

International Telecommunication Union

**ITU-T**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

**H.245**

(05/2011)

**SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS**  
Infrastructure of audiovisual services – Communication  
procedures

---

## **Control protocol for multimedia communication**

Recommendation ITU-T H.245



International  
Telecommunication  
Union

**Documents**  
Document Sharing Hub

ITU-T H-SERIES RECOMMENDATIONS  
AUDIOVISUAL AND MULTIMEDIA SYSTEMS

|                                                                               |                    |
|-------------------------------------------------------------------------------|--------------------|
| 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        |
| <b>Communication procedures</b>                                               | <b>H.240–H.259</b> |
| Coding of moving video                                                        | H.260–H.279        |
| Related systems aspects                                                       | H.280–H.299        |
| Systems and terminal equipment for audiovisual services                       | H.300–H.349        |
| Directory services architecture for audiovisual and multimedia services       | H.350–H.359        |
| Quality of service architecture for audiovisual and multimedia services       | H.360–H.369        |
| Supplementary services for multimedia                                         | H.450–H.499        |
| MOBILITY AND COLLABORATION PROCEDURES                                         |                    |
| Overview of Mobility and Collaboration, definitions, protocols and procedures | H.500–H.509        |
| Mobility for H-Series multimedia systems and services                         | H.510–H.519        |
| Mobile multimedia collaboration applications and services                     | H.520–H.529        |
| Security for mobile multimedia systems and services                           | H.530–H.539        |
| Security for mobile multimedia collaboration applications and services        | H.540–H.549        |
| Mobility interworking procedures                                              | H.550–H.559        |
| Mobile multimedia collaboration inter-working procedures                      | H.560–H.569        |
| BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES                       |                    |
| Broadband multimedia services over VDSL                                       | H.610–H.619        |
| Advanced multimedia services and applications                                 | H.620–H.629        |
| IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV                            |                    |
| General aspects                                                               | H.700–H.719        |
| IPTV terminal devices                                                         | H.720–H.729        |
| IPTV middleware                                                               | H.730–H.739        |
| IPTV application event handling                                               | H.740–H.749        |
| IPTV metadata                                                                 | H.750–H.759        |
| IPTV multimedia application frameworks                                        | H.760–H.769        |
| IPTV service discovery up to consumption                                      | H.770–H.779        |

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

# Recommendation ITU-T H.245

## Control protocol for multimedia communication

### Summary

Recommendation ITU-T H.245 specifies syntax and semantics of terminal information messages as well as procedures to use them for in-band negotiation at the start of or during communication. The messages cover receiving and transmitting capabilities as well as mode preference from the receiving end, logical channel signalling, and Control & Indication. Acknowledged signalling procedures are specified to ensure reliable audiovisual and data communication.

Products claiming compliance with Version 16 of ITU-T H.245 shall comply with all of the mandatory requirements of this Recommendation. Version 16 products can be identified by ITU-T H.245 TerminalCapabilitySet messages containing a protocolIdentifier value of {ITU-T (0) recommendation (0) h (8) 245 version (0) 16}.

Relative to ITU-T H.245 Version 15 (2009-12), this version improves harmonization with SIP by making the following changes:

- allowing a new dynamic payload number to be specified in the ITU-T H.245 OpenLogicalChannelAck message;
- revising clause B.3.2 to indicate that ITU-T H.323 systems should use the dynamic payload type specified in the ITU-T H.245 OpenLogicalChannelAck message instead of the dynamic payload type specified in the ITU-T H.245 open logical channel message;
- defining a similar mechanism for logical channels that use multiple payload types; and
- defining a new capability in a new annex, Annex T, that specifies that an endpoint is capable of using the dynamic payload type replacement mechanisms in the ITU-T H.245 OpenLogicalChannelAck message.

### History

| Edition | Recommendation  | Approval   | Study Group |
|---------|-----------------|------------|-------------|
| 1.0     | ITU-T H.245     | 1996-03-20 | 15          |
| 2.0     | ITU-T H.245     | 1997-07-11 | 16          |
| 3.0     | ITU-T H.245     | 1998-02-06 | 16          |
| 4.0     | ITU-T H.245     | 1998-09-25 | 16          |
| 5.0     | ITU-T H.245     | 1999-05-27 | 16          |
| 6.0     | ITU-T H.245     | 2000-02-17 | 16          |
| 7.0     | ITU-T H.245     | 2000-11-17 | 16          |
| 8.0     | ITU-T H.245     | 2001-07-29 | 16          |
| 9.0     | ITU-T H.245     | 2003-02-06 | 16          |
| 10.0    | ITU-T H.245     | 2003-07-14 | 16          |
| 11.0    | ITU-T H.245     | 2005-01-08 | 16          |
| 12.0    | ITU-T H.245     | 2005-10-07 | 16          |
| 13.0    | ITU-T H.245     | 2006-05-29 | 16          |
| 14.0    | ITU-T H.245     | 2008-06-13 | 16          |
| 15.0    | ITU-T H.245 v15 | 2009-12-14 | 16          |
| 16.0    | ITU-T H.245 v16 | 2011-05-14 | 16          |

## FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. 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 Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 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 expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

## INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. 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, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2012

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

## Table of Contents

|                                                                | <b>Page</b> |
|----------------------------------------------------------------|-------------|
| 1 Scope .....                                                  | 1           |
| 2 References.....                                              | 1           |
| 3 Definitions .....                                            | 6           |
| 4 Abbreviations.....                                           | 8           |
| 5 General.....                                                 | 9           |
| 5.1 Master-slave determination .....                           | 9           |
| 5.2 Capability exchange .....                                  | 9           |
| 5.3 Logical channel signalling procedures .....                | 10          |
| 5.4 Receive terminal close logical channel request.....        | 11          |
| 5.5 ITU-T H.223 multiplex table entry modification .....       | 11          |
| 5.6 Audiovisual and data mode request.....                     | 11          |
| 5.7 Round-trip delay determination.....                        | 11          |
| 5.8 Maintenance loops.....                                     | 11          |
| 5.9 Commands and indications.....                              | 11          |
| Annex A – Messages: Syntax .....                               | 12          |
| Annex B – Messages: Semantic definitions.....                  | 80          |
| B.1 Master-Slave Determination messages.....                   | 80          |
| B.2 Terminal capability messages.....                          | 81          |
| B.3 Logical channel signalling messages.....                   | 112         |
| B.4 Multiplex Table signalling messages .....                  | 122         |
| B.5 Request Multiplex Table signalling messages .....          | 123         |
| B.6 Request Mode messages.....                                 | 124         |
| B.7 Round-trip Delay messages.....                             | 128         |
| B.8 Maintenance Loop messages.....                             | 128         |
| B.9 Communication Mode Messages .....                          | 129         |
| B.10 Conference Request and Response Messages .....            | 130         |
| B.11 Multilink Messages .....                                  | 133         |
| B.12 Logical Channel Bit rate Change Messages.....             | 135         |
| B.13 Commands.....                                             | 136         |
| B.14 Indications .....                                         | 142         |
| B.15 Generic messages .....                                    | 149         |
| Annex C – Procedures .....                                     | 150         |
| C.1 Introduction .....                                         | 150         |
| C.2 Master-slave determination procedures.....                 | 153         |
| C.3 Capability exchange procedures.....                        | 163         |
| C.4 Unidirectional Logical Channel signalling procedures ..... | 171         |
| C.5 Bidirectional Logical Channel signalling procedures .....  | 184         |

|                                                                                                   | <b>Page</b> |
|---------------------------------------------------------------------------------------------------|-------------|
| C.6 Close Logical Channel procedures .....                                                        | 199         |
| C.7 ITU-T H.223 Multiplex Table Procedures .....                                                  | 205         |
| C.8 Request Multiplex Entry procedures .....                                                      | 216         |
| C.9 Mode Request procedures .....                                                                 | 223         |
| C.10 Round-trip delay procedures .....                                                            | 232         |
| C.11 Maintenance Loop procedures .....                                                            | 236         |
| Annex D – Object identifier assignments .....                                                     | 246         |
| Annex E – ISO/IEC 14496-2 Capability Definitions.....                                             | 250         |
| Annex F – Logical Channel Bit-Rate Management Capability Definitions .....                        | 253         |
| Annex G – ISO/IEC 14496-1 Capability Definitions .....                                            | 255         |
| G.1 Capability Identifier.....                                                                    | 255         |
| G.2 Capability parameters used for capability negotiations and logical<br>channel signalling..... | 255         |
| G.3 Capability parameters used for logical channel signalling only.....                           | 257         |
| Annex H – ISO/IEC 14496-3 Capability Definitions .....                                            | 258         |
| Annex I – GSM Adaptive Multi-Rate Capability Definitions .....                                    | 262         |
| I.1 Definition of mode signalling and bit stuffing to achieve octet alignment ....                | 264         |
| Annex J – TDMA ACELP Voice Codec Definitions .....                                                | 272         |
| Annex K – TDMA US1 Voice Codec Definitions .....                                                  | 274         |
| Annex L – CDMA EVRC Voice Codec Definitions .....                                                 | 276         |
| Annex M – ISO/IEC 13818-7 and Rec. ITU-R BS.1196-2 Definitions .....                              | 278         |
| Annex N – IETF RFC 3389 – RTP Payload for Comfort Noise .....                                     | 279         |
| Annex O – L16 Capability Identifier .....                                                         | 280         |
| Annex P – Bounded audio stream capability .....                                                   | 281         |
| Annex Q – Generic Capability for $N \times 64K$ Circuit Relay over IP .....                       | 282         |
| Q.1 Introduction .....                                                                            | 282         |
| Q.2 Description .....                                                                             | 282         |
| Annex R – Adaptive Multi-Rate Capability Definitions .....                                        | 286         |
| R.1 Introduction .....                                                                            | 286         |
| R.2 Description .....                                                                             | 286         |
| Annex S – Internet Low Bit Rate Codec (iLBC) Capability Definitions .....                         | 291         |
| S.1 Introduction .....                                                                            | 291         |
| S.2 Description .....                                                                             | 291         |
| Annex T – DynamicPayloadType Replacement Capability Definitions .....                             | 293         |

|                                                                                                                                                | <b>Page</b> |
|------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| Appendix I – Overview of ASN.1 syntax .....                                                                                                    | 294         |
| I.1    Introduction to ASN.1 .....                                                                                                             | 294         |
| I.2    Basic ASN.1 data types .....                                                                                                            | 294         |
| I.3    Aggregate data types .....                                                                                                              | 296         |
| I.4    Object identifier type .....                                                                                                            | 297         |
| Appendix II – Examples of ITU-T H.245 procedures .....                                                                                         | 298         |
| II.1    Introduction .....                                                                                                                     | 298         |
| II.2    Master-slave Determination Signalling Entity .....                                                                                     | 298         |
| II.3    Capability Exchange Signalling Entity .....                                                                                            | 302         |
| II.4    Logical Channel Signalling Entity .....                                                                                                | 303         |
| II.5    Close Logical Channel Signalling Entity .....                                                                                          | 305         |
| II.6    Multiplex Table Signalling Entity .....                                                                                                | 306         |
| II.7    Mode Request Signalling Entity .....                                                                                                   | 308         |
| II.8    Round-trip Delay Signalling Entity .....                                                                                               | 310         |
| II.9    Bidirectional Logical Channel Signalling Entity .....                                                                                  | 311         |
| Appendix III – Summary of procedure timers and counters .....                                                                                  | 314         |
| III.1    Timers .....                                                                                                                          | 314         |
| III.2    Counters .....                                                                                                                        | 315         |
| Appendix IV – ITU-T H.245 extension procedure .....                                                                                            | 316         |
| Appendix V – The replacementFor procedure .....                                                                                                | 318         |
| Appendix VI – Examples of ITU-T H.263 Capability Structure Settings .....                                                                      | 320         |
| VI.1    Examples of Enhancement Layer ITU-T H.245 parameter setting .....                                                                      | 320         |
| VI.2    Examples of Video Back Channel ITU-T H.245 parameter setting .....                                                                     | 321         |
| Appendix VII – Procedure and template for defining new capabilities with ITU-T H.245<br>generic capabilities .....                             | 326         |
| VII.1    Procedure .....                                                                                                                       | 326         |
| VII.2    Template .....                                                                                                                        | 327         |
| VII.3    Example Template – ITU-T H.261 .....                                                                                                  | 328         |
| Appendix VIII – List of generic capabilities and generic messages defined in<br>Recommendations/Standards other than this Recommendation ..... | 330         |
| Appendix IX – ASN.1 usage in this Recommendation .....                                                                                         | 334         |
| IX.1    Tagging .....                                                                                                                          | 334         |
| IX.2    Types .....                                                                                                                            | 334         |
| IX.3    Constraints and Ranges .....                                                                                                           | 334         |
| IX.4    Extensibility .....                                                                                                                    | 334         |

|                                                                                                       | <b>Page</b> |
|-------------------------------------------------------------------------------------------------------|-------------|
| Appendix X – Resolution of unidirectional and bidirectional conflict scenarios .....                  | 335         |
| X.1    Both terminals use bidirectional OLC .....                                                     | 335         |
| X.2    Master proposes bidirectional OLC and slave proposes unidirectional<br>OLC .....               | 335         |
| X.3    Master proposes unidirectional OLC and slave proposes bidirectional<br>OLC .....               | 336         |
| X.4    Master proposes bidirectional OLC with nullData and slave proposes<br>unidirectional OLC ..... | 337         |
| X.5    Both terminals propose bidirectional OLC with nullData.....                                    | 337         |
| Bibliography.....                                                                                     | 338         |

Copyright © 2011 ITU-T. All rights reserved.

# Recommendation ITU-T H.245

## Control protocol for multimedia communication

### 1 Scope

This Recommendation specifies syntax and semantics of terminal information messages as well as procedures to use them for in-band negotiation at the start of or during communication. The messages cover receiving and transmitting capabilities as well as mode preference from the receiving end, logical channel signalling, and Control & Indication. Acknowledged signalling procedures are specified to ensure reliable audiovisual and data communication.

