "information element non-existent or not implemented", according to 5.7.2/Q.2931.

It should be noted that the **portNumber** field in H.245 is only 16 bits in length.

The H.245 **portNumber** is used by the receiving endpoint to associate the ATM VC with the proper RTP logical channel. The endpoint that initiates the **openLogicalChannel** command is the endpoint that opens the ATM VC. It is possible for the initiating endpoint to select an H.245 **portNumber** that is already in use by the receiving endpoint. This would cause a failure in the OLC procedure.

Additionally the receiving RTCP port is also specified by the initiating endpoint by implication. H.323 states that the corresponding RTCP data shall flow on a UDP port number equal to the H.245 **portNumber** plus 1. It is possible that the resulting port number for RTCP, H.245 **portNumber** plus 1, will be in use on the receiving endpoint since the H.245 **portNumber** is selected by the initiating endpoint.

Due to the above problems the receiving endpoint should have the choice of selecting the H.245 **portNumber**. If the **portNumber** is not specified in the **openLogicalChannel** the receiving endpoint shall specify a **portNumber** in the **openLogicalChannelAck** message (or **openLogicalChannel** in the case of Fast Connect). It is recommended that the transmitting endpoint does not specify the **portNumber** in the **openLogicalChannel** thereby requiring the receiving endpoint to specify one in the **openLogicalChannelAck** message (or **openLogicalChannel** in the case of Fast Connect).

The **portNumber** field of the **openLogicalChannel** message is used to select the H.245 **portNumber**. The receiving endpoint uses this H.245 **portNumber** to associate the ATM VC with the proper RTP logical channel. If the receiving endpoint finds that the given H.245 **portNumber** is inappropriate it can select a new H.245 **portNumber** and use the **portNumber** field of the **openLogicalChannelAck** message (or **openLogicalChannel** in the case of Fast Connect) to indicate the new value to the initiating endpoint. The selected H.245 **portNumber** field is conveyed in the GIT information element. This enables the receiving side to associate the ATM VC with the proper RTP logical channel.

The VC association port number is represented in network byte order in octets 6.2 and 6.3 of the GIT (i.e. octet 6.2 holds the MSB and octet 6.3 holds the LSB).

#### C.4.1.2 Broadband High Layer Information

IE parameter	Value	Notes
Length of B-HLI contents (octets 3-4)	3	
High layer information type (octet 5)	"0000 0001"	User-specific
High layer information (octets 5-7)	H.245 <b>portNumber</b>	16-bit binary coded forward H.245 <b>portNumber</b>

The B-HLI is only used for backward compatibility with H.323 version 2 endpoints, as described in C.3.7.2.

#### C.4.1.3 ATM Adaptation Layer Parameters

IE parameter	Value	Notes
AAL type (octet 5)	"0000 0101"	AAL 5
Forward maximum AAL 5 CPCS-SDU size (octets 6.1-6.2)	MTU size	The smaller <b>mTUsize</b> in the local and remote <b>QOSCapability.atmParms</b>
Backward maximum AAL 5 CPCS-SDU size (octets 7.1-7.2)	MTU size	Same as forward
SSCS type (octet 8.1)	"0000 0000"	Null SSCS

#### C.4.1.4 ATM Broadband bearer capability Information Element

a) In the case where the ATM traffic type in H.245 is equal to "DBR":

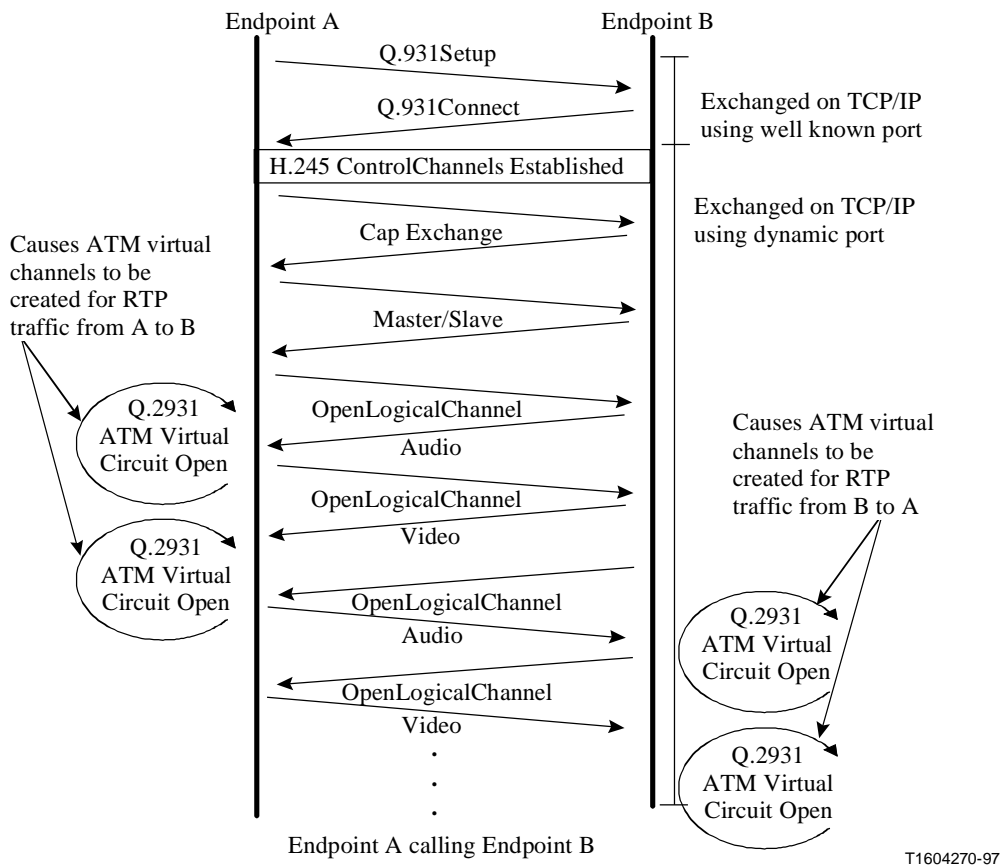
IE parameter	Value	Notes
Bearer class	BCOB-A	
Susceptibility to clipping	Susceptible to clipping	
User-plane connection configuration	Point-to-point	

b) In the case where ATM traffic type in H.245 is equal to "SBR1" with end-to-end timing required:

