INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 1997 - 2000

COM 16-R 54-E
July 1999
Original: English

Questions: 1, 2, 3, 11-14/16

Texte disponible seulement en
Text available only in
Texto disponible solamente en
E

STUDY GROUP 16 – REPORT R 54

SOURCE*: STUDY GROUP 16 (SANTIAGO MEETING, 17-28 MAY 1999)

TITLE: PART II.B (IMPLEMENTOR'S GUIDES) OF THE REPORT OF WP 2/16 (SERVICES AND HIGH RATE SYSTEMS)

_____

CONTENTS

PART II.B - IMPLEMENTOR'S GUIDES OF WP 2/16

* **Contact:** TSB

Tel: +41 22 730 5860
Fax: +41 22 730 5853
email: bigi@itu.int

1 Implementor's Guide to T.120-series

This document represents the complete, current text for the T.120-series of ITU-T Recommendations. The following changes have been made:

– Section headings have been added for all standards for which Q.3/16 has responsibility.

– Section 6.8.6.1 was added. These changes provide support for Unicode coding of conference names and passwords in GCC.

– Section 6.8.9.1 was added. These changes reflect the T.127 editor's errata that had not previously been documented in this Implementor's Guide.

6 Implementor's Guide for the ITU-T T.120 Recommendation series - Data Protocols for Multimedia Conferencing

6.1 Abstract

This document is a compilation of reported defects identified with the 1993-1999 editions of the ITU-T T.120-series Recommendations. It is intended to be read in conjunction with the Recommendations to serve as an additional authoritative source of information for implementors. The changes, clarifications and corrections defined herein are expected to be included in future versions of affected T.120-series Recommendations.

6.2 Document History

Version	Date	Description
1	15 February 1996	Initial version - completed at the ITU-T Study Group 8 meeting, Geneva, 6-15 February 1996
2	26 April 1996	Updated at the Q.10/8 Rapporteur's meeting, Boston, 21-26 April 1996 Sections added: 7.5.4 and 7.5.5
3	9 August 1996	Updated at the Q.10/8 Rapporteur's meeting, Santa Rosa, 5-9 August 1996 Sections added: 7.5.6
4	4 October 1996	Updated at the Q.10/8 Rapporteur's meeting, Ismaning, 30 September - 4 October 1996
5	17 January 1997	Updated at the Q.10/8 Rapporteur's meeting, Newport Beach, 13-17 January 1997
6	29 January 1998	Updated at the ITU-T SG 16 meeting, Geneva, 26 January - 6 February 1998 Sections added: 6.7.1.1
7	20 May 1999	Updated at the ITU-T SG 16 meeting, Santiago, 17-28 May 1999 Added sections 6.8.6.1 and 6.8.9.1

6.3 Scope

This Guide resolves defects in the following categories:

- Editorial errors
- Technical errors such as omissions or inconsistencies
- Ambiguities.

In addition, the Guide may include explanatory text found necessary because of interpretation difficulties apparent from the defect reports.

This Guide will not address proposed additions, deletions, or modifications to the Recommendations that are not strictly related to implementation difficulties in the above categories. Proposals for new features should be made in the normal way through contributions to the ITU-T.

6.4 Policies for updating this document

This document is managed by the ITU-T Study Group 16 Question 3 Rapporteur's Group. It can be revised at any recognized Q.3/16 Rapporteur's Group meeting provided the proposed revisions are unanimously accepted by the members of the group. A revision history cataloguing the evolution of this document is included.

6.5 Defect Resolution Procedure

Upon discovering technical defects with any components of the T.120 Recommendations series, please provide a written description directly to the Q.3/16 Rapporteur.