This Recommendation covers a wide range of applications, including storage/retrieval, messaging and distribution services as well as conversational. It applies to, but is not limited to, multimedia systems that use the multiplexes defined in Recs ITU-T H.222.0, ITU-T H.223, and ITU-T H.225.0. These different systems share the same syntax and semantics, and are therefore bit-wise compatible. Some of the procedures are applicable to all systems, while the others are more specific to particular systems.

The different systems that make use of this Recommendation may specify the use of different transport protocols. However, it is intended to be used with a reliable transport layer, that is, one that provides guaranteed delivery of correct data.

NOTE – There should be no confusion with the ITU-T T.120 management system, which is carried within the data stream, and covers different functionalities from those described here – the ITU-T H.245 stream and the ITU-T T.120-data stream are complementary.

### 2 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; 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. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

- [1] Recommendation ITU-T E.164 (2005), *The international public telecommunication numbering plan*.
- [2] Recommendation ITU-T G.711 (1988), *Pulse code modulation (PCM) of voice frequencies*.
- [3] Recommendation ITU-T G.722 (1988), *7 kHz audio-coding within 64 kbit/s*.
- [4] Recommendation ITU-T G.723.1 (2006), *Dual rate speech coder for multimedia communication transmitting at 5.3 and 6.3 kbit/s*.
- [5] Recommendation ITU-T G.728 (1992), *Coding of speech at 16 kbit/s using low-delay code excited linear prediction*.
- [6] Recommendation ITU-T G.729 (1996), *Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)*.
- [7] Recommendation ITU-T H.221 (2004), *Frame structure for a 64 to 1920 kbit/s channel in audiovisual teleservices*.
- [8] Recommendation ITU-T H.222.0 (2000) | ISO/IEC 13818-1:2000, *Information technology – Generic coding of moving pictures and associated audio information: Systems*.

- [9] Recommendation ITU-T H.222.1 (1996), *Multimedia multiplex and synchronization for audiovisual communication in ATM environments.*
- [10] Recommendation ITU-T H.223 (2001), *Multiplexing protocol for low bit rate multimedia communication.*
- [11] Recommendation ITU-T H.224 (2005), *A real time control protocol for simplex applications using the H.221 LSD/HSD/MLP channels.*
- [12] Recommendation ITU-T H.225.0 (2006), *Call signalling protocols and media stream packetization for packet-based multimedia communications systems.*
- [13] Recommendation ITU-T H.230 (2004), *Frame-synchronous control and indication signals for audiovisual systems.*
- [14] Recommendation ITU-T H.233 (2002), *Confidentiality system for audiovisual services.*
- [15] Recommendation ITU-T H.234 (2002), *Encryption and key management and authentication system for audiovisual services.*
- [16] Recommendation ITU-T H.235.0 (2005), *H.323 security: Framework for security in H-series (H.323 and other H.245-based) multimedia systems.*
- [17] Recommendation ITU-T H.243 (2005), *Procedures for establishing communication between three or more audiovisual terminals using digital channels up to 1920 kbit/s.*
- [18] Recommendation ITU-T H.261 (1993), *Video codec for audiovisual services at  $p \times 64$  kbit/s.*
- [19] Recommendation ITU-T H.262 (2000) | ISO/IEC 13818-2:2000, *Information technology – Generic coding of moving pictures and associated audio: Video.*
- [20] Recommendation ITU-T H.263 (2005), *Video coding for low bit rate communication.*
- [21] Recommendation ITU-T H.310 (1998), *Broadband audiovisual communication systems and terminals.*
- [22] Recommendation ITU-T H.320 (2004), *Narrow-band visual telephone systems and terminal equipment.*
- [23] Recommendation ITU-T H.323 (2003), *Packet-based multimedia communications systems.*
- [24] Recommendation ITU-T H.324 (2005), *Terminal for low bit-rate multimedia communication.*
- [25] Recommendation ITU-T I.363.x series, *B-ISDN ATM adaptation layer (AAL) specification.*
- [26] Recommendation ITU-T Q.2931 (1995), *Digital subscriber signalling system No. 2 – User-Network Interface (UNI) layer 3 specification for basic call/connection control.*
- [27] Recommendation ITU-T T.30 (2005), *Procedures for document facsimile transmission in the general switched telephone network.*
- [28] Recommendation ITU-T T.35 (2000), *Procedure for the allocation of ITU-T defined codes for non-standard facilities.*
- [29] Recommendation ITU-T T.38 (2005), *Procedures for real-time Group 3 facsimile communication over IP networks.*
- [30] Recommendation ITU-T T.51 (1992), *Latin based coded character sets for telematic services.*
- [31] Recommendation ITU-T T.84 (1996) | ISO/IEC 10918-3:1997, *Information technology – Digital compression and coding of continuous-tone still images: Extensions.*

- [32] Recommendation ITU-T T.120 (1996), *Data protocols for multimedia conferencing*.
- [33] Recommendation ITU-T T.123 (1999), *Network-specific data protocol stacks for multimedia conferencing*.
- [34] Recommendation ITU-T T.140 (1998), *Protocol for multimedia application text conversation*.
- [35] Recommendation ITU-T T.434 (1999), *Binary file transfer format for the telematic services*.
- [36] Recommendation ITU-T V.14 (1993), *Transmission of start-stop characters over synchronous bearer channels*.
- [37] Recommendation ITU-T V.34 (1998), *A modem operating at data signalling rates of up to 33 600 bit/s for use on the general switched telephone network and on leased point-to-point 2-wire telephone-type circuits*.
- [38] Recommendation ITU-T V.42 (2002), *Error-correcting procedures for DCEs using asynchronous-to-synchronous conversion*.
- [39] Recommendation ITU-T V.140 (2005), *Procedures for establishing communication between two multiprotocol audiovisual terminals using digital channels at a multiple of 64 or 56 kbit/s*.
- [40] Recommendation ITU-T X.680 (2008) | ISO/IEC 8824-1:2008, *Information technology – Abstract Syntax Notation One (ASN.1) – Specification of basic notation*.
- [41] Recommendation ITU-T X.681 (2008) | ISO/IEC 8824-2:2008, *Information technology – Abstract Syntax Notation One (ASN.1) – Information object specification*.
- [42] Recommendation ITU-T X.691 (2008) | ISO/IEC 8825-2:2008, *Information technology – ASN.1 encoding rules – Specification of Packed Encoding Rules (PER)*.
- [43] ISO/IEC 13239:2002, *Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures*.
- [44] ISO/IEC 11172-2:1993, *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s – Part 2: Video*.
- [45] ISO/IEC 11172-3:1993, *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s – Part 3: Audio*.
- [46] ISO/IEC 13818-3:1998, *Information technology – Generic coding of moving pictures and associated audio information – Part 3: Audio*.
- [47] ISO/IEC 13818-6:1998, *Information technology – Generic coding of moving pictures and associated audio information – Part 6: Extensions for DSM-CC*.
- [48] ISO/IEC 14496-1:2004, *Information technology – Coding of audio-visual objects – Part 1: Systems*.
- [49] ISO/IEC 14496-2:2004, *Information technology – Coding of audio-visual objects – Part 2: Visual*.
- [50] ISO/IEC 14496-3:2005, *Information technology – Coding of audio-visual objects – Part 3: Audio*.
- [51] ISO/IEC 14496-3/Amd.1:2003, *Information technology – Coding of audio-visual objects – Part 3: Audio – Amendment 1: Audio extensions*.
- [52] ISO/IEC TR 9577:1999, *Information technology – Protocol identification in the network layer*.

- [53] ETSI ETS 300 961 (GSM 06.10), *Full rate speech transcoding*.
- [54] ETSI ETS 300 969 (GSM 06.20), *Half rate speech transcoding*.
- [55] ETSI ETS 300 726 (GSM 06.60), *Enhanced Full Rate (EFR) speech transcoding*.
- [56] ETSI ETS 300 963 (GSM 06.12), *Comfort noise aspect for full rate speech traffic channels*.
- [57] ETSI ETS 300 971 (GSM 06.22), *Comfort noise aspects for half rate speech traffic channels*.
- [58] ETSI ETS 300 728 (GSM 06.62), *Comfort noise aspects for Enhanced Full Rate (EFR) speech traffic channels*.
- [59] ETSI ETS 300 964 (GSM 06.31), *Discontinuous Transmission (DTX) for full rate speech traffic channels*.
- [60] ETSI ETS 300 972 (GSM 06.41), *Discontinuous transmission (DTX) for half rate speech traffic channels*.
- [61] ETSI ETS 300 729 (GSM 06.81), *Discontinuous Transmission (DTX) for Enhanced Full Rate (EFR) speech traffic channels*.
- [62] ETSI ETS 300 962 (GSM 06.11), *Substitution and muting of lost frames for full rate speech traffic channels*.
- [63] ETSI ETS 300 970 (GSM 06.21), *Substitution and muting of lost frames for half rate speech traffic channels*.
- [64] ETSI ETS 300 727 (GSM 06.61), *Substitution and muting of lost frames for Enhanced Full Rate (EFR) speech traffic channels*.
- [65] ETSI ETS 300 965 (GSM 06.32), *Voice Activity Detector (VAD) for full rate speech traffic channels*.
- [66] ETSI ETS 300 973 (GSM 06.42), *Voice Activity Detector (VAD) for half rate speech traffic channels*.
- [67] ETSI ETS 300 730 (GSM 06.82), *Voice activity detection for enhanced full rate speech traffic channels*.
- [68] ETSI ETS 300 724 (GSM 06.53), *ANSI-C code for the GSM Enhanced Full Rate Speech (EFR) speech codec*.
- [69] ETSI EN 301 712 GSM 06.73, *ANSI-C code for the AMR speech codec*.
- [70] ETSI EN 301 704 GSM 06.90, *Adaptive Multi-Rate (AMR) speech transcoding*.
- [71] ETSI EN 301 705 GSM 06.91, *Substitution and muting of lost frames for Adaptive Multi Rate (AMR) speech traffic channels*.
- [72] ETSI EN 301 706 GSM 06.92, *Comfort noise aspects for Adaptive Multi-Rate (AMR) speech traffic channels*.
- [73] ETSI EN 301 708 GSM 06.94, *Voice Activity Detection (VAD) for Adaptive Multi-Rate (AMR) speech traffic channels*.
- [74] ARIB RCR STD-27H, *Personal Digital Cellular Telecommunication System RCR Standard*.
- [75] TIA/EIA – 136-Rev.A, Part 410, *TDMA Cellular/PCS – Radio Interface, Enhanced Full Rate Voice Codec (ACELP)*. Formerly IS-641. TIA published standard, 1998.
- [76] TIA/EIA/IS 641-A (1998), *TDMA Cellular/PCS – Radio Interface, Enhanced Full-Rate Speech Codec*.

- [77] Recommendation ITU-T H.239 (2005), *Role management and additional media channels for H.300-series terminals*.
- [78] Recommendation ITU-T H.241 (2006), *Extended video procedures and control signals for H.300-series terminals*.
- [79] Recommendation ITU-T H.235.7 (2005), *H.323 security: Usage of the MIKEY key management protocol for the Secure Real Time Transport Protocol (SRTP) within H.235*.
- [80] Recommendation ITU-T H.235.8 (2005), *H.323 security: Key exchange for SRTP using secure signalling channels*.
- [81] Recommendation ITU-T Y.1413 (2004), *TDM-MPLS network interworking – User plane interworking*.
- [82] IETF RFC 2198 (1997), *RTP Payload for Redundant Audio Data*.
- [83] IETF RFC 2733 (1999), *An RTP Payload Format for Forward Error Correction*.
- [84] IETF RFC 3550 (2003), *RTP: A Transport Protocol for Real-Time Applications*.
- [85] IETF RFC 3711 (2004), *The secure Real-time Transport Protocol*.
- [86] IETF RFC 3985 (2005), *Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture*.
- [87] IETF RFC 3267 (2002), *Real-Time Transport Protocol (RTP) Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs*.
- [88] IETF RFC 4733 (2006), *RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals*.
- [89] Recommendation ITU-T H.222 (1988), *Frame structure for 384-1920 kbit/s channels in audiovisual teleservices*.
- [90] Recommendation ITU-T H.235.6 (2009), *H.323 security: Voice encryption profile with native H.235/H.245 key management*.
- [91] Recommendation ITU-T H.226 (1998), *Channel aggregation protocol for multilink operation on circuit-switched networks*.
- [92] Recommendation ITU-T H.361 (2006), *End-to-end quality of service (QoS) and service priority signalling in H.323 systems*.
- [93] Recommendation ITU-T V.76 (1996), *Generic multiplexer using V.42 LAPM-based procedures*.
- [94] Recommendation ITU-T V.75 (1996), *DSVD terminal control procedures*.
- [95] Recommendation ITU-T V.70 (1996), *Procedures for the simultaneous transmission of data and digitally encoded voice signals over the GSTN, or over 2-wire leased point-to-point telephone type circuits*.
- [96] Recommendation ITU-T G.726 (1990), *40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)*.
- [97] Recommendation ITU-T Q.931 (1998), *ISDN user-network interface layer 3 specification for basic call control*.
- [98] Recommendation ITU-T T.39 (1997), *Application profiles for simultaneous voice and facsimile terminals*.
- [99] Recommendation ITU-T T.124 (2007), *Generic Conference Control*.