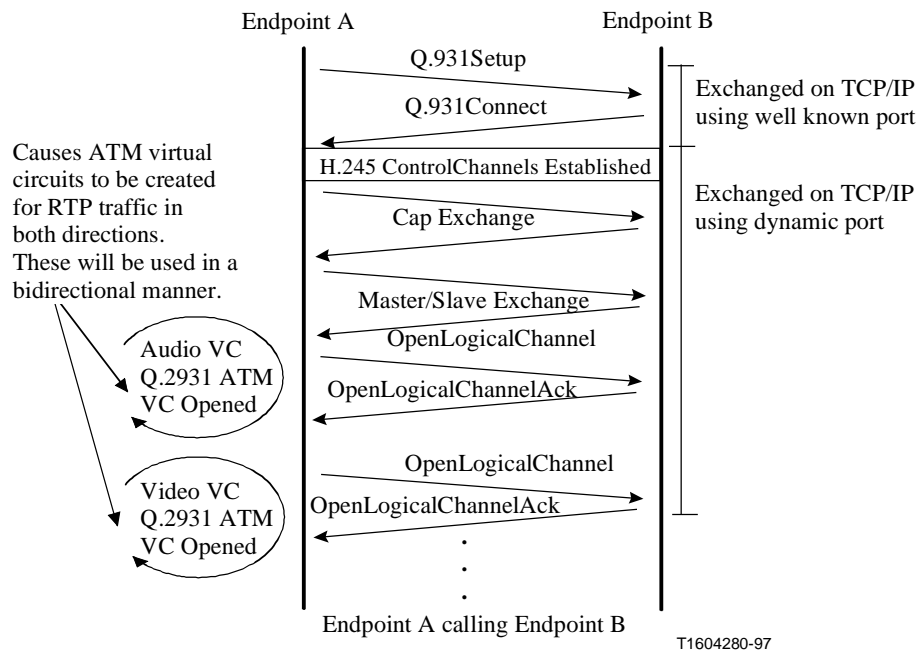
IE parameter	Value	Notes
Bearer class	BCOB-X	
Broadband bearer capability	"0010011" (SBR1)	SBR1 with end-to-end timing required
Susceptibility to clipping	Susceptible to clipping	
User-plane connection configuration	Point-to-point	

#### C.4.2 H.245 Usage

The establishment of a H.323 call using AAL 5 media streams is done in a manner similar to the basic mode of H.323 on IP. The difference is that the completed **openLogicalChannel** exchange in H.245 should result in an AAL 5 VC being established. This is illustrated in Figures C.2 and C.3 for the unidirectional VC usage and the bidirectional VC usage respectively.



**Figure C.2/H.323 – H.323 call establishment showing ATM effect – ATM VC's used unidirectionally**



**Figure C.3/H.323 – H.323 call establishment showing ATM effect – ATM VC's used bidirectionally**



It should be noted that the ATM VC setups will occur in only one direction if bidirectional logical channels are used. In this case, the endpoint acknowledging the **openLogicalChannel** will merely bind the incoming ATM connection to an RTP session using the VC Association port number.

### C.4.3 RTP usage

RTP and RTCP are defined in Annex A/H.225.0. RTCP is currently required for all H.323 connections, and therefore is required even when using an AAL 5 VC. The RTCP is carried by UDP/IP, not directly by the AAL 5 VC.

### C.4.4 Interoperation with H.323 on IP

Since the H.225.0 and H.245 communications are on IP, the endpoint will be able to receive calls from any other endpoint that is properly connected to the IP network. It is possible that H.323 endpoints will be used on ATM that do not support the methods described in this annex. They will strictly follow the basic method of using UDP/IP for the A/V streams. In this case, the endpoint will not declare the new **transportCapabilities** in H.245 and will refuse to open logical channels using ATM addressed VCs.

The protocol to **openLogicalChannel** using AAL 5 VC's for A/V streams should only be used if the received Capabilities have indicated that the method of this annex is supported. If this capability parameter is not present in the Terminal Capability Set, then the endpoint should only use **openLogicalChannel** using UDP/IP over ATM. This will insure that the endpoint can communicate with other endpoints that support H.323, but may not support the methods in this annex.

## ANNEX D

### Real-time facsimile over H.323 systems

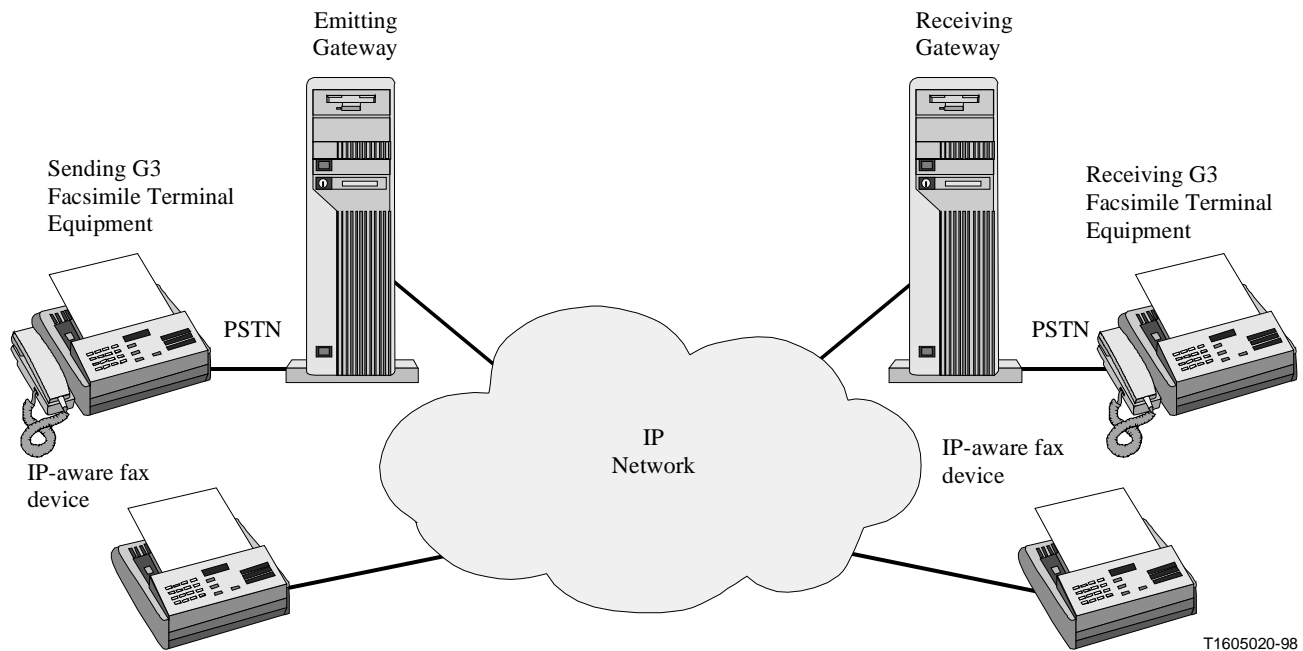
#### D.1 Introduction

Currently, facsimile and speech are typically sent using the PSTN with the same calling and addressing infrastructure. It is highly desirable to continue this approach in the context of this Recommendation. From a high level, facsimile can be viewed as another kind of real-time traffic similar to a particular speech coder. This seems appropriate, as facsimile entering the packet world via a gateway from the PSTN should logically be treated in a fashion similar to speech if the customer expects a real-time, assured end-to-end transmission service. The conversion of facsimile to email or other store-and-forward methods represents a new service that is beyond the scope of this Recommendation, which is a real-time protocol. It is recognized that manufacturers may wish to provide a gateway that falls back to a store-and-forward service when the real-time facsimile call fails. It is beyond the scope of this Recommendation when and how this decision is made, or by what means a store-and-forward facsimile service is implemented.