6.6 References

This document refers to the following T.120-series Recommendations:

- ITU-T Recommendation T.120 (07/96) - *Data Protocols for Multimedia Conferencing.*
- ITU-T Recommendation T.120, Annex C (02/98) - *Lightweight Profiles for the T.120 Architecture*.
- ITU-T Recommendation T.121 (07/96) - *Generic Application Template.*
- ITU-T Recommendation T.122 (02/98) - *Multipoint Communication Service - Service Definition*.
- ITU-T Recommendation T.123 (05/99) - *Network Specific Data Protocol Stacks for Multimedia Conferencing.*
- ITU-T Recommendation T.124 (02/98) - *Generic Conference Control*.
- ITU-T Recommendation T.125 (02/98) - *Multipoint Communication Service - Protocol Specification.*
- ITU-T Recommendation T.126 (07/97) - *Multipoint Still Image and Annotation Protocol.*
- ITU-T Recommendation T.127 (08/95) - *Multipoint Binary File Transfer Protocol.*
- ITU-T Recommendation T.128 (02/98) - *Multipoint Application Sharing.*
- ITU-T Recommendation T.134 (02/98) - *Text Chat Application Entity*.
- ITU-T Recommendation T.135 (02/98) - *User-To-Reservation System Transactions within T.120 Conferences.*
- ITU-T Recommendation T.136 (05/99) - *Remote Device Control Application Protocol*.
- ITU-T Recommendation H.282 (05/99)- *A Far End Camera Control Protocol for Video Conferencing Using H.224.*

6.7 Nomenclature

In addition to traditional revision marks, the following marks and symbols are used to indicate to the reader how changes to the text of a Recommendation should be applied:

Symbol	Description
[Begin Correction]	Identifies the start of revision marked text based on extractions from the published Recommendations affected by the correction being described.
[End Correction]	Identifies the end of revision marked text based on extractions from the published Recommendations affected by the correction being described.
...	Indicates that the portion of the Recommendation between the text appearing before and after this symbol has remained unaffected by the correction being described and has been omitted for brevity.
--- SPECIAL INSTRUCTIONS --- {instructions}	Indicates a set of special editing instructions to be followed.

6.8 Technical and Editorial Corrections

6.8.1 Technical and Editorial Corrections to ITU-T Recommendation T.120

6.8.1.1 Addition of new MCS static channel assignments

Description: T.120 Annex A1.1 contains a table of MCS Static Channel ID numbers. Two new Application Protocol Entities have been defined, each with a new Channel ID. The table is updated to add Channel IDs 11 and 12.

[Begin Correction]

A1.1 Static Channels

Symbolic Name	MCS Channel ID	Description	Recommendation
GCC-CHANNEL-0	1	GCC Broadcast Channel	T.124
GCC-CHANNEL-1	2	Convenor Channel	T.124
SI-CHANNEL-0	8	MSIA Communications Channel	T.126
MBFT-CHANNEL-0	9	Control Channel	T.127
MBFT-CHANNEL-1	10	Data Channel	T.127
AS-CHANNEL-0	11	Application Sharing Channel	T.128
CHAT-CHANNEL-0	12	Chat Data Channel	T.134

[End Correction]

6.8.2 Technical and Editorial Corrections to ITU-T Recommendation T.120, Annex C

-None-

6.8.3 Technical and Editorial Corrections to ITU-T Recommendation T.121

-None-

6.8.4 Technical and Editorial Corrections to ITU-T Recommendation T.122

-None-

6.8.5 Technical and Editorial Corrections to ITU-T Recommendation T.123

-None-

6.8.6 Technical and Editorial Corrections to ITU-T Recommendation T.124

6.8.6.1 Addition of Unicode Support for Conference Names and Passwords

Description: In order to better support these users of Unicode languages the definitions of ConferenceName and Password shall be modified with additional optional fields past the extension markers. These fields will allow the full support of Unicode strings for the identification of conference names and for conference passwords. The following are the proposed modifications to the ASN.1.

[Begin Correction]

```
ConferenceName ::= SEQUENCE
{
        numeric             SimpleNumericString,
        text                SimpleTextString OPTIONAL,
        ...
        unicode_text        TextString OPTIONAL
}

ConferenceNameSelector ::= CHOICE
{
        numeric             SimpleNumericString,
        text                SimpleTextString,
        ...
        unicode_text        TextString
}

Password ::= SEQUENCE
{
        numeric             SimpleNumericString,
        text                SimpleTextString OPTIONAL,
        ...
        unicode_text        TextString OPTIONAL
}

PasswordSelector ::= CHOICE
{
        numeric             SimpleNumericString,
        text                SimpleTextString,
        ...
        unicode_text        TextString
}
```

[End Correction]

6.8.7 Technical and Editorial Corrections to ITU-T Recommendation T.125

-None-

6.8.8 Technical and Editorial Corrections to ITU-T Recommendation T.126

-None-

6.8.9 Technical and Editorial Corrections to ITU-T Recommendation T.127

6.8.9.1 ASN.1 Errata

Description: The following additions to the ASN.1 for the File-OfferPDU were omitted from the published text, as documented in the editor's errata sheet issued at the Paris Rapporteur meeting of 10/95.