- [100] Recommendation ITU-T Y.1541 (2006), *Network performance objectives for IP-based services*.
- [101] Recommendation ITU-R BS.1196-2 (2010), *Audio coding for digital broadcasting*.
- [102] ISO/IEC 10646:2011, *Information technology – Universal Coded Character Set (UCS)*.
- [103] ISO/IEC 13818-7:2006, *Information technology – Generic coding of moving pictures and associated audio information – Part 7: Advanced Audio Coding (AAC)*.
- [104] IETF RFC 1890 (1996), *RTP Profile for Audio and Video Conferences with Minimal Control*.
- [105] IETF RFC 3389 (2002), *Real-time Transport Protocol (RTP) Payload for Comfort Noise (CN)*.
- [106] IETF RFC 3951 (2004), *Internet Low Bit Rate Codec (iLBC)*.
- [107] IETF RFC 3952 (2004), *Real-time Transport Protocol (RTP) Payload Format for internet Low Bit Rate Codec (iLBC) Speech*.
- [108] IETF RFC 3389 (2002), *Real-time Transport Protocol (RTP) Payload for Comfort Noise (CN)*.
- [109] TIA/EIA/ IS-127 (1997), *Enhanced Variable Rate Codec, Speech Service Option 3 for Wideband Spread Spectrum Digital Systems*.

### 3 Definitions

This Recommendation defines the following terms:

- 3.1 bidirectional logical channel:** A bidirectional logical channel consists of a pair of associated transmission paths between two terminals, one in each direction of transmission.
- 3.2 capability:** A terminal has a particular capability if it is able to encode and transmit or receive and decode that particular signal.
- 3.3 channel:** A channel is a unidirectional link between two endpoints.
- 3.4 command:** A command is a message that requires action but no explicit response.
- 3.5 elementary stream:** Elementary stream is a generic term for a coded video, coded audio or other coded bitstream.
- 3.6 entry:** The word "entry" is used to refer to elements in sets or tables, such as capability sets and multiplex tables.
- 3.7 forward:** Forward is used to refer to transmission directed from the terminal making the request for a bidirectional logical channel to the other terminal.
- 3.8 in-band:** In-band messages are those that are transported within the channel or logical channel to which they refer.
- 3.9 incoming:** An incoming signalling entity cannot initiate a procedure, but responds to messages from the remote signalling entity and its own user's primitives.
- 3.10 indication:** An indication is a message that contains information but does not require action or response.
- 3.11 logical channel:** A logical channel is a unidirectional path or bidirectional path for the transmission of information.
- 3.12 logical channel number:** A logical channel number is a number that identifies a single logical channel.

- 3.13 logical channel signalling:** Logical channel signalling is a set of procedures that are used to open and close logical channels.
- 3.14 master terminal:** A master terminal is the terminal that is determined as being the master terminal by the master-slave determination procedure defined in this Recommendation, or by some other procedure.
- 3.15 medium type:** A medium type is a single form of information that is presented to a user or the data representing that information: video, audio and text are example medium types.
- 3.16 mode:** A mode is a set of elementary streams that a terminal is transmitting, intends to transmit, or would like to receive.
- 3.17 multimedia communication:** Multimedia communication refers to the transmission and/or reception of signals of two or more medium types simultaneously.
- 3.18 non-standard:** Not conforming to a national or international standard referenced in this Recommendation.
- 3.19 outgoing:** An outgoing signalling entity is one which initiates a procedure.
- 3.20 multipoint:** Multipoint refers to the simultaneous interconnection of three or more terminals to allow communication among several sites through the use of multipoint control units (bridges) that centrally direct the flow of information.
- 3.21 request:** A request is a message that results in action by the remote terminal and requires an immediate response from it.
- 3.22 response:** A response is a message that is the response to a request.
- 3.23 reverse:** "Reverse" is used to refer to transmission directed from the terminal receiving a request for a bidirectional logical channel to the terminal making the request.
- 3.24 session:** A session is a period of communication between two terminals which may be conversational or non-conversational (for example, retrieval from a database).
- 3.25 slave terminal:** A slave terminal is the terminal that is determined as being the slave terminal by the master-slave determination procedure defined in this Recommendation, or by some other procedure.
- 3.26 support:** The ability to operate in a given mode; however, a requirement to support a mode does not mean that the mode must actually be used at all times: unless prohibited, other modes may be used by mutual negotiation.
- 3.27 terminal:** A terminal is any endpoint and may be a user's terminal or some other communication system such as an MCU or an information server.
- 3.28 TSAP identifier:** The piece of information used to multiplex several transport connections of the same type on a single ITU-T H.323 entity with all transport connections sharing the same LAN address, (e.g., the port number in a TCP/UDP/IP environment). TSAP identifiers may be (pre)assigned by some international authority or may be allocated dynamically during 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.
- 3.29 unidirectional logical channel:** A unidirectional logical channel is a path for the transmission of a single elementary stream from one terminal to another.

## 4 Abbreviations

This Recommendation uses the following abbreviations:

|           |                                                                                            |
|-----------|--------------------------------------------------------------------------------------------|
| AAL       | ATM Adaptation Layer                                                                       |
| AL1, 2, 3 | ITU-T H.223 Adaptation Layers 1, 2 and 3                                                   |
| ASN.1     | Abstract Syntax Notation One                                                               |
| ATM       | Asynchronous Transfer Mode                                                                 |
| B-LCSE    | Bidirectional Logical Channel Signalling Entity                                            |
| CESE      | Capability Exchange Signalling Entity                                                      |
| CIF       | Common Intermediate Format (of a video picture: refer to Recs ITU-T H.261 and ITU-T H.263) |
| CLCSE     | Close Logical Channel Signalling Entity                                                    |
| CPCS      | Common Part Convergence Sublayer (of ATM Adaptation Layer 5)                               |
| DSM-CC    | Digital Storage Media/Command and Control                                                  |
| DTMF      | Dual Tone Multi-Frequency                                                                  |
| GOB       | Group of Blocks (of a video picture: refer to Recs ITU-T H.261 and ITU-T H.263)            |
| GSTN      | General Switched Telephone Network                                                         |
| HDLC      | High-level Data Link Control                                                               |
| HRD       | Hypothetical Reference Decoder (refer to Recs ITU-T H.261 and ITU-T H.263)                 |
| IV        | Initialization Vector (used for encryption: refer to Recs ITU-T H.233 and ITU-T H.234)     |
| LAPM      | Link Access Protocol for Modems                                                            |
| LCSE      | Logical Channel Signalling Entity                                                          |
| MC        | ITU-T H.323 Multipoint Control Entity                                                      |
| MCU       | Multipoint Control Unit                                                                    |
| MLSE      | Maintenance Loop Signalling Entity                                                         |
| MPI       | Minimum Picture Interval                                                                   |
| MRSE      | Mode Request Signalling Entity                                                             |
| MSDSE     | Master-Slave Determination Signalling Entity                                               |
| MTSE      | Multiplex Table Signalling Entity                                                          |
| PCR       | Program Clock Reference (refer to Rec. ITU-T H.222.0   ISO/IEC 13818-1)                    |
| PID       | Packet Identifier (refer to Rec. ITU-T H.222.0   ISO/IEC 13818-1)                          |
| QCIF      | Quarter CIF                                                                                |
| RMESE     | Request Multiplex Entry Signalling Entity                                                  |
| RTCP      | Real-time Transport Control Protocol                                                       |
| RTDSE     | Round-Trip Delay Signalling Entity                                                         |
| RTP       | Real-time Transport Protocol                                                               |
| SDL       | Specification and Description Language                                                     |

|       |                                                                                           |
|-------|-------------------------------------------------------------------------------------------|
| SDU   | Service Data Unit                                                                         |
| SE    | Session Exchange Message (used for encryption: refer to Recs ITU-T H.233 and ITU-T H.234) |
| SQCIF | Sub QCIF                                                                                  |
| STD   | System Target Decoder (refer to Rec. ITU-T H.222.0   ISO/IEC 13818-1)                     |
| VC    | ATM Virtual Channel                                                                       |

## 5 General

This Recommendation provides a number of different services, some of which are expected to be applicable to all terminals that use it and some that are more specific to particular ones. Procedures are defined to allow the exchange of audiovisual and data capabilities; to request the transmission of a particular audiovisual and data mode; to manage the logical channels used to transport the audiovisual and data information; to establish which terminal is the master terminal and which is the slave terminal for the purposes of managing logical channels; to carry various control and indication signals; to control the bit rate of individual logical channels and the whole multiplex; and to measure the round-trip delay, from one terminal to the other and back. These procedures are explained in more detail below.

Following this general introduction, there are clauses detailing the message syntax and semantics and the procedures. The syntax has been defined using ASN.1 notation [40] and the semantics define the meaning of syntax elements as well as providing syntactic constraints that are not specified in the ASN.1 syntax. The procedures clause defines the protocols that use the messages defined in the other clauses.

Although not all of the messages and procedures defined in this Recommendation will be applicable to all terminals, no indication of such restrictions is given here. These restrictions are the responsibility of the recommendations that use this Recommendation.

This Recommendation has been defined to be independent of the underlying transport mechanism, but is intended to be used with a reliable transport layer, that is, one that provides guaranteed delivery of correct data.

### 5.1 Master-slave determination

Conflicts may arise when two terminals involved in a call initiate similar events simultaneously and only one such event is possible or desired, for example, when resources are available for only one occurrence of the event. To resolve such conflicts, one terminal shall act as a master and the other terminal shall act as a slave terminal. Rules specify how the master and slave terminal shall respond at times of conflict.

The master-slave determination procedure allows terminals in a call to determine which terminal is the master and which terminal is the slave. The terminal status may be re-determined at any time during a call; however, a terminal may only initiate the master-slave determination process if no procedure which depends upon its result is locally active.

### 5.2 Capability exchange

The capability exchange procedures are intended to ensure that the only multimedia signals to be transmitted are those that can be received and treated appropriately by the receive terminal. This requires that the capabilities of each terminal to receive and decode be known to the other terminal. It is not necessary that a terminal understand or store all incoming capabilities; those that are not understood, or cannot be used shall be ignored, and no fault shall be considered to have occurred.

When a capability is received which contains extensions not understood by the terminal, the capability shall be accepted as if it did not contain the extensions.

The total capability of a terminal to receive and decode various signals is made known to the other terminal by transmission of its capability set.

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).

These capability sets provide for more than one stream of a given medium type to be sent simultaneously. For example, a terminal may declare its ability to receive (or send) two independent ITU-T H.262 video streams and two independent ITU-T G.722 audio streams at the same time. Capability messages have been defined to allow a terminal to indicate that it does not have fixed capabilities, but that they depend on which other modes are being used simultaneously. For example, it is possible to indicate that higher resolution video can be decoded when a simpler audio algorithm is used; or that either two low resolution video sequences can be decoded or a single high resolution one. It is also possible to indicate trade-offs between the capability to transmit and the capability to receive.

Non-standard capabilities and control messages may be issued using the NonStandardParameter structure. 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 reissue capability sets at any time.

### 5.3 Logical channel signalling procedures

An acknowledged protocol is defined for the opening and closing of logical channels which carry the audiovisual and data information. The aim of these procedures is to ensure that a terminal is capable of receiving and decoding the data that will be transmitted on a logical channel at the time the logical channel is opened rather than at the time the first data is transmitted on it; and to ensure that the receive terminal is ready to receive and decode the data that will be transmitted on the logical channel before that transmission starts. The OpenLogicalChannel message includes a description of the data to be transported, for example, H.262 MP@ML at 6 Mbit/s. Logical channels should only be opened when there is sufficient capability to receive data on all open logical channels simultaneously.

A part of this protocol is concerned with the opening of bidirectional channels. To avoid conflicts which may arise when two terminals initiate similar events simultaneously, one terminal is defined as the master terminal, and the other as the slave terminal. A protocol is defined to establish which terminal is the master and which is the slave. However, systems that use this Recommendation may specify the procedure specified in this Recommendation or another means of determining which terminal is the master and which is the slave.

## 5.4 Receive terminal close logical channel request

A logical channel is opened and closed from the transmitter side. A mechanism is defined which allows a receive terminal to request the closure of an incoming logical channel. The transmit terminal may accept or reject the logical channel closure request. A terminal may, for example, use these procedures to request the closure of an incoming logical channel which, for whatever reason, cannot be decoded. These procedures may also be used to request the closure of a bidirectional logical channel by the terminal that did not open the channel.

## 5.5 ITU-T H.223 multiplex table entry modification

The ITU-T H.223 multiplex table associates each octet within an ITU-T H.223 MUX message with a particular logical channel number. The ITU-T H.223 multiplex table may have up to 15 entries. A mechanism is provided that allows the transmit terminal to specify and inform the receiver of new ITU-T H.223 multiplex table entries. A receive terminal may also request the retransmission of a multiplex table entry.

## 5.6 Audiovisual and data mode request

When the capability exchange protocol has been completed, both terminals will be aware of each other's capability to transmit and receive as specified in the capability descriptors that have been exchanged. It is not mandatory for a terminal to declare all its capabilities; it only needs to declare those that it wishes to be used.

A terminal may indicate its capabilities to transmit. A terminal that receives transmission capabilities from the remote terminal may request a particular mode to be transmitted to it. A terminal indicates that it does not want its transmission mode to be controlled by the remote terminal by sending no transmission capabilities.

## 5.7 Round-trip delay determination

It may be useful in some applications to have knowledge of the round-trip delay between a transmit terminal and a receive terminal. A mechanism is provided to measure this round-trip delay. This mechanism may also be useful as a means to detect whether the remote terminal is still functioning.

## 5.8 Maintenance loops

Procedures are specified to establish maintenance loops. It is possible to specify the loop of a single logical channel either as a digital loop or decoded loop, and the loop of the whole multiplex.

## 5.9 Commands and indications

Commands and indications are provided for various purposes: video/audio active/inactive signals to inform the user; fast update request for source switching in multipoint applications are some examples. Neither commands nor indications elicit response messages from the remote terminal. Commands force an action at the remote terminal whilst indications merely provide information and do not force any action.