Recommendation T.38 [1] defines an Internet facsimile protocol consisting of messages and data exchanged between Facsimile Gateways connected via an IP network. This annex uses Recommendation T.38. Communication between the Gateways and G3/G4 Facsimile terminals is beyond the scope of Recommendation T.38. The reference model for T.38 is shown in Figure D.1 with three scenarios. In the first scenario, the two traditional Group 3 Facsimile Equipment (G3FE) terminals are virtually connected through the Gateways once the PSTN calls are established. All T.30 session establishment and capabilities negotiation is carried out between the terminals. In the second scenario, the traditional Group 3 Facsimile (IAF) terminal is connected with an Internet Aware Fax terminal (IAF).

The IAF is directly connected to the IP network. In the third scenario, the two IAFs are directly connected to the IP network. In all the scenarios, T.38 packets are used on the IP network to

communicate T.4/T.30 facsimile information. The transport of T.38 packets is either on TCP/IP or UDP/IP using the mechanism of this Recommendation.



**Figure D.1/H.323 – Model for facsimile transmission over IP networks**

## D.2 Scope

The scope of this annex is to use H.323 procedures to transfer T.38 packets in real time over the IP network. H.323 entities supporting facsimile capabilities shall use T.38 to support real-time facsimile services as described in this annex.

H.323 facsimile capable endpoints shall support the usage of TCP and UDP as described in Recommendation T.38. Annex B/T.38 describes a T.38-only capable terminal that supports a subset of H.245 messages using H.245 tunnelling. However, the Annex B/T.38 terminal can interwork with an Annex D/H.323 terminal using 8.1.7 "Fast Connect Procedure", and 8.2.1 "Encapsulation of H.245 Messages within Q.931 Messages" procedures in this Recommendation. Annex B/T.38 terminals interwork with H.323 terminals without being conformant to Recommendation H.323. An H.323 terminal that supports the procedures of this annex shall interwork with Annex B/T.38 terminals.

## D.3 Normative references

The following ITU-T Recommendations and other references contain provisions which, through reference in this text constitute provisions of this Recommendation. At the time of publication the editions indicated were valid. All Recommendations and other references are subject to revision; all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

- ITU-T Recommendation T.38 (1998), *Procedures for real-time Group 3 facsimile communication over IP networks*.

## D.4 Procedures for opening channels to send T.38 packets

Fast Connect is used to describe the H.323 procedures for opening channels for the transportation of T.38 packets. The traditional sequence can also be used, though it is not described here.

### D.4.1 Opening the voice channel

Zero, one (sender to receiver channel or receiver to sender channel), or two (sender to receiver channel and receiver to sender channel) logical channels for voice can be opened depending on the capability of the sender and the receiver. The voice channel shall be opened as specified by the procedures in 8.1.7 "Fast connect". Support of voice by facsimile applications is not mandatory in this annex.

### D.4.2 Opening the facsimile channels

Two logical channels (sender to receiver channel and receiver to sender channel) shall be opened for the transfer of T.38 packets. T.38 packets can be transferred using either TCP or UDP. In general, the usage of TCP is more effective when the bandwidth for facsimile communication is limited. On the other hand, the usage of UDP may be more effective when the bandwidth for facsimile communication is sufficient.

The sender terminal specifies a TCP/UDP port in the **OpenLogicalChannel** in the **fastStart** element of *Setup*. The receiver terminal shall provide its TCP (or UDP) port in the **OpenLogicalChannel** of the **fastStart** element as specified by the procedures in 8.1.7 "Fast connect procedure".

The receiver shall open the TCP/UDP port based on the preference of the sender. If the sender terminal has a preference for UDP or TCP then it shall provide its preference in the **OpenLogicalChannel** with the appropriate port in the **fastStart** sequence. The receiving terminal can select the transport, TCP or UDP, by specifying one of the two in **OpenLogicalChannel** structures in the **fastStart** element of *Connect*.

### D.4.3 DTMF transmission

DTMF tones shall be sent by Annex D/H.323 terminals using **UserInputIndication** to interwork with Annex B/T.38 terminals. Annex D/H.323 terminals may send DTMF tones in-band with the voice when Annex B/T.38 terminals are not involved in the call.

## D.5 Non-Fast Start procedures

It is noted that in Non-Fast Start, the normal H.245-based **OpenLogicalChannel** procedures can be used to open and close both UDP and TCP fax channels. Tunnelled H.245 can also be used to open and close channels. It is also noted that non-fast start and non-tunnelled H.245 procedures do not apply to interworking with Recommendation T.38.

## D.6 Usage of the maxBitRate in messages

When TCP is used for T.38 fax transmission, **maxBitRate** in the ARQ/BRQ does not include the fax data rate, and if a voice link is switched off when the fax session starts, a BRQ shall be used to indicate to the Gatekeeper that the bandwidth has changed. When UDP is used for T.38 fax transmission, **maxBitRate** in the ARQ/BRQ does include the bit rate needed for the fax session. The endpoint (terminal, gateway) shall send BRQs to the gatekeeper as bandwidth needs change during the call. It is noted that the **maxBitRate** in the **OpenLogicalChannel** element in the *Setup* during fast start is different from the **maxBitRate** in ARQ/BRQ, and does refer to the peak bit rate that the fax call will use.

## D.7 Interactions with gateways and Annex B/T.38 devices

The following case must be considered:

Annex D/H.323 device (with voice) <-> Annex B/T.38 device (without voice)

Note that these devices may be terminals or gateways; it does not affect the discussion. A fax call arrives from the "voiceless" side, but the voice side must generate an outgoing voice call that is not connected to anything although tones or announcements might be played. In the opposite direction, the Annex D/H.323 device cannot offer a voice call to a "voiceless" device, as it cannot receive voice.

The Annex D/H.323 gateway shall send both a voice and fax **OpenLogicalChannel** element in the Setup message. If it encounters a T.38 device, only the fax channel will be opened. If the call mistakenly encounters a non-fax H.323 device, the fax port will not be opened. This is the equivalent of a fax machine calling a telephone.

An Annex D/H.323 device becomes aware that it is talking to a Annex B/T.38 device due to the following sequence of events:

- 1) The Annex B/T.38 device does not supply an H.245 port in the Connect or Setup.
- 2) The Annex D/H.323 device uses the Facility message as described in 8.2.3 and transmits a **FACILITY** message with a **FacilityReason** of **startH245** and provides its H.245 address in the **h245Address** element. The Annex B/T.38 endpoint receiving a **FACILITY** message with a **FacilityReason** of **startH245** will respond with a **FACILITY** message having a **FacilityReason** of **noH245**. At this point the Annex D/H.323 device should cease all attempts to open the H.245 channel.

## APPENDIX I

### Sample MC to Terminal Communication Mode Command

#### I.1 Sample conference Scenario A

Endpoints A, B and C are in an audio and video distributed conference using multicast. The MC (which could be anyone of the nodes) has decided to place the media and media control channels on the following multicast addresses:

Stream	Multicast address
Audio for all endpoints:	MCA1
Audio Control for all endpoints:	MCA2
Video from endpoint A:	MCA3
Video Control data about endpoint A:	MCA4
Video from endpoint B:	MCA5
Video Control data about endpoint B:	MCA6
Video from endpoint C:	MCA7
Video Control data about endpoint C:	MCA8

## I.2 CommunicationModeTable sent to all Endpoints

All entries are commands for endpoints to open a logical channel for transmission. **terminalLabel** is only present when the entry is specific to a single endpoint in the conference.