The ASN.1 text shown below shall be added.

[Begin Correction]

```
File-OfferPDU ::= SEQUENCE
{
        file-header             FileHeader,
        data-channel-id         ChannelID,
        file-handle             Handle,
        roster-instance         INTEGER (0..65535) OPTIONAL,
        file-transmit-token     TokenID OPTIONAL,
        file-request-token      TokenID OPTIONAL,
        file-handle             Handle OPTIONAL,
        mbftID                  UserID OPTIONAL
        compression-specifier   CompressionSpecifier OPTIONAL,
        compressed-filesize     INTEGER OPTIONAL,
        ack-flag                BOOLEAN,            -- True if acknowledgements
                                                    -- required
        ...,
}
```

[End Correction]

6.8.10 Technical and Editorial Corrections to ITU-T Recommendation T.128

-None-

6.8.11 Technical and Editorial Corrections to ITU-T Recommendation T.134

-None-

6.8.12 Technical and Editorial Corrections to ITU-T Recommendation T.135

-None-

6.8.13 Technical and Editorial Corrections to ITU-T Recommendation T.136

-None-

6.8.14 Technical and Editorial Corrections to ITU-T Recommendation H.282

-None-

2 Implementor's Guide for the ITU-T H.223 Recommendation series - Multiplexing protocol for low bit rate Multimedia Communication

VERSION: 2

STATUS: To be discussed at the ITU-T SG 16 meeting in Santiago

DATE: May 1999

ABSTRACT: This document is a compilation of reported defects identified with the 1997-2000 editions of the ITU-T H.223-series Recommendations. It is intended to be read in conjunction with the Recommendations to serve as an additional authoritative source of information for implementors. The changes, clarifications and corrections defined herein are expected to be included in future versions of affected H.223-series Recommendations.

Contact Information

ITU-T Study Group 16/Question 11 Rapporteur	Tom Geary Conexant Systems, Inc. 4311 Jamboree Road, MC 510-350 Newport Beach, CA 92660-3095 USA	Tel: +1.949 483.4092 Fax: +1.949 483.6511 Email: tom.geary@conexant.com
Implementor's Guide Co-Editor	Toshiro Kawahara NTT Mobile Communications Network, Inc. 3-5 Hikarinooka, Yokosuka Kanagawa 239-8536 Japan	Tel: +81.468.40.3518 Fax: +81.468.40.3788 Email: kawahara@spg.yrp.nttdocomo.co.jp
Implementor's Guide Co-Editor	Bernhard G. Wimmer SIEMENS AG ZT IK 2 Otto-Hahn-Ring 6 81730 Munich Germany	Tel: +49.89.636.50417 Fax: +49.89.636.52393 Email: Bernhard.Wimmer@ties.itu.int
Implementor's Guide Co-Editor	Takashi Suzuki NTT Mobile Communications Network, Inc. 3-5 Hikarinooka, Yokosuka Kanagawa 239-8536 Japan	Tel: +81.468.40.3515 Fax: +81.468.40.3788 Email: suzuki@spg.yrp.nttdocomo.co.jp

Document history

Revision	Date	Description
1	22 September 1998	Initial version and approved
2	26 May 1999	Second version and approved

2 Introduction

This document is a compilation of reported defects identified with the 1997-2000 editions of the ITU-T H.223-series Recommendations. It is intended to be read in conjunction with the Recommendations to serve as an additional authoritative source of information for implementors. The changes, clarifications and corrections defined herein are expected to be included in future versions of affected H.223-series Recommendations.

The first version of the guide was produced following the September 1998 ITU-T Study Group 16 meeting. Wide distribution of this document is expected and encouraged.