A command is defined to allow the bit rate of logical channels and the whole multiplex to be controlled from the remote terminal. This has a number of purposes: interworking with terminals using multiplexes in which only a finite number of bit rates are available; multipoint applications where the rates from different sources should be matched; and flow control in congested networks.

## Annex A

### Messages: Syntax

(This annex forms an integral part of this Recommendation.)

This annex specifies the syntax of messages using the notation defined in ASN.1 [40]. Messages shall be encoded for transmission by applying the packed encoding rules specified in [42] using the basic aligned variant. The first bit in each octet which is transmitted is the most significant bit of the octet as is specified in Rec. ITU-T X.691| ISO/IEC 8825-2.

```
MULTIMEDIA-SYSTEM-CONTROL {itu-t(0) recommendation(0) h(8) h245(245) version(0)
16 multimedia-system-control(0)} DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
```

```
-- Export all symbols
```

```
-- =====
-- Top level Messages
-- =====
```

```
MultimediaSystemControlMessage ::= CHOICE
{
    request                RequestMessage,
    response                ResponseMessage,
    command                CommandMessage,
    indication              IndicationMessage,
    ...
}
```

```
-- A RequestMessage results in action and requires an immediate response
```

```
RequestMessage ::= CHOICE
{
    nonStandard                NonStandardMessage,

    masterSlaveDetermination    MasterSlaveDetermination,

    terminalCapabilitySet        TerminalCapabilitySet,

    openLogicalChannel          OpenLogicalChannel,
    closeLogicalChannel          CloseLogicalChannel,

    requestChannelClose         RequestChannelClose,

    multiplexEntrySend          MultiplexEntrySend,

    requestMultiplexEntry        RequestMultiplexEntry,

    requestMode                 RequestMode,

    roundTripDelayRequest        RoundTripDelayRequest,

    maintenanceLoopRequest       MaintenanceLoopRequest,

    ...,
    communicationModeRequest      CommunicationModeRequest,
    conferenceRequest             ConferenceRequest,

    multilinkRequest             MultilinkRequest,
    logicalChannelRateRequest      LogicalChannelRateRequest,
```

```

    genericRequest                GenericMessage
}

-- A ResponseMessage is the response to a RequestMessage

ResponseMessage ::= CHOICE
{
    nonStandard                NonStandardMessage,

    masterSlaveDeterminationAck    MasterSlaveDeterminationAck,
    masterSlaveDeterminationReject MasterSlaveDeterminationReject,

    terminalCapabilitySetAck        TerminalCapabilitySetAck,
    terminalCapabilitySetReject     TerminalCapabilitySetReject,

    openLogicalChannelAck          OpenLogicalChannelAck,
    openLogicalChannelReject        OpenLogicalChannelReject,
    closeLogicalChannelAck          CloseLogicalChannelAck,

    requestChannelCloseAck          RequestChannelCloseAck,
    requestChannelCloseReject       RequestChannelCloseReject,

    multiplexEntrySendAck           MultiplexEntrySendAck,
    multiplexEntrySendReject        MultiplexEntrySendReject,

    requestMultiplexEntryAck        RequestMultiplexEntryAck,
    requestMultiplexEntryReject     RequestMultiplexEntryReject,

    requestModeAck                  RequestModeAck,
    requestModeReject               RequestModeReject,

    roundTripDelayResponse          RoundTripDelayResponse,

    maintenanceLoopAck              MaintenanceLoopAck,
    maintenanceLoopReject           MaintenanceLoopReject,

    ...,
    communicationModeResponse       CommunicationModeResponse,

    conferenceResponse              ConferenceResponse,

    multilinkResponse               MultilinkResponse,

    logicalChannelRateAcknowledge    LogicalChannelRateAcknowledge,
    logicalChannelRateReject         LogicalChannelRateReject,
    genericResponse                  GenericMessage
}

-- A CommandMessage requires action, but no explicit response

CommandMessage ::= CHOICE
{
    nonStandard                NonStandardMessage,

    maintenanceLoopOffCommand    MaintenanceLoopOffCommand,

    sendTerminalCapabilitySet     SendTerminalCapabilitySet,

    encryptionCommand             EncryptionCommand,

    flowControlCommand            FlowControlCommand,

```

|                                       |                                        |
|---------------------------------------|----------------------------------------|
| endSessionCommand                     | EndSessionCommand,                     |
| miscellaneousCommand                  | MiscellaneousCommand,                  |
| ...                                   |                                        |
| communicationModeCommand              | CommunicationModeCommand,              |
| conferenceCommand                     | ConferenceCommand,                     |
| h223MultiplexReconfiguration          | H223MultiplexReconfiguration,          |
| newATMVCCCommand                      | NewATMVCCCommand,                      |
| mobileMultilinkReconfigurationCommand | MobileMultilinkReconfigurationCommand, |
| genericCommand                        | GenericMessage                         |

}

-- An IndicationMessage is information that does not require action or response

IndicationMessage ::= CHOICE

|                                 |                                  |
|---------------------------------|----------------------------------|
| {                               |                                  |
| nonStandard                     | NonStandardMessage,              |
| functionNotUnderstood           | FunctionNotUnderstood,           |
| masterSlaveDeterminationRelease | MasterSlaveDeterminationRelease, |
| terminalCapabilitySetRelease    | TerminalCapabilitySetRelease,    |
| openLogicalChannelConfirm       | OpenLogicalChannelConfirm,       |
| requestChannelCloseRelease      | RequestChannelCloseRelease,      |
| multiplexEntrySendRelease       | MultiplexEntrySendRelease,       |
| requestMultiplexEntryRelease    | RequestMultiplexEntryRelease,    |
| requestModeRelease              | RequestModeRelease,              |
| miscellaneousIndication         | MiscellaneousIndication,         |
| jitterIndication                | JitterIndication,                |
| h223SkewIndication              | H223SkewIndication,              |
| newATMVCIndication              | NewATMVCIndication,              |
| userInput                       | UserInputIndication,             |
| ...                             |                                  |
| h2250MaximumSkewIndication      | H2250MaximumSkewIndication,      |
| mcLocationIndication            | MCLocationIndication,            |
| conferenceIndication            | ConferenceIndication,            |
| vendorIdentification            | VendorIdentification,            |
| functionNotSupported            | FunctionNotSupported,            |
| multilinkIndication             | MultilinkIndication,             |
| logicalChannelRateRelease       | LogicalChannelRateRelease,       |
| flowControlIndication           | FlowControlIndication,           |

```

mobileMultilinkReconfigurationIndication      MobileMultilinkReconfigurationIndication,
genericIndication                             GenericMessage

}

-- SequenceNumber is defined here as it is used in a number of Messages

SequenceNumber ::= INTEGER (0..255)

-- =====
-- Generic Message definitions
-- =====

GenericMessage ::= SEQUENCE
{
    messageIdentifier          CapabilityIdentifier,
    subMessageIdentifier       INTEGER(0..127)  OPTIONAL,

    messageContent             SEQUENCE OF GenericParameter  OPTIONAL,
    ...
}

GenericInformation ::= GenericMessage

-- =====
-- Non-standard Message definitions
-- =====

NonStandardMessage ::= SEQUENCE
{
    nonStandardData            NonStandardParameter,
    ...
}

NonStandardParameter ::= SEQUENCE
{
    nonStandardIdentifier      NonStandardIdentifier,
    data                       OCTET STRING
}

NonStandardIdentifier ::= CHOICE
{
    object                     OBJECT IDENTIFIER,
    h221NonStandard            SEQUENCE
    {
        t35CountryCode        INTEGER (0..255),      -- country, per
                                                    -- Annex A of
                                                    ITU-T T.35

        t35Extension           INTEGER (0..255),
                                -- assigned nationally unless
                                -- t35CountryCode is binary
                                -- 1111 1111, in which case it shall
                                -- contain the country code
                                -- according to Annex B of ITU-T T.35

        manufacturerCode       INTEGER (0..65535)    -- assigned nationally
    }
}

```

```

-- =====
-- Master-slave determination definitions
-- =====

MasterSlaveDetermination ::= SEQUENCE
{
    terminalType                INTEGER (0..255),
    statusDeterminationNumber    INTEGER (0..16777215),
    ...
}

MasterSlaveDeterminationAck ::= SEQUENCE
{
    decision                    CHOICE
    {
        master                  NULL,
        slave                   NULL
    },
    ...
}

MasterSlaveDeterminationReject ::= SEQUENCE
{
    cause                        CHOICE
    {
        identicalNumbers        NULL,
        ...
    },
    ...
}

MasterSlaveDeterminationRelease ::= SEQUENCE
{
    ...
}

-- =====
-- Capability exchange definitions
-- =====

TerminalCapabilitySet ::= SEQUENCE
{
    sequenceNumber              SequenceNumber,

    protocolIdentifier           OBJECT IDENTIFIER,
    -- shall be set to the value
    -- {ITU-T (0) recommendation (0) h (8) 245}
    -- version (0) 16}

    multiplexCapability          MultiplexCapability OPTIONAL,

    capabilityTable              SET SIZE (1..256) OF CapabilityTableEntry OPTIONAL,

    capabilityDescriptors        SET SIZE (1..256) OF CapabilityDescriptor OPTIONAL,

    ...,
    genericInformation           SEQUENCE OF GenericInformation OPTIONAL
    -- generic information associated
    -- with the message
}

```

```

CapabilityTableEntry ::= SEQUENCE
{
    capabilityTableEntryNumber      CapabilityTableEntryNumber,
    capability                       Capability OPTIONAL
}

CapabilityDescriptor ::= SEQUENCE
{
    capabilityDescriptorNumber      CapabilityDescriptorNumber,
    simultaneousCapabilities        SET SIZE (1..256) OF AlternativeCapabilitySet OPTIONAL
}
AlternativeCapabilitySet ::= SEQUENCE SIZE (1..256) OF CapabilityTableEntryNumber

CapabilityTableEntryNumber ::= INTEGER (1..65535)

CapabilityDescriptorNumber ::= INTEGER (0..255)

TerminalCapabilitySetAck ::= SEQUENCE
{
    sequenceNumber                  SequenceNumber,
    ...,
    genericInformation              SEQUENCE OF GenericInformation OPTIONAL
                                     -- generic information associated
                                     -- with the message
}

TerminalCapabilitySetReject ::= SEQUENCE
{
    sequenceNumber                  SequenceNumber,
    cause                           CHOICE
    {
        unspecified                 NULL,
        undefinedTableEntryUsed     NULL,
        descriptorCapacityExceeded  NULL,
        tableEntryCapacityExceeded  CHOICE
        {
            highestEntryNumberProcessed  CapabilityTableEntryNumber,
            noneProcessed                 NULL
        },
        ...
    },
    ...,
    genericInformation              SEQUENCE OF GenericInformation OPTIONAL
                                     -- generic information associated
                                     -- with the message
}

TerminalCapabilitySetRelease ::= SEQUENCE
{
    ...,
    genericInformation              SEQUENCE OF GenericInformation OPTIONAL
                                     -- generic information associated
                                     -- with the message
}

-- =====
-- Capability exchange definitions: top level capability description
-- =====

Capability ::= CHOICE
{
    nonStandard                     NonStandardParameter,

```

```

receiveVideoCapability          VideoCapability,
transmitVideoCapability        VideoCapability,
receiveAndTransmitVideoCapability VideoCapability,

receiveAudioCapability          AudioCapability,
transmitAudioCapability        AudioCapability,
receiveAndTransmitAudioCapability AudioCapability,

receiveDataApplicationCapability DataApplicationCapability,
transmitDataApplicationCapability DataApplicationCapability,
receiveAndTransmitDataApplicationCapability DataApplicationCapability,

h233EncryptionTransmitCapability BOOLEAN,
h233EncryptionReceiveCapability SEQUENCE
{
    h233IVResponseTime          INTEGER (0..255),    -- units milliseconds

    ...

},
...,
conferenceCapability           ConferenceCapability,
h235SecurityCapability         H235SecurityCapability,
maxPendingReplacementFor       INTEGER (0..255),
receiveUserInputCapability     UserInputCapability,
transmitUserInputCapability     UserInputCapability,
receiveAndTransmitUserInputCapability UserInputCapability,

genericControlCapability        GenericCapability,
receiveMultiplexedStreamCapability MultiplexedStreamCapability,
transmitMultiplexedStreamCapability MultiplexedStreamCapability,
receiveAndTransmitMultiplexedStreamCapability MultiplexedStreamCapability,
receiveRTPAudioTelephonyEventCapability AudioTelephonyEventCapability,
receiveRTPAudioToneCapability  AudioToneCapability,
depFecCapability               DepFECCapability, -- Deprecated, do not use
multiplePayloadStreamCapability MultiplePayloadStreamCapability,
fecCapability                   FECCapability,
redundancyEncodingCap          RedundancyEncodingCapability,
oneOfCapabilities              AlternativeCapabilitySet
}

H235SecurityCapability ::= SEQUENCE
{
    encryptionAuthenticationAndIntegrity EncryptionAuthenticationAndIntegrity,

    mediaCapability           CapabilityTableEntryNumber,
        -- NOTE - The mediaCapability shall refer to Capability Table Entries
        -- that do contain, directly or indirectly, one or more transmit,
        -- receive, or receiveAndTransmit AudioCapability, VideoCapability,
        -- DataApplicationCapability, or similar capabilities indicated by a
        -- NonStandardParameter or GenericCapability only

    ...
}

-- =====
-- Capability exchange definitions: Multiplex capabilities
-- =====

MultiplexCapability ::= CHOICE
{
    nonStandard           NonStandardParameter,
    h222Capability        H222Capability,
    h223Capability        H223Capability,
    v76Capability         V76Capability,
    ...