### ENTRY 1 – AUDIO & AUDIO CONTROL FOR CONFERENCE

<b>sessionID</b>	<b>1</b>
<b>sessionDescription</b>	<b>Audio</b>
<b>dataType</b>	<b>Audio Capability</b>
<b>mediaChannel</b>	<b>MCA1</b>
<b>mediaControlChannel</b>	<b>MCA2</b>

### ENTRY 2 – VIDEO & VIDEO CONTROL FOR NODE A

<b>sessionID</b>	<b>2</b>
<b>associatedSessionID</b>	<b>1</b>
<b>terminalLabel</b>	<b>M/T for A</b>
<b>sessionDescription</b>	<b>Video for Node A</b>
<b>dataType</b>	<b>Video Capability</b>
<b>mediaChannel</b>	<b>MCA3</b>
<b>mediaControlChannel</b>	<b>MCA4</b>

### ENTRY 3 – VIDEO & VIDEO CONTROL FOR NODE B

<b>sessionID</b>	<b>3</b>
<b>associatedSessionID</b>	<b>1</b>
<b>terminalLabel</b>	<b>M/T for B</b>
<b>sessionDescription</b>	<b>Video for Node B</b>
<b>dataType</b>	<b>Video Capability</b>
<b>mediaChannel</b>	<b>MCA5</b>
<b>mediaControlChannel</b>	<b>MCA6</b>

### ENTRY 4 – VIDEO & VIDEO CONTROL FOR NODE C

<b>sessionID</b>	<b>4</b>
<b>associatedSessionID</b>	<b>1</b>
<b>terminalLabel</b>	<b>M/T for C</b>
<b>sessionDescription</b>	<b>Video for Node C</b>
<b>dataType</b>	<b>Video Capability</b>
<b>mediaChannel</b>	<b>MCA7</b>
<b>mediaControlChannel</b>	<b>MCA8</b>

## I.3 Sample conference Scenario B

Endpoints A, B and C are in a multipoint conference where audio is unicast from each endpoint and centrally mixed, but video is multicast from the endpoints. The MC may send a unique CommunicationModeCommand to each endpoint, or it may send the same message to all endpoints if the table entries are identified by the destination endpoint's label. For this example, assume that the same message is sent to all endpoints.

<b>Stream</b>	<b>Multicast Address</b>
Audio from endpoint A:	UCA1
Audio Control data about endpoint A:	UCA2
Audio from endpoint B:	UCA3
Audio Control data about endpoint B:	UCA4
Audio from endpoint C:	UCA5
Audio Control data about endpoint C:	UCA6
Video from endpoint A:	MCA1
Video Control data about endpoint A:	MCA2
Video from endpoint B:	MCA3
Video Control data about endpoint B:	MCA4
Video from endpoint C:	MCA5
Video Control data about endpoint C:	MCA6

#### **I.4 CommunicationModeTable sent to all Endpoints**

All entries are commands for endpoints to open a logical channel for transmission. **terminalLabel** is only present when the entry is specific to a single endpoint in the conference.

##### **ENTRY 1 – AUDIO & AUDIO CONTROL FOR NODE A**

**sessionID** 1  
**sessionDescription** Audio  
**terminalLabel** M/T for A  
**dataType** Audio Capability  
**mediaChannel** UCA1  
**mediaControlChannel** UCA2

##### **ENTRY 2 – AUDIO & AUDIO CONTROL FOR NODE B**

**sessionID** 2  
**sessionDescription** Audio  
**terminalLabel** M/T for B  
**dataType** Audio Capability  
**mediaChannel** UCA3  
**mediaControlChannel** UCA4

##### **ENTRY 3 – AUDIO & AUDIO CONTROL FOR NODE C**

**sessionID** 3  
**sessionDescription** Audio  
**terminalLabel** M/T for C  
**dataType** Audio Capability  
**mediaChannel** UCA5  
**mediaControlChannel** UCA6

##### **ENTRY 4 – VIDEO & VIDEO CONTROL FOR NODE A**

**sessionID** 4  
**associatedSessionID** 1  
**terminalLabel** M/T for A  
**sessionDescription** Video for Node A  
**dataType** Video Capability  
**mediaChannel** MCA1  
**mediaControlChannel** MCA2

#### **ENTRY 5 – VIDEO & VIDEO CONTROL FOR NODE B**

<b>sessionID</b>	<b>5</b>
<b>associatedSessionID</b>	<b>2</b>
<b>terminalLabel</b>	<b>M/T for B</b>
<b>sessionDescription</b>	<b>Video for Node B</b>
<b>dataType</b>	<b>Video Capability</b>
<b>mediaChannel</b>	<b>MCA3</b>
<b>mediaControlChannel</b>	<b>MCA4</b>

#### **ENTRY 6 – VIDEO & VIDEO CONTROL FOR NODE C**

<b>sessionID</b>	<b>6</b>
<b>associatedSessionID</b>	<b>3</b>
<b>terminalLabel</b>	<b>M/T for C</b>
<b>sessionDescription</b>	<b>Video for Node C</b>
<b>dataType</b>	<b>Video Capability</b>
<b>mediaChannel</b>	<b>MCA5</b>
<b>mediaControlChannel</b>	<b>MCA6</b>

## **APPENDIX II**

### **Transport Level Resource Reservation Procedures**

#### **II.1 Introduction**

This Recommendation recommends the use of transport level resource reservation mechanisms to fulfil the QOS requirements of real-time video and audio streams. Although the transport level resource reservation mechanisms themselves are beyond the scope of this Recommendation, the general method and coordination of these transport level mechanisms between H.323 entities is described in this appendix to prevent conflicting interoperability issues.

This appendix describes the use of RSVP (Resource reSerVation Protocol) as a possible mechanism for providing transport level QOS over IP-based networks. Other protocols may be used; however, the basic procedures defined in this appendix should still apply. Participants in a conference should be able to signal their intentions, capabilities, and requirements in a standard, protocol-specific manner. In addition, the signalling sequence of the resource reservation mechanisms must be specified such that the call establishment interval is minimal.

RSVP is the transport level signalling protocol for reserving resources in unreliable IP-based networks. Using RSVP, H.323 endpoints can reserve resources for a given real-time traffic stream based on its QOS requirements. If the network fails to reserve the required resources, or in the absence of RSVP, only best-effort delivery of the packets is possible.

#### **II.2 QOS Support for H.323**

When an endpoint requests admission with a Gatekeeper, it should indicate in the ARQ message whether or not it is capable of reserving resources. The Gatekeeper should then decide, based on the information it receives from the endpoint and on information it has about the state of the network, either:

- to permit the endpoint to apply its own reservation mechanism for its H.323 session; or
- to perform resource reservation on behalf of the endpoint; or
- that no resource reservation is needed at all. Best-effort is sufficient.