```

|                             |                                        |
|-----------------------------|----------------------------------------|
| h2250Capability             | H2250Capability,                       |
| genericMultiplexCapability  | GenericCapability                      |
| }                           |                                        |
| H222Capability ::= SEQUENCE |                                        |
| {                           |                                        |
| numberOfVCs                 | INTEGER (1..256),                      |
| vcCapability                | SET OF VCCapability,                   |
| ...                         |                                        |
| }                           |                                        |
| VCCapability ::= SEQUENCE   |                                        |
| {                           |                                        |
| aall                        | SEQUENCE                               |
| {                           |                                        |
| nullClockRecovery           | BOOLEAN,                               |
| srtsClockRecovery           | BOOLEAN,                               |
| adaptiveClockRecovery       | BOOLEAN,                               |
| nullErrorCorrection         | BOOLEAN,                               |
| longInterleaver             | BOOLEAN,                               |
| shortInterleaver            | BOOLEAN,                               |
| errorCorrectionOnly         | BOOLEAN,                               |
| structuredDataTransfer      | BOOLEAN,                               |
| partiallyFilledCells        | BOOLEAN,                               |
| ...                         |                                        |
| } OPTIONAL,                 |                                        |
| aal5                        | SEQUENCE                               |
| {                           |                                        |
| forwardMaximumSDUSize       | INTEGER (0..65535), -- units octets    |
| backwardMaximumSDUSize      | INTEGER (0..65535), -- units octets    |
| ...                         |                                        |
| } OPTIONAL,                 |                                        |
| transportStream             | BOOLEAN,                               |
| programStream               | BOOLEAN,                               |
| availableBitRates           | SEQUENCE                               |
| {                           |                                        |
| type                        | CHOICE                                 |
| {                           |                                        |
| singleBitRate               | INTEGER (1..65535), -- units 64 kbit/s |
| rangeOfBitRates             | SEQUENCE                               |
| {                           |                                        |
| lowerBitRate                | INTEGER (1..65535), -- units 64 kbit/s |
| higherBitRate               | INTEGER (1..65535) -- units 64 kbit/s  |
| }                           |                                        |
| },                          |                                        |
| ...                         |                                        |
| },                          |                                        |
| ...,                        |                                        |
| aallViaGateway              | SEQUENCE                               |
| {                           |                                        |
| gatewayAddress              | SET SIZE(1..256) OF Q2931Address,      |
| nullClockRecovery           | BOOLEAN,                               |
| srtsClockRecovery           | BOOLEAN,                               |
| adaptiveClockRecovery       | BOOLEAN,                               |
| nullErrorCorrection         | BOOLEAN,                               |
| longInterleaver             | BOOLEAN,                               |
| shortInterleaver            | BOOLEAN,                               |
| errorCorrectionOnly         | BOOLEAN,                               |
| structuredDataTransfer      | BOOLEAN,                               |
| partiallyFilledCells        | BOOLEAN,                               |
| ...                         |                                        |
| } OPTIONAL                  |                                        |

```

}

H223Capability ::= SEQUENCE
{
    transportWithI-frames          BOOLEAN,          -- I-frame transport
                                                -- of ITU-T H.245

    videoWithAL1                   BOOLEAN,
    videoWithAL2                   BOOLEAN,
    videoWithAL3                   BOOLEAN,
    audioWithAL1                   BOOLEAN,
    audioWithAL2                   BOOLEAN,
    audioWithAL3                   BOOLEAN,
    dataWithAL1                    BOOLEAN,
    dataWithAL2                    BOOLEAN,
    dataWithAL3                    BOOLEAN,

    maximumAL2SDUSize              INTEGER (0..65535), -- units octets
    maximumAL3SDUSize              INTEGER (0..65535), -- units octets

    maximumDelayJitter             INTEGER (0..1023),  -- units milliseconds

    h223MultiplexTableCapability CHOICE
    {
        basic                      NULL,
        enhanced SEQUENCE
        {
            maximumNestingDepth    INTEGER (1..15),
            maximumElementListSize INTEGER (2..255),
            maximumSubElementListSize INTEGER (2..255),
            ...
        }
    },
    ...,
    maxMUXPDUSizeCapability        BOOLEAN,
    nsrpSupport                    BOOLEAN,
    mobileOperationTransmitCapability SEQUENCE
    {
        modeChangeCapability       BOOLEAN,
        h223AnnexA                 BOOLEAN,
        h223AnnexADoubleFlag       BOOLEAN,
        h223AnnexB                 BOOLEAN,
        h223AnnexBwithHeader       BOOLEAN,
        ...
    } OPTIONAL,
    h223AnnexCCapability           H223AnnexCCapability OPTIONAL,

    bitrate                        INTEGER (1..19200) OPTIONAL, -- units of
                                                                -- 100 bit/s

    mobileMultilinkFrameCapability SEQUENCE
    {
        maximumSampleSize          INTEGER (1..255),      -- units octets
        maximumPayloadLength        INTEGER (1..65025),    -- units octets
        ...
    } OPTIONAL
}

H223AnnexCCapability ::= SEQUENCE
{
    videoWithAL1M                 BOOLEAN,
    videoWithAL2M                 BOOLEAN,
    videoWithAL3M                 BOOLEAN,
    audioWithAL1M                 BOOLEAN,

```

```

audioWithAL2M          BOOLEAN,
audioWithAL3M          BOOLEAN,
dataWithAL1M           BOOLEAN,
dataWithAL2M           BOOLEAN,
dataWithAL3M           BOOLEAN,
alpduInterleaving      BOOLEAN,

maximumAL1MPDUSize     INTEGER (0..65535),  -- units octets
maximumAL2MSDUSize     INTEGER (0..65535),  -- units octets
maximumAL3MSDUSize     INTEGER (0..65535),  -- units octets
...,
rsCodeCapability       BOOLEAN OPTIONAL
}

V76Capability ::= SEQUENCE
{
    suspendResumeCapabilitywAddress  BOOLEAN,
    suspendResumeCapabilitywoAddress  BOOLEAN,
    rejCapability                     BOOLEAN,
    sREJCapability                     BOOLEAN,
    mREJCapability                     BOOLEAN,
    crc8bitCapability                 BOOLEAN,
    crc16bitCapability                 BOOLEAN,
    crc32bitCapability                 BOOLEAN,
    uihCapability                     BOOLEAN,
    numOfDLCS INTEGER (2..8191),
    twoOctetAddressFieldCapability     BOOLEAN,
    loopBackTestCapability              BOOLEAN,
    n401Capability                     INTEGER (1..4095),
    maxWindowSizeCapability             INTEGER (1..127),
    v75Capability                     V75Capability,
    ...
}

V75Capability ::= SEQUENCE
{
    audioHeader                     BOOLEAN,
    ...
}

H2250Capability ::= SEQUENCE
{
    maximumAudioDelayJitter         INTEGER(0..1023),  -- units in
                                                -- milliseconds

    receiveMultipointCapability     MultipointCapability,
    transmitMultipointCapability     MultipointCapability,
    receiveAndTransmitMultipointCapability MultipointCapability,
    mcCapability                     SEQUENCE
    {
        centralizedConferenceMC      BOOLEAN,
        decentralizedConferenceMC     BOOLEAN,
        ...
    },
    rtcpVideoControlCapability       BOOLEAN,  -- FIR and NACK
    mediaPacketizationCapability     MediaPacketizationCapability,
    ...,
    transportCapability              TransportCapability OPTIONAL,
    redundancyEncodingCapability     SEQUENCE SIZE(1..256) OF RedundancyEncodingCapability OPTIONAL,
    logicalChannelSwitchingCapability BOOLEAN,
    t120DynamicPortCapability        BOOLEAN
}

```

```

MediaPacketizationCapability ::= SEQUENCE
{
    h261aVideoPacketization      BOOLEAN,
    ...,
    rtpPayloadType                SEQUENCE SIZE(1..256) OF RTPPayloadType OPTIONAL
}

RSVPPParameters ::= SEQUENCE
{
    qosMode                      QOSMode OPTIONAL,
    tokenRate                    INTEGER (1..4294967295) OPTIONAL,
                                -- rate in bytes/s
    bucketSize                   INTEGER (1..4294967295) OPTIONAL,
                                -- size in bytes
    peakRate                    INTEGER (1..4294967295) OPTIONAL,
                                -- peak bandwidth bytes/s
    minPoliced                   INTEGER (1..4294967295) OPTIONAL,
                                --
    maxPktSize                   INTEGER (1..4294967295) OPTIONAL,
                                -- size in bytes
    ...
}

QOSMode ::= CHOICE
{
    guaranteedQOS                NULL,
    controlledLoad               NULL,
    ...
}

ATMPParameters ::= SEQUENCE
{
    maxNTUSize                   INTEGER(0..65535), -- units in octets
    atmUBR                      BOOLEAN,           -- unspecified bit rate
    atmrtVBR                    BOOLEAN,           -- real time variable
                                -- bit rate
    atmnrVBR                    BOOLEAN,           -- non real time
                                -- variable bit rate
    atmABR                      BOOLEAN,           -- available bit rate
    atmCBR                      BOOLEAN,           -- constant bit rate
    ...
}

ServicePriorityValue ::= SEQUENCE
{
    nonStandardParameter         NonStandardParameter OPTIONAL,
    ...,
    value                       INTEGER(0..255)
}

ServicePriority ::= SEQUENCE
{
    nonStandardData              NonStandardParameter OPTIONAL,
    servicePrioritySignalled     BOOLEAN,
    servicePriorityValue         ServicePriorityValue OPTIONAL,
    ...,
    serviceClass                 INTEGER(0..4095) OPTIONAL,
    serviceSubclass              INTEGER(0..255) OPTIONAL
}

AuthorizationParameters ::= SEQUENCE
{
    nonStandardData              NonStandardParameter OPTIONAL,
    ...
}

```

```

}

QOSType ::= CHOICE
{
    desired                NULL,
    required                NULL,
    ...
}

QOSClass ::= CHOICE
{
    class0                  NULL,
    class1                  NULL,
    class2                  NULL,
    class3                  NULL,
    class4                  NULL,
    class5                  NULL,
    ...
}

QOSDescriptor ::= SEQUENCE
{
    nonStandardData        NonStandardParameter OPTIONAL,
    qosType                 QOSType,
    qosClass                QOSClass,
    ...
}

GenericTransportParameters ::= SEQUENCE
{
    nonStandardData        NonStandardParameter OPTIONAL,
    averageRate             INTEGER (1..4294967295) OPTIONAL,
                           -- average bandwidth bytes/s
    burst                   INTEGER (1..4294967295) OPTIONAL,
                           -- size in bytes
    peakRate                INTEGER (1..4294967295) OPTIONAL,
                           -- peak bandwidth bytes/s
    maxPktSize              INTEGER (1..4294967295) OPTIONAL,
                           -- size in bytes
    ...
}

QOSCapability ::= SEQUENCE
{
    nonStandardData        NonStandardParameter OPTIONAL,
    rsvpParameters          RSVPParameters OPTIONAL,
    atmParameters           ATMPParameters OPTIONAL,
    ...,
    localQoS                BOOLEAN OPTIONAL,
    genericTransportParameters GenericTransportParameters OPTIONAL,
    servicePriority          ServicePriority OPTIONAL,
    authorizationParameter  AuthorizationParameters OPTIONAL,
    qosDescriptor            QOSDescriptor OPTIONAL,
    dscpValue                INTEGER (0..63) OPTIONAL
}

MediaTransportType ::= CHOICE
{
    ip-UDP                  NULL,
    ip-TCP                  NULL,
    atm-AAL5-UNIDIR         NULL, -- virtual circuits used as unidirectional
    atm-AAL5-BIDIR          NULL, -- virtual circuits used as bidirectional
    ...,
    atm-AAL5-compressed      SEQUENCE
    {

```

```

        variable-delta                BOOLEAN,
        ...
    }
}

MediaChannelCapability ::= SEQUENCE
{
    mediaTransport                    MediaTransportType OPTIONAL,
    ...
}

TransportCapability ::= SEQUENCE
{
    nonStandard                      NonStandardParameter OPTIONAL,
    qosCapabilities                   SEQUENCE SIZE(1..256) OF QOSCapability OPTIONAL,
    mediaChannelCapabilities          SEQUENCE SIZE(1..256) OF MediaChannelCapability OPTIONAL,
    ...
}

RedundancyEncodingCapability ::= SEQUENCE
{
    redundancyEncodingMethod          RedundancyEncodingMethod,
    primaryEncoding                   CapabilityTableEntryNumber,
    secondaryEncoding                 SEQUENCE SIZE(1..256) OF CapabilityTableEntryNumber OPTIONAL,
    -- must be Audio, Video, or Data capabilities, not derived
    -- capabilities; redundancy order is inferred from number of
    -- secondary encodings
    ...
}

RedundancyEncodingMethod ::= CHOICE
{
    nonStandard                      NonStandardParameter,
    rtpAudioRedundancyEncoding       NULL,
    ...,
    rtpH263VideoRedundancyEncoding   RTPH263VideoRedundancyEncoding
}

RTPH263VideoRedundancyEncoding ::= SEQUENCE
{
    numberOfThreads                   INTEGER (1..16),
    framesBetweenSyncPoints           INTEGER (1..256),
    frameToThreadMapping              CHOICE
    {
        roundrobin                   NULL,
        custom                       SEQUENCE SIZE(1..256) OF
            RTPH263VideoRedundancyFrameMapping,
        -- empty SEQUENCE for capability negotiation
        -- meaningful contents only OpenLogicalChannel
        ...
    },
    containedThreads                   SEQUENCE SIZE(1..256) OF INTEGER (0..15) OPTIONAL,
    -- only used for opening of logical channels
    ...
}

RTPH263VideoRedundancyFrameMapping ::= SEQUENCE
{
    threadNumber                      INTEGER (0..15),
    frameSequence                     SEQUENCE SIZE(1..256) OF INTEGER (0..255),
    ...
}