This decision is conveyed to the endpoint in the ACF message. The endpoint shall accept the Gatekeeper's decision in order to place a call.

The Gatekeeper should reject an endpoint's ARQ, if the endpoint does not indicate that it is capable of resource reservation, and the Gatekeeper decides that resource reservation must be controlled by the endpoint. In this case, the Gatekeeper should send an ARJ back to the endpoint.

The specific field in H.225 RAS signalling to permit this functionality is the **transportQOS** field.

In addition to **transportQOS**, an endpoint should also calculate and report the bandwidth it currently intends to use in all channels of the call. This bandwidth should be reported in the **bandWidth** field of the ARQ message independent of the decision by the endpoint to use RSVP signalling or not. In addition, if bandwidth requirements change during the course of the call, an endpoint should report changes in bandwidth requirements to the Gatekeeper using BRQ independent of the decision to use RSVP.

RSVP reservations can only be made by network entities which are in the path of media flow between endpoints. It is possible through Gatekeeper routed call signalling to route media streams through a Gatekeeper. However, most of the time media channels will be routed between endpoints without passing through the Gatekeeper. If a Gatekeeper decides to route media streams, then the procedures followed should be identical to those for RSVP signalling directly from the endpoints. It is best if RSVP reservations are made directly by the endpoints since this will reserve resources along the entire routed path of the call. The remainder of this appendix discusses the use of RSVP by the H.323 endpoints.

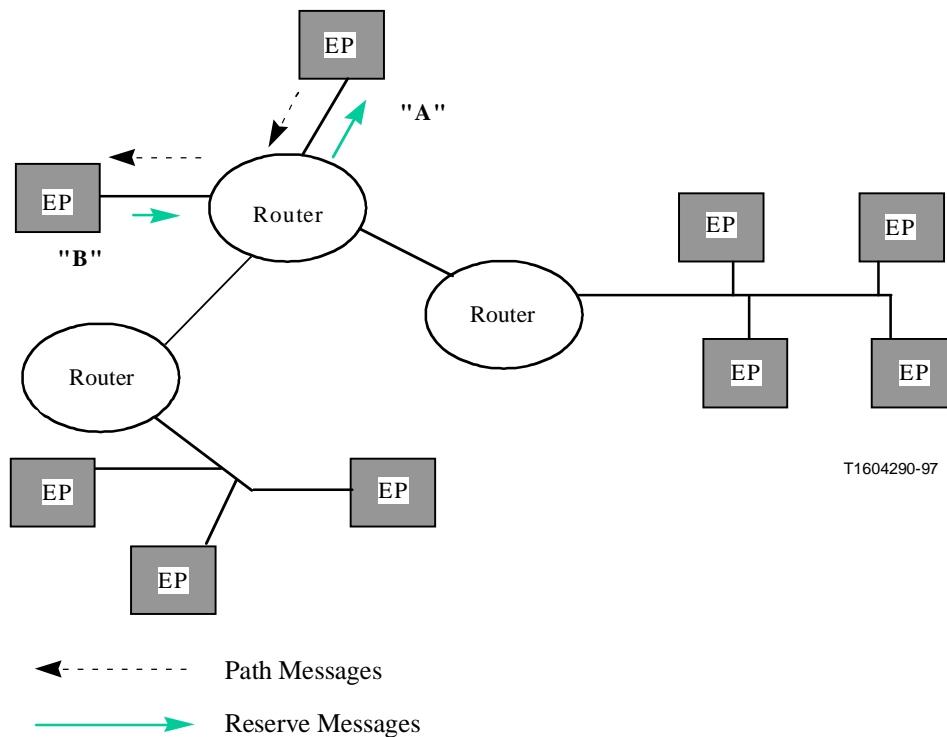
Some of the salient points of RSVP are as follows:

- RSVP supports both unicast and multicast environments;
- RSVP is tied to specific streams (i.e. specific Transport Address pairs);
- RSVP is soft-state based, and therefore adapts dynamically to changing group membership and routes;
- RSVP is unidirectional;
- RSVP is receiver-oriented – the recipient of the media stream makes the reservation (scalable).



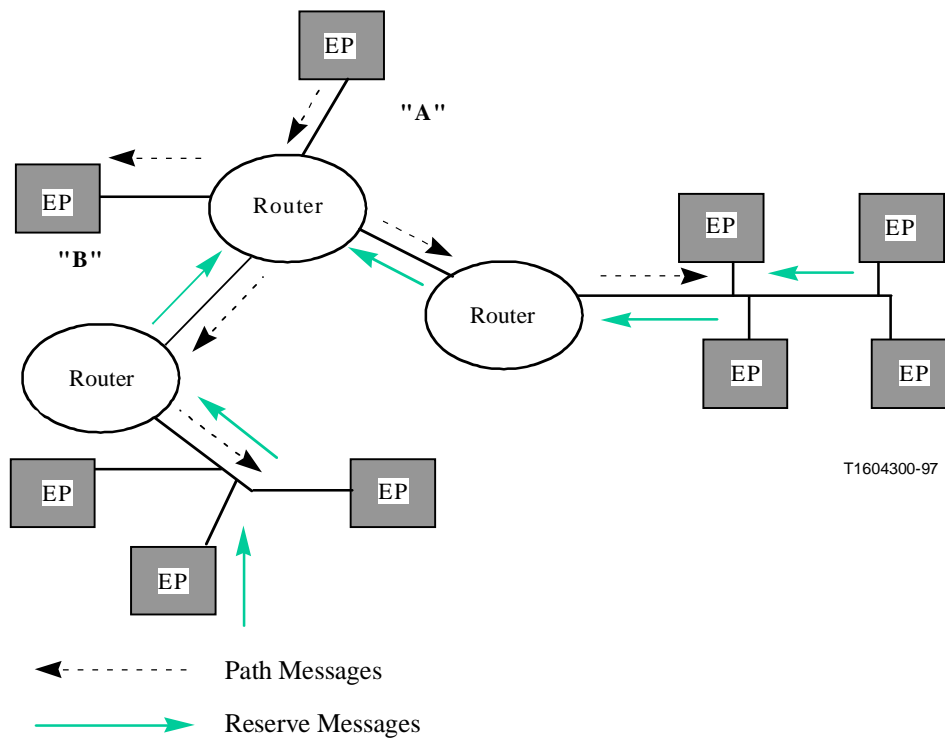
### II.3 RSVP background

In the following description, the high-level usage of RSVP in a simple H.323 conference will be outlined.



**Figure II.1/H.323 – Resource reservation for a point-to-point connection**

In Figure II.1 above, Endpoint A wishes to send a media stream to Endpoint B. Therefore, it has to open a logical channel to B. RSVP signalling for resource reservation should be a part of the opening logical channel procedure. Endpoint A would cause RSVP *Path* messages to be sent out to B. These *Path* messages go through routers and leave "state" on their way tracing towards B. *Path* messages contain the complete source and destination addresses of the stream and a characterization of the traffic that the source will send. Endpoint B would use the information from the *Path* to make the RSVP *Resv* request for the full length of the path. *Resv* messages contain the actual reservation and will generally be the same as the traffic specification in the *Path* message.



**Figure II.2/H.323 – Resource reservation for a point-to-multipoint connection**

In Figure II.2 above, a multipoint conference is shown. The *Path* messages are utilized in the same manner as the simpler point-to-point case. It should be noted that the *Resv* requests are aggregated by the routers to keep redundant reservation requests from occurring upstream.

*Path* messages must contain the complete destination/source addresses and a traffic specification. *Resv* messages contain the reservation parameters and the required service. *Path* and *Resv* messages for a given traffic stream should be sent as part of the **openLogicalChannel** procedure for that particular stream. The reservation should be released during the **closeLogicalChannel** procedure using the RSVP *PathTear* and *ResvTear* messages.

Note that RSVP *Path* and *Resv* messages use the same IP address/port pair as the media to be delivered between endpoints. This means that these messages must be filtered out of the media stream by the endpoints. This is not an issue for endpoints which do UDP filtering since RSVP messages themselves are not UDP messages. Even so, the sender of a media stream should not use RSVP when the receiver is not capable of it. RSVP capabilities are exchanged as part of the capability exchange and open logical channel procedures.

RSVP is only a signalling protocol. Together with the appropriate QOS services (e.g. guaranteed QOS or controlled-load service), scheduling mechanisms (e.g. weighted fair queuing), and policy-based admission control module (e.g. local policy manager), RSVP is capable of satisfying the QOS requirements of H.323 conference participants. In addition, RSVP is designed for point-to-point links. If a path traverses a shared link, RSVP invokes the appropriate resource reservation mechanism for the specific shared medium, e.g. SBM (Subnet Bandwidth Management) in case of Ethernet. All the mechanisms mentioned in this paragraph are controlled completely from within RSVP. Therefore, all that an H.323 endpoint needs is RSVP signalling.

#### II.4 The H.245 capability exchange phase

During the H.245 capability exchange phase, each endpoint indicates its transmit and receive capabilities to the other endpoint. The **qOSCapability** is part of the capability exchange. However, it is not stream-specific. Therefore, the RSVP parameters if specified in the **qOSCapability** would

represent an aggregate for all streams (either those to be transmitted or those to be received). Such parameters will not be of any use to the other endpoint. Therefore, the only RSVP-related information an endpoint should convey to the other endpoint in the capability set is whether or not it is RSVP-capable.

To signal RSVP capability, an endpoint should set the appropriate available **qOSMode** fields within the capability PDU during capability exchange. Endpoints which do not receive RSVP capabilities from the receiving endpoint should not use RSVP when opening logical channels.

## II.5 Open logical channel and setting up reservations

In this clause, we describe the steps that should be followed for opening an H.245 logical channel and reserving resources for a given traffic stream. Reservations are established only if both endpoints indicate that they are RSVP enabled during capability exchange. We consider only the point-to-point case. The case of point-to-multipoint (multicast) connections will be discussed in II.7.

The sender should specify the RSVP parameters of the stream to be transmitted, and the integrated services the sender supports, in the **qOSCapability** field of the **openLogicalChannel** message. In case of a point-to-point stream, the sender does not specify a receiver port ID in the **openLogicalChannel** message. This ID is selected by the receiver after receiving the **openLogicalChannel**, and is returned to the sender in the **openLogicalChannelAck** message. Only then can the sender create an RSVP session for that stream (to create an RSVP session for a given stream means that the endpoint registers with RSVP to get notified when messages arrive that may affect the state of the RSVP reservation for that stream), and start emitting RSVP *Path* messages. The receiver has sufficient information to create an RSVP session for the same stream before sending the **openLogicalChannelAck** message. The information needed to create an RSVP session and initiate RSVP processing are: the receiver IP address in case of point-to-point or the group multicast IP address in case of point-to-multipoint, the receiver port ID, and the protocol (always UDP in case of H.323 audio and video streams on IP networks).

A receiver may not want to start receiving stream packets until the RSVP reservations are in place. To achieve this, the receiver may set the Boolean **flowcontrolToZero** field of the **openLogicalChannelAck** message to TRUE to indicate that it does not wish to receive any traffic on that channel before the resource reservations are complete. When a sender receives an **openLogicalChannelAck** message with **flowControlToZero** set to TRUE, the sender shall not transmit any traffic on that channel.

When the receiver starts receiving the sender's *Path* messages, it should start sending RSVP *Resv* messages. When the receiver receives an RSVP *ResvConf* message confirming that reservations have been established, it may send a **flowControlCommand** to the sender unrestricting the bit rate of the traffic stream, i.e. cancelling the effect of the previous **flowcontrolToZero** field in the **openLogicalChannelAck** message. When the sender receives the **flowControlCommand** it starts transmitting packets.

Note that the *ResvConf* message, and similarly all other RSVP messages, are transmitted unreliably. As a result, they may get delayed or even lost. An endpoint should be aware of that fact, and set timers with appropriate value while waiting for a *ResvConf*. The action taken if the endpoint times out without receiving a *ResvConf* is up to the individual endpoint vendors.

The behaviour of an endpoint if RSVP reservations fail at any point during an H.323 call is not specified in this appendix, and is left to the individual vendors. However, if an RSVP reservation fails and the receiving endpoint decides that best-effort level of service is not acceptable, it may request to close its logical channel using the **requestChannelClose** message. The **closeReason** field is available in the **requestChannelClose** message to allow the receiver to signal to the sender that the RSVP reservation has failed. Along with the failure indication, **requestChannelClose** includes **qOSCapability** which can be used by the receiver to tell the sender the resources which are actually

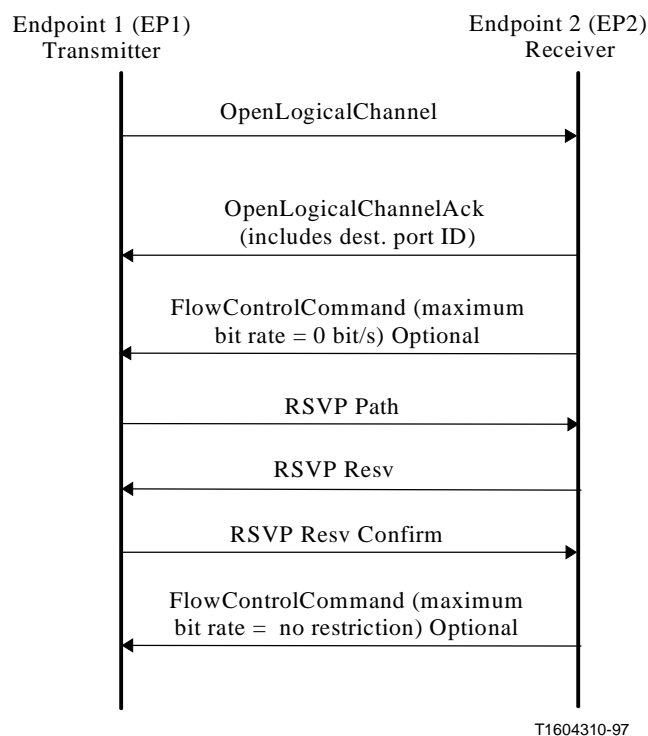
currently available on the path from the sender to the receiver. At this point, the sender can decide to try to re-open the channel with a lower bandwidth codec and/or data format and go through the Open Logical Channel procedure again.