```

```

MultipointCapability ::= SEQUENCE
{
    multicastCapability          BOOLEAN,
    multiUniCastConference      BOOLEAN,
    mediaDistributionCapability  SEQUENCE OF MediaDistributionCapability,
    ...
}

MediaDistributionCapability ::= SEQUENCE
{
    centralizedControl          BOOLEAN,
    distributedControl          BOOLEAN,
    -- for further study in
    -- Rec. ITU-T H.323

    centralizedAudio           BOOLEAN,
    distributedAudio           BOOLEAN,
    centralizedVideo           BOOLEAN,
    distributedVideo           BOOLEAN,
    centralizedData            SEQUENCE OF DataApplicationCapability OPTIONAL,
    distributedData            SEQUENCE OF DataApplicationCapability OPTIONAL,
    -- for further study in
    -- Rec. ITU-T H.323
    ...
}

-- =====
-- Capability exchange definitions: Video capabilities
-- =====

VideoCapability ::= CHOICE
{
    nonStandard                NonStandardParameter,
    h261VideoCapability        H261VideoCapability,
    h262VideoCapability        H262VideoCapability,
    h263VideoCapability        H263VideoCapability,
    is11172VideoCapability     IS11172VideoCapability,
    ...,
    genericVideoCapability     GenericCapability,
    extendedVideoCapability    ExtendedVideoCapability
}

ExtendedVideoCapability ::= SEQUENCE
{
    videoCapability            SEQUENCE OF VideoCapability,
    videoCapabilityExtension   SEQUENCE OF GenericCapability OPTIONAL,
    ...
}

H261VideoCapability ::= SEQUENCE
{
    qcifMPI                   INTEGER (1..4) OPTIONAL, -- units 1/29.97 Hz
    cifMPI                    INTEGER (1..4) OPTIONAL, -- units 1/29.97 Hz
    temporalSpatialTradeOffCapability  BOOLEAN,
    maxBitRate                INTEGER (1..19200),
    -- units of
    -- 100 bit/s
    stillImageTransmission    BOOLEAN,
    -- Annex D of
    -- ITU-T H.261
    ...,
    videoBadMBsCap            BOOLEAN
}

```

H262VideoCapability ::= SEQUENCE

```
{
    profileAndLevel-SpatML          BOOLEAN,
    profileAndLevel-MPatLL          BOOLEAN,
    profileAndLevel-MPatML          BOOLEAN,
    profileAndLevel-MPatH-14        BOOLEAN,
    profileAndLevel-MPatHL          BOOLEAN,
    profileAndLevel-SNRatLL         BOOLEAN,
    profileAndLevel-SNRatML         BOOLEAN,
    profileAndLevel-SpatialatH-14   BOOLEAN,
    profileAndLevel-HPatML          BOOLEAN,
    profileAndLevel-HPatH-14        BOOLEAN,
    profileAndLevel-HPatHL          BOOLEAN,
    videoBitRate                    INTEGER (0.. 1073741823) OPTIONAL, -- units 400 bit/s
    vbvBufferSize                   INTEGER (0.. 262143) OPTIONAL,      -- units 16 384 bits
    samplesPerLine                   INTEGER (0..16383) OPTIONAL,      -- units samples/line
    linesPerFrame                    INTEGER (0..16383) OPTIONAL,      -- units lines/frame
    framesPerSecond                  INTEGER (0..15) OPTIONAL,         -- frame_rate_code
    luminanceSampleRate              INTEGER (0..4294967295) OPTIONAL,  -- units samples/s
    ...,
    videoBadMBsCapBOOLEAN
}
```

H263VideoCapability ::= SEQUENCE

```
{
    sqcifMPI                        INTEGER (1..32) OPTIONAL, -- units 1/29.97 Hz
    qcifMPI                        INTEGER (1..32) OPTIONAL, -- units 1/29.97 Hz
    cifMPI                         INTEGER (1..32) OPTIONAL, -- units 1/29.97 Hz
    cif4MPI                       INTEGER (1..32) OPTIONAL, -- units 1/29.97 Hz
    cif16MPI                      INTEGER (1..32) OPTIONAL, -- units 1/29.97 Hz
    maxBitRate INTEGER (1..192400), -- units 100 bit/s
    unrestrictedVector             BOOLEAN,
    arithmeticCoding               BOOLEAN,
    advancedPrediction             BOOLEAN,
    pbFrames                       BOOLEAN,
    temporalSpatialTradeOffCapability BOOLEAN,
    hrd-B                          INTEGER (0..524287) OPTIONAL, -- units 128 bits
    bppMaxKb                      INTEGER (0..65535) OPTIONAL, -- units 1024 bits
    ...,

    slowSqcifMPI                  INTEGER (1..3600) OPTIONAL, -- units seconds/frame
    slowQcifMPI                   INTEGER (1..3600) OPTIONAL, -- units seconds/frame
    slowCifMPI                    INTEGER (1..3600) OPTIONAL, -- units seconds/frame
    slowCif4MPI                   INTEGER (1..3600) OPTIONAL, -- units seconds/frame
    slowCif16MPI                  INTEGER (1..3600) OPTIONAL, -- units seconds/frame
    errorCompensation              BOOLEAN,

    enhancementLayerInfo           EnhancementLayerInfo OPTIONAL,
    h263Options                    H263Options OPTIONAL
}
```

EnhancementLayerInfo ::= SEQUENCE

```
{
    baseBitRateConstrained         BOOLEAN,
    snrEnhancement                 SET SIZE(1..14) OF EnhancementOptions OPTIONAL,
    spatialEnhancement             SET SIZE(1..14) OF EnhancementOptions OPTIONAL,
    bPictureEnhancement           SET SIZE(1..14) OF BEnhancementParameters OPTIONAL,
    ...
}
```

BEnhancementParameters ::= SEQUENCE

```
{
    enhancementOptions             EnhancementOptions,
    numberOfBPictures              INTEGER (1..64),
    ...
}
```

EnhancementOptions ::= SEQUENCE

|                                   |                             |                        |
|-----------------------------------|-----------------------------|------------------------|
| {                                 |                             |                        |
| sqcifMPI                          | INTEGER (1..32) OPTIONAL,   | -- units 1/29.97 Hz    |
| qcifMPI                           | INTEGER (1..32) OPTIONAL,   | -- units 1/29.97 Hz    |
| cifMPI                            | INTEGER (1..32) OPTIONAL,   | -- units 1/29.97 Hz    |
| cif4MPI                           | INTEGER (1..32) OPTIONAL,   | -- units 1/29.97 Hz    |
| cif16MPI                          | INTEGER (1..32) OPTIONAL,   | -- units 1/29.97 Hz    |
| maxBitRate                        | INTEGER (1..192400),        | -- units 100 bit/s     |
| unrestrictedVector                | BOOLEAN,                    |                        |
| arithmeticCoding                  | BOOLEAN,                    |                        |
| temporalSpatialTradeOffCapability | BOOLEAN,                    |                        |
| slowSqcifMPI                      | INTEGER (1..3600) OPTIONAL, | -- units seconds/frame |
| slowQcifMPI                       | INTEGER (1..3600) OPTIONAL, | -- units seconds/frame |
| slowCifMPI                        | INTEGER (1..3600) OPTIONAL, | -- units seconds/frame |
| slowCif4MPI                       | INTEGER (1..3600) OPTIONAL, | -- units seconds/frame |
| slowCif16MPI                      | INTEGER (1..3600) OPTIONAL, | -- units seconds/frame |
| errorCompensation                 | BOOLEAN,                    |                        |
| h263Options                       | H263Options OPTIONAL,       |                        |
| ...                               |                             |                        |
| }                                 |                             |                        |

H263Options ::= SEQUENCE

|                                    |                                                           |  |
|------------------------------------|-----------------------------------------------------------|--|
| {                                  |                                                           |  |
| advancedIntraCodingMode            | BOOLEAN,                                                  |  |
| deblockingFilterMode               | BOOLEAN,                                                  |  |
| improvedPBFramesMode               | BOOLEAN,                                                  |  |
| unlimitedMotionVectors             | BOOLEAN,                                                  |  |
| fullPictureFreeze                  | BOOLEAN,                                                  |  |
| partialPictureFreezeAndRelease     | BOOLEAN,                                                  |  |
| resizingPartPicFreezeAndRelease    | BOOLEAN,                                                  |  |
| fullPictureSnapshot                | BOOLEAN,                                                  |  |
| partialPictureSnapshot             | BOOLEAN,                                                  |  |
| videoSegmentTagging                | BOOLEAN,                                                  |  |
| progressiveRefinement              | BOOLEAN,                                                  |  |
| dynamicPictureResizingByFour       | BOOLEAN,                                                  |  |
| dynamicPictureResizingSixteenthPel | BOOLEAN,                                                  |  |
| dynamicWarpingHalfPel              | BOOLEAN,                                                  |  |
| dynamicWarpingSixteenthPel         | BOOLEAN,                                                  |  |
| independentSegmentDecoding         | BOOLEAN,                                                  |  |
| slicesInOrder-NonRect              | BOOLEAN,                                                  |  |
| slicesInOrder-Rect                 | BOOLEAN,                                                  |  |
| slicesNoOrder-NonRect              | BOOLEAN,                                                  |  |
| slicesNoOrder-Rect                 | BOOLEAN,                                                  |  |
| alternateInterVLCMode              | BOOLEAN,                                                  |  |
| modifiedQuantizationMode           | BOOLEAN,                                                  |  |
| reducedResolutionUpdate            | BOOLEAN,                                                  |  |
| transparencyParameters             | TransparencyParameters OPTIONAL,                          |  |
| separateVideoBackChannel           | BOOLEAN,                                                  |  |
| refPictureSelection                | RefPictureSelection OPTIONAL,                             |  |
| customPictureClockFrequency        | SET SIZE (1..16) OF CustomPictureClockFrequency OPTIONAL, |  |
| customPictureFormat                | SET SIZE (1..16) OF CustomPictureFormat OPTIONAL,         |  |
| modeCombos                         | SET SIZE (1..16) OF H263VideoModeCombos OPTIONAL,         |  |
| ....,                              |                                                           |  |
| videoBadMBsCap                     | BOOLEAN,                                                  |  |
| h263Version3Options                | H263Version3Options                                       |  |
| }                                  |                                                           |  |

```

TransparencyParameters ::= SEQUENCE
{
    presentationOrder          INTEGER(1..256),
    offset-x                    INTEGER(-262144..262143), -- 1/8 pixels
    offset-y                    INTEGER(-262144..262143), -- 1/8 pixels
    scale-x                     INTEGER(1..255),
    scale-y                     INTEGER(1..255),
    ...
}

RefPictureSelection ::= SEQUENCE
{
    additionalPictureMemory     SEQUENCE
    {
        sqcifAdditionalPictureMemory  INTEGER(1..256) OPTIONAL, -- units frame
        qcifAdditionalPictureMemory    INTEGER(1..256) OPTIONAL, -- units frame
        cifAdditionalPictureMemory     INTEGER(1..256) OPTIONAL, -- units frame
        cif4AdditionalPictureMemory    INTEGER(1..256) OPTIONAL, -- units frame
        cif16AdditionalPictureMemory   INTEGER(1..256) OPTIONAL, -- units frame
        bigCpfAdditionalPictureMemory  INTEGER(1..256) OPTIONAL, -- units frame
        ...
    } OPTIONAL,
    videoMux                     BOOLEAN,
    videoBackChannelSend        CHOICE
    {
        none                     NULL,
        ackMessageOnly           NULL,
        nackMessageOnly          NULL,
        ackOrNackMessageOnly     NULL,
        ackAndNackMessage        NULL,
        ...
    },
    ...,
    enhancedReferencePicSelect   SEQUENCE
    {
        subPictureRemovalParameters SEQUENCE
        {
            mpuHorizMBs           INTEGER(1..128),
            mpuVertMBs            INTEGER(1..72),
            mpuTotalNumber        INTEGER(1..65536),
            ...
        } OPTIONAL,
        ...
    }
}

CustomPictureClockFrequency ::= SEQUENCE
{
    clockConversionCode          INTEGER(1000..1001),
    clockDivisor                 INTEGER(1..127),
    sqcifMPI                     INTEGER(1..2048) OPTIONAL,
    qcifMPI                      INTEGER(1..2048) OPTIONAL,
    cifMPI                       INTEGER(1..2048) OPTIONAL,
    cif4MPI                     INTEGER(1..2048) OPTIONAL,
    cif16MPI                    INTEGER(1..2048) OPTIONAL,
    ...
}

CustomPictureFormat ::= SEQUENCE
{
    maxCustomPictureWidth        INTEGER(1..2048), -- units 4 pixels
    maxCustomPictureHeight       INTEGER(1..2048), -- units 4 pixels
    minCustomPictureWidth        INTEGER(1..2048), -- units 4 pixels
    minCustomPictureHeight       INTEGER(1..2048), -- units 4 pixels
}