All RSVP *Resv* requests shall use the same reservation style, the **Fixed Filter** style, for the following reasons:

- Shared filter styles reduce to fixed filters in case of point-to-point calls.
- Different reservation styles for the same session cannot be merged in the network. For example, if in a multipoint call some of the receivers request fixed filter reservations while the rest request shared explicit reservations, then either the fixed filter reservations or the shared explicit reservations will fail.
- Shared reservations, created by wildcard filter and shared explicit filter styles, are appropriate for those multicast applications in which multiple data sources are unlikely to transmit simultaneously. In distributed multipoint H.323 calls, there is no mechanism to permit only one source to transmit at a specific time. On the other hand, in centralized multipoint H.323 calls, the MCU is the only multicast source. Shared reservation styles are not suited for either case.

It is up to the endpoint vendors to choose which intserv QOS service (guaranteed QOS or controlled-load) to use. However, any RSVP-enabled H.323 endpoint shall support the controlled-load service as a least common service. This requirement is necessary to avoid interoperability problems that may arise from RSVP-enabled H.323 endpoints which do not support a common intserv QOS service.

Figure II.3 shows the sequence of messages in case of successful RSVP reservation.



**Figure II.3/H.323 – Message sequence for opening a unicast logical channel with RSVP**

## II.6 Close logical channel and tearing down reservations

Before sending out a **closeLogicalChannel** message for a given traffic stream, a sending endpoint should send a *PathTear* message if an RSVP session has been previously created for that stream. When a receiving endpoint receives a **closeLogicalChannel** for a given traffic stream, it should send a *ResvTear* message if an RSVP session has been previously created for that stream.

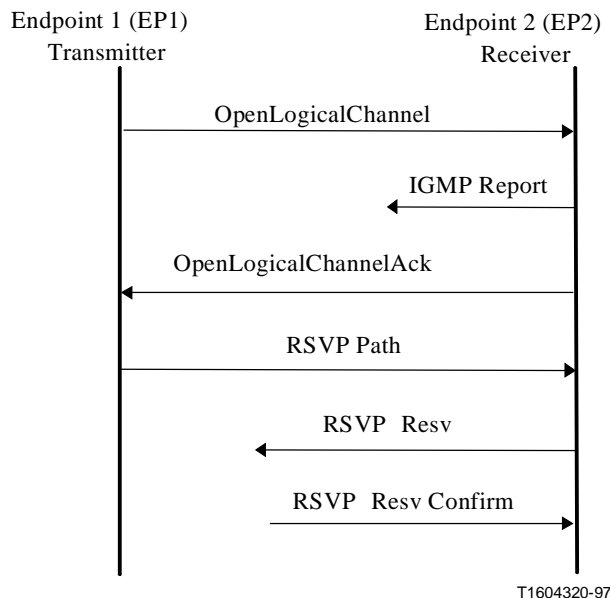
## II.7 Resource reservation for multicast H.323 logical channels

The H.245 **openLogicalChannel** procedure is point-to-point even if the traffic stream involved is a multicast stream. However for the receiving endpoint to start receiving packets of a multicast stream, it has to join the multicast group and get connected to the source's multicast tree. When a receiver receives an **openLogicalChannel** message, it joins the multicast group and the source's multicast tree using standard IGMP procedures. The IGMP join (using IGMP *Report* message) takes place before the receiver sends an **openLogicalChannelAck** back to the sender.

In case of a multicast stream, the sender specifies the receiver port ID in the **openLogicalChannel** message instead of receiving the receiver port ID in the **openLogicalChannelAck** message.

The receiver may set the **flowControlToZero** field of the **openLogicalChannelAck** message to TRUE, similar to the unicast case. However, the sender (an endpoint in a distributed conference or an MCU in centralized conference) should decide not to interrupt the data stream on the opened channel, if it determines this interruption may affect other receivers of the same multicast group which are already receiving that stream. As a result, in the multicast case, the receiver may initially receive the data at best-effort until the RSVP reservations are established.

Figure II.4 shows the sequence of messages required to open a logical channel and to join the multicast tree and to reserve resources for a multicast stream.



**Figure II.4/H.323 – Message sequence for opening a multicast logical channel with RSVP**

Before sending out a **closeLogicalChannel** message for a given multicast stream, a sending endpoint should send an RSVP *PathTear* message if the logical channel being closed is the last channel carrying that multicast stream and if an RSVP session has been previously created for that stream. When a receiving endpoint receives a **closeLogicalChannel** for a given multicast stream, it should send an RSVP *ResvTear* message and an IGMP *Leave* message, if an RSVP session has been previously created for that stream.

## APPENDIX III

### Gatekeeper Based User Location

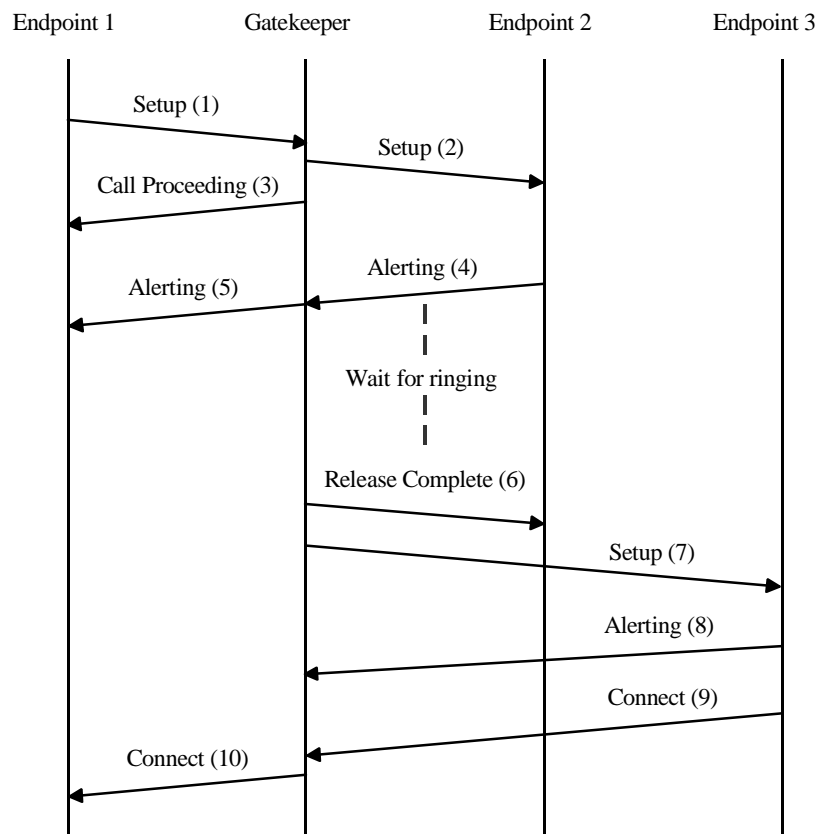
#### III.1 Introduction

This appendix gives examples of how a Gatekeeper/proxy can implement user location services. These services depend on the Gatekeeper using the Gatekeeper routed call signalling model.