```

|                                    |                                         |
|------------------------------------|-----------------------------------------|
| mPI                                | SEQUENCE                                |
| {                                  |                                         |
| standardMPI                        | INTEGER (1..31) OPTIONAL,               |
| customPCF                          | SET SIZE (1..16) OF SEQUENCE            |
| {                                  |                                         |
| clockConversionCode                | INTEGER (1000..1001),                   |
| clockDivisor                       | INTEGER (1..127),                       |
| customMPI                          | INTEGER (1..2048),                      |
| ...                                |                                         |
| } OPTIONAL,                        |                                         |
| ...                                |                                         |
| },                                 |                                         |
| pixelAspectInformation             | CHOICE                                  |
| {                                  |                                         |
| anyPixelAspectRatio                | BOOLEAN,                                |
| pixelAspectCode                    | SET SIZE (1..14) OF INTEGER(1..14),     |
| extendedPAR                        | SET SIZE (1..256) OF SEQUENCE           |
| {                                  |                                         |
| width                              | INTEGER(1..255),                        |
| height                             | INTEGER(1..255),                        |
| ...                                |                                         |
| },                                 |                                         |
| ...                                |                                         |
| },                                 |                                         |
| ...                                |                                         |
| }                                  |                                         |
| H263VideoModeCombos ::= SEQUENCE   |                                         |
| {                                  |                                         |
| h263VideoUncoupledModes            | H263ModeComboFlags,                     |
| h263VideoCoupledModes              | SET SIZE (1..16) OF H263ModeComboFlags, |
| ...                                |                                         |
| }                                  |                                         |
| H263ModeComboFlags ::= SEQUENCE    |                                         |
| {                                  |                                         |
| unrestrictedVector                 | BOOLEAN,                                |
| arithmeticCoding                   | BOOLEAN,                                |
| advancedPrediction                 | BOOLEAN,                                |
| pbFrames                           | BOOLEAN,                                |
| advancedIntraCodingMode            | BOOLEAN,                                |
| deblockingFilterMode               | BOOLEAN,                                |
| unlimitedMotionVectors             | BOOLEAN,                                |
| slicesInOrder-NonRect              | BOOLEAN,                                |
| slicesInOrder-Rect                 | BOOLEAN,                                |
| slicesNoOrder-NonRect              | BOOLEAN,                                |
| slicesNoOrder-Rect                 | BOOLEAN,                                |
| improvedPBFramesMode               | BOOLEAN,                                |
| referencePicSelect                 | BOOLEAN,                                |
| dynamicPictureResizingByFour       | BOOLEAN,                                |
| dynamicPictureResizingSixteenthPel | BOOLEAN,                                |
| dynamicWarpingHalfPel              | BOOLEAN,                                |
| dynamicWarpingSixteenthPel         | BOOLEAN,                                |
| reducedResolutionUpdate            | BOOLEAN,                                |
| independentSegmentDecoding         | BOOLEAN,                                |
| alternateInterVLCMode              | BOOLEAN,                                |
| modifiedQuantizationMode           | BOOLEAN,                                |
| ...,                               |                                         |
| enhancedReferencePicSelect         | BOOLEAN,                                |
| h263Version3Options                | H263Version3Options}                    |

```

H263Version3Options ::= SEQUENCE
{
    dataPartitionedSlices          BOOLEAN,
    fixedPointIDCT0                BOOLEAN,
    interlacedFields               BOOLEAN,
    currentPictureHeaderRepetition BOOLEAN,
    previousPictureHeaderRepetition BOOLEAN,
    nextPictureHeaderRepetition    BOOLEAN,
    pictureNumber                  BOOLEAN,
    spareReferencePictures          BOOLEAN,
    ...
}

IS11172VideoCapability ::= SEQUENCE
{
    constrainedBitstream          BOOLEAN,
    videoBitRate                  INTEGER (0.. 1073741823) OPTIONAL, -- units 400 bit/s
    vbvBufferSize                 INTEGER (0.. 262143) OPTIONAL,    -- units 16 384 bits
    samplesPerLine                 INTEGER (0..16383) OPTIONAL,      -- units samples/line
    linesPerFrame                  INTEGER (0..16383) OPTIONAL,      -- units lines/frame
    pictureRate                    INTEGER (0..15) OPTIONAL,
    luminanceSampleRate            INTEGER (0..4294967295) OPTIONAL, -- units samples/s
    ...,
    videoBadMBsCap                 BOOLEAN
}

-- =====
-- Capability exchange definitions: Audio capabilities
-- =====

-- For an ITU-T H.222 multiplex, the integers indicate the size of the STD
-- buffer
-- in units of 256 octets
-- For an ITU-T H.223 multiplex, the integers indicate the maximum number of
-- audio
-- frames per AL-SDU
-- For an ITU-T H.225.0 multiplex, the integers indicate the maximum number of
-- audio
-- frames per packet

AudioCapability ::= CHOICE
{
    nonStandard                    NonStandardParameter,
    g711Alaw64k                   INTEGER (1..256),
    g711Alaw56k                   INTEGER (1..256),
    g711Ulaw64k                   INTEGER (1..256),
    g711Ulaw56k                   INTEGER (1..256),

    g722-64k                       INTEGER (1..256),
    g722-56k                       INTEGER (1..256),
    g722-48k                       INTEGER (1..256),

    g7231                          SEQUENCE
    {
        maxAl-sduAudioFrames      INTEGER (1..256),
        silenceSuppression         BOOLEAN
    },

    g728                           INTEGER (1..256),
    g729                           INTEGER (1..256),
    g729AnnexA                     INTEGER (1..256),
    is11172AudioCapability          IS11172AudioCapability,
    is13818AudioCapability          IS13818AudioCapability,
    ...,

```

```

g729wAnnexB          INTEGER(1..256),
g729AnnexAwAnnexB    INTEGER(1..256),
g7231AnnexCCapability G7231AnnexCCapability,
gsmFullRate          GSMAudioCapability,
gsmHalfRate          GSMAudioCapability,
gsmEnhancedFullRate  GSMAudioCapability,
genericAudioCapability GenericCapability,
g729Extensions       G729Extensions,
vbd                  VBDCapability,
audioTelephonyEvent  NoPTAudioTelephonyEventCapability,
audioTone             NoPTAudioToneCapability
}

G729Extensions ::= SEQUENCE
{
    audioUnit          INTEGER (1..256) OPTIONAL,
    annexA             BOOLEAN,
    annexB             BOOLEAN,
    annexD             BOOLEAN,
    annexE             BOOLEAN,
    annexF             BOOLEAN,
    annexG             BOOLEAN,
    annexH             BOOLEAN,
    ...
}

G7231AnnexCCapability ::= SEQUENCE
{
    maxAl-sduAudioFrames INTEGER (1..256),
    silenceSuppression    BOOLEAN,
    g723AnnexCAudioMode  SEQUENCE
    {
        highRateMode0    INTEGER (27..78), -- units octets
        highRateMode1    INTEGER (27..78), -- units octets
        lowRateMode0     INTEGER (23..66), -- units octets
        lowRateMode1     INTEGER (23..66), -- units octets
        sidMode0         INTEGER (6..17),  -- units octets
        sidMode1         INTEGER (6..17),  -- units octets
        ...
    } OPTIONAL,
    ...
}

IS11172AudioCapability ::= SEQUENCE
{
    audioLayer1         BOOLEAN,
    audioLayer2         BOOLEAN,
    audioLayer3         BOOLEAN,

    audioSampling32k    BOOLEAN,
    audioSampling44k1   BOOLEAN,
    audioSampling48k    BOOLEAN,

    singleChannel       BOOLEAN,
    twoChannels         BOOLEAN,

    bitRate             INTEGER (1..448), -- units kbit/s
    ...
}

IS13818AudioCapability ::= SEQUENCE
{
    audioLayer1         BOOLEAN,
    audioLayer2         BOOLEAN,

```

```

        audioLayer3                                BOOLEAN,

        audioSampling16k                           BOOLEAN,
        audioSampling22k05                         BOOLEAN,
        audioSampling24k                           BOOLEAN,
        audioSampling32k                           BOOLEAN,
        audioSampling44k1                          BOOLEAN,
        audioSampling48k                           BOOLEAN,

        singleChannel                              BOOLEAN,
        twoChannels                                BOOLEAN,
        threeChannels2-1                           BOOLEAN,
        threeChannels3-0                           BOOLEAN,
        fourChannels2-0-2-0                        BOOLEAN,
        fourChannels2-2                            BOOLEAN,
        fourChannels3-1                            BOOLEAN,
        fiveChannels3-0-2-0                        BOOLEAN,
        fiveChannels3-2                            BOOLEAN,

        lowFrequencyEnhancement                    BOOLEAN,

        multilingual                                BOOLEAN,

        bitRate                                     INTEGER (1..1130),  -- units kbit/s
        ...
    }

    GSMAudioCapability ::= SEQUENCE
    {
        audioUnitSize                               INTEGER (1..256),
        comfortNoise                                BOOLEAN,
        scrambled                                    BOOLEAN,
        ...
    }

    VBDCapability ::= SEQUENCE
    {
        type                                         AudioCapability,  -- shall not be "vbd"
        ...
    }

-- =====
-- Capability exchange definitions: Data capabilities
-- =====

    DataApplicationCapability ::= SEQUENCE
    {
        application                                 CHOICE
        {
            nonStandard                             NonStandardParameter,
            t120                                     DataProtocolCapability,
            dsm-cc                                   DataProtocolCapability,
            userData                                 DataProtocolCapability,
            t84                                       SEQUENCE
            {
                t84Protocol                          DataProtocolCapability,
                t84Profile                           T84Profile
            },
            t434                                     DataProtocolCapability,
            h224                                     DataProtocolCapability,
            nlpid                                    SEQUENCE
            {
                nlpidProtocol                        DataProtocolCapability,
                nlpidData OCTET STRING
            }
        }
    }

```

```

    },
    dsvdControl                NULL,
    h222DataPartitioning       DataProtocolCapability,
    ...,
    t30fax                     DataProtocolCapability,
    t140                       DataProtocolCapability,
    t38fax                     SEQUENCE
    {
        t38FaxProtocol         DataProtocolCapability,
        t38FaxProfile          T38FaxProfile
    },
    genericDataCapability       GenericCapability
},
maxBitRate                    INTEGER (0..4294967295), -- units 100 bit/s
...
}

```

DataProtocolCapability ::= CHOICE

```

{
    nonStandard                NonStandardParameter,
    v14buffered                NULL,
    v42lapm                    NULL, -- may negotiate to ITU-T V.42
                                -- bis
    hdlcFrameTunnelling        NULL,
    h310SeparateVCStack        NULL,
    h310SingleVCStack          NULL,
    transparent                 NULL,
    ...,
    segmentationAndReassembly  NULL,
    hdlcFrameTunnelingwSAR      NULL,
    v120                       NULL, -- as in ITU-T H.230
    separateLANStack           NULL,
    v76wCompression           CHOICE
    {
        transmitCompression     CompressionType,
        receiveCompression       CompressionType,
        transmitAndReceiveCompression CompressionType,
        ...
    },
    tcp                        NULL,
    udp                        NULL
}

```

CompressionType ::= CHOICE

```

{
    v42bis                    V42bis,
    ...
}

```

V42bis ::= SEQUENCE

```

{
    numberOfCodewords          INTEGER (1..65536),
    maximumStringLength        INTEGER (1..256),
    ...
}

```

T84Profile ::= CHOICE

```

{
    t84Unrestricted           NULL,
    t84Restricted              SEQUENCE
    {
        qcif                   BOOLEAN,
        cif                    BOOLEAN,
        ccir601Seq             BOOLEAN,
    }
}

```



```

        ccir601Prog          BOOLEAN,
        hdtvSeq              BOOLEAN,
        hdtvProg              BOOLEAN,

        g3Facsmh200x100      BOOLEAN,
        g3Facsmh200x200      BOOLEAN,
        g4Facsmmr200x100     BOOLEAN,
        g4Facsmmr200x200     BOOLEAN,
        jbig200x200Seq        BOOLEAN,
        jbig200x200Prog       BOOLEAN,
        jbig300x300Seq        BOOLEAN,
        jbig300x300Prog       BOOLEAN,

        digPhotoLow           BOOLEAN,
        digPhotoMedSeq         BOOLEAN,
        digPhotoMedProg        BOOLEAN,
        digPhotoHighSeq        BOOLEAN,
        digPhotoHighProg       BOOLEAN,

        ...
    }
}

T38FaxProfile ::= SEQUENCE
{
    fillBitRemoval            BOOLEAN,
    transcodingJBIG            BOOLEAN,
    transcodingMMR              BOOLEAN,
    ...,
    version                    INTEGER (0..255),
                                -- Version 0, the default, refers to
                                -- ITU-T T.38 (2005)

    t38FaxRateManagement       T38FaxRateManagement,
                                -- The default Data Rate Management is
                                -- determined by the choice of
                                -- DataProtocolCapability

    t38FaxUdpOptions            T38FaxUdpOptions OPTIONAL,
                                -- For UDP, t38UDPRedundancy is the default

    t38FaxTcpOptions            T38FaxTcpOptions OPTIONAL
}

T38FaxRateManagement ::= CHOICE
{
    localTCF                    NULL,
    transferredTCF               NULL,
    ...
}

T38FaxUdpOptions ::= SEQUENCE
{
    t38FaxMaxBuffer             INTEGER OPTIONAL,
    t38FaxMaxDatagram           INTEGER OPTIONAL,
    t38FaxUdpEC                  CHOICE
    {
        t38UDPFEC                NULL,
        t38UDPRedundancy          NULL,
        ...
    }
}

T38FaxTcpOptions ::= SEQUENCE
{
    t38TCPBidirectionalMode     BOOLEAN,
    ...
}

```

```

}

-- =====
-- Encryption Capability Definitions
-- =====

EncryptionAuthenticationAndIntegrity ::= SEQUENCE
{
    encryptionCapability          EncryptionCapability OPTIONAL,
    authenticationCapability      AuthenticationCapability OPTIONAL,
    integrityCapability           IntegrityCapability OPTIONAL,
    ...,
    genericH235SecurityCapability GenericCapability OPTIONAL
}

EncryptionCapability ::= SEQUENCE SIZE(1..256) OF MediaEncryptionAlgorithm

MediaEncryptionAlgorithm ::= CHOICE
{
    nonStandard                  NonStandardParameter,
    algorithm                    OBJECT IDENTIFIER, -- many defined
                                     -- inter alia in
                                     -- ITU-T H.235 series
    ...
}

AuthenticationCapability ::= SEQUENCE
{
    nonStandard                  NonStandardParameter OPTIONAL,
    ...,
    antiSpamAlgorithm           OBJECT IDENTIFIER OPTIONAL
}

IntegrityCapability ::= SEQUENCE
{
    nonStandard                  NonStandardParameter OPTIONAL,
    ...
}

-- =====
-- Capability Exchange Definitions: UserInput
-- =====

UserInputCapability ::= CHOICE
{
    nonStandard                  SEQUENCE SIZE(1..16) OF NonStandardParameter,
    basicString                  NULL, -- alphanumeric
    iA5String                    NULL, -- alphanumeric
    generalString                NULL, -- alphanumeric
    dtmf                        NULL, -- supports dtmf using signal
                                     -- and signalUpdate
    hookflash                    NULL, -- supports hookflash using signal
    ...,
    extendedAlphanumeric         NULL,
    encryptedBasicString         NULL, -- encrypted Basic string in
                                     -- encryptedAlphanumeric
    encryptedIA5String           NULL, -- encrypted IA5 string in
                                     -- encryptedSignalType
    encryptedGeneralString       NULL, -- encrypted general string in
                                     -- extendedAlphanumeric.encryptedalphanumeric
    secureDTMF                   NULL, -- secure DTMF using encryptedSignalType
    genericUserInputCapability   GenericCapability
}

```

```

-- =====
-- Capability Exchange Definitions: Conference
-- =====

ConferenceCapability ::= SEQUENCE
{
    nonStandardData                SEQUENCE OF NonStandardParameter OPTIONAL,
    chairControlCapability          BOOLEAN,
    ...,
    videoIndicateMixingCapability  BOOLEAN,
    multipointVisualizationCapability  BOOLEAN OPTIONAL -- same as ITU-T H.230 MVC
}

-- =====
-- Capability Exchange Definitions: Generic Capability
-- =====

GenericCapability ::= SEQUENCE
{
    capabilityIdentifier            CapabilityIdentifier,

    maxBitRate                     INTEGER (0..4294967295) OPTIONAL,
                                   -- Units 100 bit/s
    collapsing                     SEQUENCE OF GenericParameter OPTIONAL,
    nonCollapsing                  SEQUENCE OF GenericParameter OPTIONAL,
    nonCollapsingRaw               OCTET STRING OPTIONAL,
                                   -- Typically contains ASN.1
                                   -- PER encoded data describing capability
    transport                      DataProtocolCapability OPTIONAL,
    ...
}

CapabilityIdentifier ::= CHOICE
{
    standard                       OBJECT IDENTIFIER,
                                   -- e.g., { ITU-T (0) recommendation (0) h (8) 267
                                   -- version (0) 2 subIdentifier (0) }
    h221NonStandard                NonStandardParameter,
    uuid                           OCTET STRING ( SIZE (16) ),
    domainBased                    IA5String ( SIZE (1..64) ),
    ...
}

-- NOTE - The ranges of parameter values have been selected to ensure that the
-- GenericParameter preamble, standard part of ParameterIdentifier and the
-- encoding of that choice, and the preamble of ParameterValue to fit into
-- 2 octets.

GenericParameter ::= SEQUENCE
{
    parameterIdentifier            ParameterIdentifier,
    parameterValue                 ParameterValue,
    supersedes                     SEQUENCE OF ParameterIdentifier OPTIONAL,
    ...
}

ParameterIdentifier ::= CHOICE
{
    standard                       INTEGER (0..127), -- Assigned by
                                                -- Capability
                                                -- specifications
    h221NonStandard                NonStandardParameter, -- NOTE -
                                                -- NonStandardIdentifier
                                                -- is not sufficient in
                                                -- this case
}

```

```

    uuid                                OCTET STRING ( SIZE (16) ), -- For non-
                                         -- standard
    domainBased                         IA5String ( SIZE (1..64) ),
    ...
}

ParameterValue ::= CHOICE
{
    logical                             NULL, -- Only acceptable if
                                         -- all entities
                                         -- include this option
    booleanArray                        INTEGER (0..255), -- array of 8 logical
                                         -- types
    unsignedMin                         INTEGER (0..65535), -- Look for min
                                         -- common value
    unsignedMax                         INTEGER (0..65535), -- Look for max
                                         -- common value
    unsigned32Min                       INTEGER (0..4294967295), -- Look for min
                                         -- common value
    unsigned32Max                       INTEGER (0..4294967295), -- Look for max
                                         -- common value

    octetString                         OCTET STRING, -- non-collapsing
                                         -- octet string

    genericParameter SEQUENCE OF GenericParameter,
    ...
}

-- =====
-- Capability Exchange Definitions: Multiplexed Stream Capability
-- =====

MultiplexedStreamCapability ::= SEQUENCE
{
    multiplexFormat                      MultiplexFormat,
    controlOnMuxStream                  BOOLEAN,
    capabilityOnMuxStream                SET SIZE (1..256) OF
AlternativeCapabilitySet OPTIONAL,
    ...
}

MultiplexFormat ::= CHOICE
{
    nonStandard                         NonStandardParameter,
    h222Capability                      H222Capability,
    h223Capability                      H223Capability,
    ...
}

-- =====
-- Capability Exchange Definitions: AudioTelephonyEventCapability and AudioToneCapability
-- =====

AudioTelephonyEventCapability ::= SEQUENCE
{
    dynamicRTPPayloadType               INTEGER(96..127),
    audioTelephoneEvent                 GeneralString, -- As per <list of values>
                                         -- in IETF RFC 4733
    ...
}

AudioToneCapability ::= SEQUENCE
{

```

```

        dynamicRTPPayloadType          INTEGER(96..127),
        ...
    }

-- The following definitions are as above but without a Payload Type field.

NoPTAudioTelephonyEventCapability ::= SEQUENCE
{
    audioTelephoneEvent                GeneralString, -- As per <list of values>
                                                -- in IETF RFC 4733
    ...
}

NoPTAudioToneCapability ::= SEQUENCE
{
    ...
}

-- =====
-- Capability Exchange Definitions: MultiplePayloadStreamCapability
-- =====

MultiplePayloadStreamCapability ::= SEQUENCE
{
    capabilities                       SET SIZE(1..256) OF AlternativeCapabilitySet,
    ...
}

-- =====
-- Capability Exchange Definitions: FECCapability
-- =====

DepFECCapability ::= CHOICE -- Deprecated, do not use
{
    rfc2733                            SEQUENCE
    {
        redundancyEncoding             BOOLEAN,
        separateStream                 SEQUENCE
        {
            separatePort               BOOLEAN,
            samePort                   BOOLEAN,
            ...
        },
        ...
    },
    ...
}

FECCapability ::= SEQUENCE
{
    protectedCapability                CapabilityTableEntryNumber,
    fecScheme                          OBJECT IDENTIFIER OPTIONAL,
                                        -- identifies encoding scheme
    rfc2733Format                      CHOICE
    {
        rfc2733rfc2198                MaxRedundancy, -- IETF RFC 2198
        rfc2733sameport               MaxRedundancy,
                                        -- separate packet, same port
        rfc2733diffport               MaxRedundancy
                                        -- separate packet and port
    } OPTIONAL,
    ...
}

```

MaxRedundancy ::= INTEGER (1..MAX)

-- =====  
-- Logical channel signalling definitions  
-- =====

-- "Forward" is used to refer to transmission in the direction from the terminal  
-- making the original request for a logical channel to the other terminal, and  
-- "reverse" is used to refer to the opposite direction of transmission, in the  
-- case of a bidirectional channel request.

OpenLogicalChannel ::= SEQUENCE

```
{
    forwardLogicalChannelNumber    LogicalChannelNumber,

    forwardLogicalChannelParameters SEQUENCE
    {
        portNumber                INTEGER (0..65535) OPTIONAL,
        dataType                   DataType,
        multiplexParameters        CHOICE
        {
            h222LogicalChannelParameters H222LogicalChannelParameters,
            h223LogicalChannelParameters H223LogicalChannelParameters,
            v76LogicalChannelParameters  V76LogicalChannelParameters,
            ...,
            h2250LogicalChannelParameters H2250LogicalChannelParameters,
            none                          NULL -- for use with Separate Stack when
                                           -- multiplexParameters are not
                                           -- required or appropriate
        },
        ...,
        forwardLogicalChannelDependency LogicalChannelNumber OPTIONAL,
                                           -- also used to refer to the primary
                                           -- logical channel when using video
                                           -- redundancy coding
        replacementFor              LogicalChannelNumber OPTIONAL
    },

    -- Used to specify the reverse channel for bidirectional open request

    reverseLogicalChannelParameters SEQUENCE
    {
        dataType                   DataType,
        multiplexParameters        CHOICE
        {
            -- ITU-T H.222 parameters are never present in reverse direction
            h223LogicalChannelParameters H223LogicalChannelParameters,
            v76LogicalChannelParameters  V76LogicalChannelParameters,
            ...,
            h2250LogicalChannelParameters H2250LogicalChannelParameters
        } OPTIONAL, -- Not present for ITU-T H.222
        ...,
        reverseLogicalChannelDependency LogicalChannelNumber OPTIONAL,
                                           -- also used to refer to the primary logical channel when using
                                           -- video redundancy coding
        replacementFor              LogicalChannelNumber OPTIONAL
    } OPTIONAL, -- Not present for unidirectional channel request
    ...,
    separateStack                  NetworkAccessParameters OPTIONAL,
    -- for Open responder to establish the stack
}
```

```

    encryptionSync          EncryptionSync OPTIONAL,
        genericInformation   SEQUENCE OF GenericInformation OPTIONAL
                                -- generic information associated
                                -- with the message
}

LogicalChannelNumber ::= INTEGER (1..65535)

NetworkAccessParameters ::= SEQUENCE
{
    distribution CHOICE
    {
        unicast          NULL,
        multicast NULL,
        ...
    } OPTIONAL,

    networkAddress CHOICE
    {
        q2931Address      Q2931Address,
        e164Address        IA5String(SIZE(1..128)) (FROM ("0123456789#*,")),
        localAreaAddress   TransportAddress,
        ...
    },
    associateConference    BOOLEAN,
    externalReference      OCTET STRING(SIZE(1..255)) OPTIONAL,
    ...,
    t120SetupProcedure     CHOICE
    {
        originateCall      NULL,
        waitForCall         NULL,
        issueQuery          NULL,
        ...
    } OPTIONAL
}

Q2931Address ::= SEQUENCE
{
    address CHOICE
    {
        internationalNumber NumericString(SIZE(1..16)),
        nsapAddress          OCTET STRING (SIZE(1..20)),
        ...
    },
    subaddress OCTET STRING (SIZE(1..20)) OPTIONAL,
    ...
}

V75Parameters ::= SEQUENCE
{
    audioHeaderPresent    BOOLEAN,
    ...
}

DataType ::= CHOICE
{
    nonStandard           NonStandardParameter,
    nullData              NULL,
    videoData             VideoCapability,
    audioData             AudioCapability,

```

```

data
encryptionData
...
h235Control
h235Media
multiplexedStream
redundancyEncoding
multiplePayloadStream
depFec
fec
}

H235Media ::= SEQUENCE
{
    encryptionAuthenticationAndIntegrity
mediaType
{
    nonStandard
videoData
audioData
data
...
redundancyEncoding
multiplePayloadStream
depFec
fec
},
...
}

MultiplexedStreamParameter ::= SEQUENCE
{
    multiplexFormat
controlOnMuxStream
...
}

H222LogicalChannelParameters ::= SEQUENCE
{
    resourceID
subChannelID
pcr-pid
programDescriptors
streamDescriptors
...
}

H223LogicalChannelParameters ::= SEQUENCE
{
    adaptationLayerType
{
    nonStandard
allFramed
allNotFramed
al2WithoutSequenceNumbers
al2WithSequenceNumbers
al3
{
    controlFieldOctets
sendBufferSize
},
...
}

```

DataApplicationCapability,  
EncryptionMode,  
NonStandardParameter,  
H235Media,  
MultiplexedStreamParameter,  
RedundancyEncoding,  
MultiplePayloadStream,  
DepFECData, -- Deprecated, do not use  
FECData  
EncryptionAuthenticationAndIntegrity,  
CHOICE  
NonStandardParameter,  
VideoCapability,  
AudioCapability,  
DataApplicationCapability,  
RedundancyEncoding,  
MultiplePayloadStream,  
DepFECData, -- Deprecated, do not use  
FECData  
MultiplexFormat,  
BOOLEAN,  
INTEGER (0..65535),  
INTEGER (0..8191),  
INTEGER (0..8191) OPTIONAL,  
OCTET STRING OPTIONAL,  
OCTET STRING OPTIONAL,  
CHOICE  
NonStandardParameter,  
NULL,  
NULL,  
NULL,  
NULL,  
SEQUENCE  
INTEGER (0..2),  
INTEGER (0..16777215) -- units octets

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>         al1M         al2M         al3M     },     segmentableFlag     ... } H223AL1MParameters ::= SEQUENCE {     transferMode     {         framed         unframed         ...     },     headerFEC     {         sebch16-7         golay24-12         ...     },     crcLength     {         crc4bit         crc12bit         crc20bit         crc28bit         ...,         crc8bit         crc16bit         crc32bit         crcNotUsed     },     rcpcCodeRate     INTEGER (8..32),     arqType     {         noArq         typeIArq         typeIIArq         ...     },     alpduInterleaving     alsduSplitting     ...,     rsCodeCorrection     INTEGER (0..127) OPTIONAL } H223AL2MParameters ::= SEQUENCE {     headerFEC     {         sebch16-5         golay24-12         ...     },     alpduInterleaving     ... } </pre> | <pre> H223AL1MParameters, H223AL2MParameters, H223AL3MParameters  BOOLEAN,  CHOICE     NULL,     NULL,     ... CHOICE     NULL,     NULL,     ... CHOICE     NULL,     NULL,     NULL,     NULL,     NULL,     NULL,     NULL,     NULL INTEGER (8..32), CHOICE     NULL,     H223AnnexCArqParameters,     H223AnnexCArqParameters,     ... BOOLEAN, BOOLEAN, INTEGER (0..127) OPTIONAL  CHOICE     NULL,     NULL,     ... BOOLEAN, </pre> |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|