#### III.2 Signalling

In the scenario shown in Figure III.1, the Gatekeeper implements a "divert on no reply" service. Endpoint 1 calls Endpoint 2 with the Call Signalling Channel routed through the Gatekeeper. If there is no answer after some timeout, the Gatekeeper diverts the call to an alternate endpoint. Messages (1) to (5) show the Gatekeeper attempting to establish a call between Endpoint 1 and Endpoint 2. In this example, Endpoint 2 does not answer and so the Gatekeeper clears the call to Endpoint 2 by sending Release Complete (6). The Gatekeeper then tries Endpoint 3 by sending Setup (7). When Endpoint 3 answers the call using Connect (9), the Gatekeeper forwards the Connect (10) back to Endpoint 1.

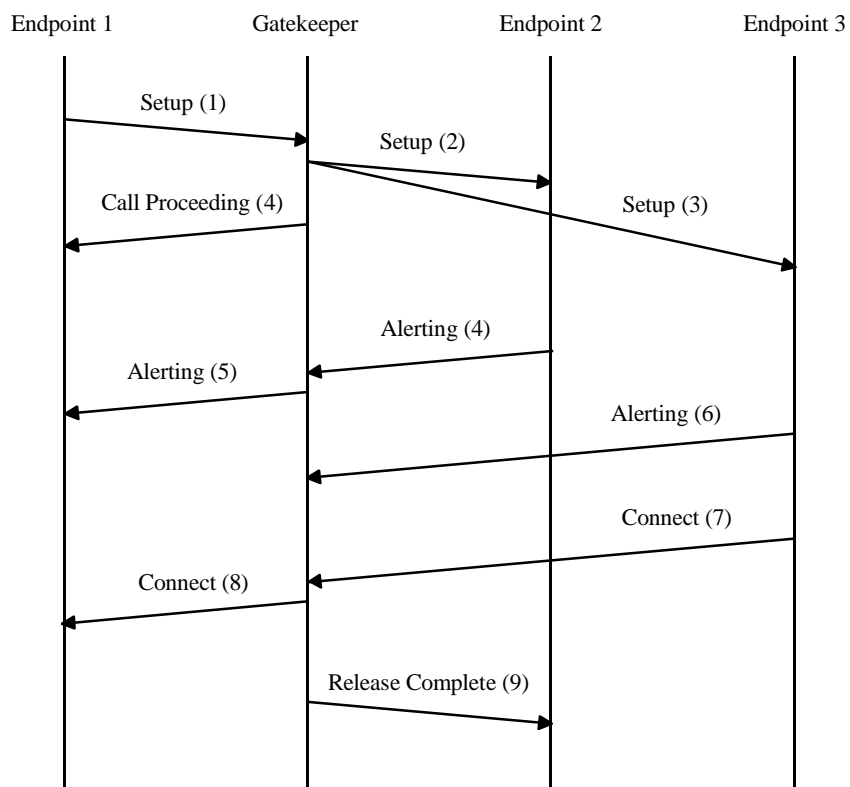
A similar approach can be used to provide "divert on busy" service. In this case, Endpoint 2 would return a Release Complete indicating that is busy. The Gatekeeper would then attempt to establish a call to Endpoint 3.



T1604330-97

**Figure III.1/H.323 – Example of user location using H.225/Q.931 signalling (RAS signalling not shown for clarity)**

In the scenario shown in Figure III.2, the Gatekeeper attempts to establish contact with Endpoints 2 and 3 simultaneously by sending Setups (2) and (3). In this example the user at Endpoint 3 answers by sending Connect (7). The Gatekeeper forwards the Connect (8) back to Endpoint 1 and clears the call attempt to Endpoint 2 using Release Complete (9). The Gatekeeper should ignore any Connect message received from Endpoint 2 which arrives after the Connect (8) message from Endpoint 3 so that only one call is completed.



T1604340-97

**Figure III.2/H.323 – Example of user location using H.225/Q.931 signalling (RAS signalling not shown for clarity)**

Note that if the Gatekeeper is performing this type of user location algorithm, it should not pass the **h245Address** field in any of the Setup Acknowledge, Call Proceeding, and Alerting messages from Endpoint 2 or Endpoint 3 to Endpoint 1 as this may give the wrong result.

## APPENDIX IV

### Signalling Prioritized Alternative Logical Channels in H.245

#### IV.1 Introduction

This appendix describes a simple method by which alternative logical channels may be signalled. No coding or semantic changes are required.

This method depends upon the guaranteed ordered delivery that is provided by TCP and is consequently equally applicable to both tunnelled and non-tunnelled H.245 signalling. Tunnelled signalling further depends upon guaranteed processing order where multiple H.245 messages are tunnelled in a single H.225.0 call signalling message.

#### IV.2 Signalling

All alternative logical channels are identified by the use of a common **forwardLogicalChannelNumber** in **openLogicalChannel** messages, one alternative per message. Messages may be sent either via the H.245 tunnel (one or more OLC messages per call signalling message) or via separate H.245 connection. Alternative logical channels are signalled in order of



decreasing desirability, i.e. the first OLC message specifies the **dataType** that the sender of the OLC would prefer to use on the logical channel.

The receiver of these OLC messages is not required to be aware that this method of alternative propositions is being used. Prior to reception of an acceptable OLC request, it will reject unacceptable OLC requests, typically with a cause code of **dataTypeNotSupported**, **dataTypeNotAvailable**, or **unknownDataType**. When an acceptable OLC request is received, the endpoint will respond with an **openLogicalChannelAck** message. Any subsequently received alternative OLCs are rejected by the receiver with a cause code of **unspecified**, as the requested logical channel number will map to a currently open channel.

The sender of such a prioritized sequence of **openLogicalChannel** messages must keep track of the number of OLC reject messages received prior to reception of an **openLogicalChannelAck** message in order to determine which proposed alternative was accepted by the peer.



## ITU-T RECOMMENDATIONS SERIES

- Series A Organization of the work of the ITU-T
- Series B Means of expression: definitions, symbols, classification
- Series C General telecommunication statistics
- Series D General tariff principles
- Series E Overall network operation, telephone service, service operation and human factors
- Series F Non-telephone telecommunication services
- Series G Transmission systems and media, digital systems and networks
- Series H Audiovisual and multimedia systems**
- Series I Integrated services digital network
- Series J Transmission of television, sound programme and other multimedia signals
- Series K Protection against interference
- Series L Construction, installation and protection of cables and other elements of outside plant
- Series M TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
- Series N Maintenance: international sound programme and television transmission circuits
- Series O Specifications of measuring equipment
- Series P Telephone transmission quality, telephone installations, local line networks
- Series Q Switching and signalling
- Series R Telegraph transmission
- Series S Telegraph services terminal equipment
- Series T Terminals for telematic services
- Series U Telegraph switching
- Series V Data communication over the telephone network
- Series X Data networks and open system communications
- Series Y Global information infrastructure and Internet protocol aspects
- Series Z Languages and general software aspects for telecommunication systems