3 Scope

This guide resolves defects in the following categories:

- editorial errors;
- technical errors such as omissions or inconsistencies;
- ambiguities.

In addition the Guide may include explanatory text found necessary as a result of interpretation difficulties apparent from the defect reports.

This Guide will not address proposed additions, deletions or modifications to the Recommendations that are not strictly related to implementation difficulties in the above categories. Proposals for new features should be made in the normal way through contributions to the ITU-T.

4 Policies for updating this document

This document is managed by the ITU-T Study Group 16 Question 11 Rapporteur's Group. It can be revised at any recognized Q.11/16 Rapporteur's Group meeting provided the proposed revisions are unanimously accepted by the members of the group. A revision history cataloguing the evolution of this document is included.

5 Defect resolution procedure

Upon discovering technical defects with any components of the H.223 Recommendations series, please provide a written description directly to the editors of the affected Recommendations with a copy to the Q.11/16 Rapporteur. The template for a defect report is enclosed. Contact information for these parties is included in this document. Return contact information should also be supplied so a dialogue can be established to resolve the matter and an appropriate reply to the defect report can be conveyed. This defect resolution process is open to anyone interested in H.223-series Recommendations. Formal membership in the ITU is not required to participate in this process.

6 References

This document refers to the following H.223-series Recommendations:

– ITU-T Recommendation H.223 (1996), *Multiplexing Protocol for low bit-rate Multimedia Communication.*

– ITU-T Recommendation H.223/Annex A (1998), *Multiplexing Protocol for low bit-rate Multimedia Communication over low error-prone Channels.*

– ITU-T Recommendation H.223/Annex B (1998), *Multiplexing Protocol for low bit-rate Multimedia Communication over moderate error-prone Channels.*

– ITU-T Recommendation H.223/Annex C (1998), *Multiplexing Protocol for low bit-rate Multimedia Communication over highly error-prone channels.*

7 Nomenclature

In addition to traditional revision marks, the following marks and symbols are used to indicate to the reader how changes to the text of a Recommendation should be applied:

Symbol	Description
[Begin Correction]	Identifies the start of revision marked text based on extractions from the published Recommendations affected by the correction being described.
[End Correction]	Identifies the end of revision marked text based on extractions from the published Recommendations affected by the correction being described.
...	Indicates that the portion of the Recommendation between the text appearing before and after this symbol has remained unaffected by the correction being described and has been omitted for brevity.
--- SPECIAL INSTRUCTIONS --- {instructions}	Indicates a set of special editing instructions to be followed.

8 Technical and editorial corrections

8.1 Replacement in Section C.4.1.4

Description:	Clarify that tail bits are also required in FEC_ONLY mode.

[Begin Correction]

FEC_ONLY — In this mode a AL-SDU* with ~~an~~mandatory tails bits (TB)[1] and CRC is RCPC encoded with a code rate $r \leq 1.0$. The resulting AL-PDU only consists of an AL-PDU payload field. Splitting mode is not supported.

[End Correction]

8.2 Replacement in Section C.4.1.7.2

Description:	The CRC is appended to AL-SDU* not to AL-PDU, as seen in Figure C.2/H.223.

[Begin Correction]

C.4.1.7.2 Cyclic redundancy check (CRC)

The CRC provides error detection capability ~~across the entire AL-SDU*~~. The CRC is appended to the AL-~~PDU~~SDU* before the error correction coding procedure is done. The CRC is used by the AL1M receiver to verify whether the decoding procedure of the error correction algorithm is error-free. CRC lengths of 4, 12, 20 and 28 bits are supported. The length of the CRC field shall be specified during the H.245 OpenLogicalChannel procedure. The evaluation of the CRC shall be performed by the same procedure as described as in 7.3.3.2.3 of Recommendation H.223.

[End Correction]

8.3 Replacement in Section C.4.1.8

Description:	The wording interleaver may be misleading. Therefore a more particular definition is required. In this chapter the wording is corrected, in chapter 8.2 an enhanced description is provided. In addition to that, the information about the range of the value *b* is not correct. So it is deleted.

[Begin Correction]

C.4.1.8 Interleaving

For some channels block interleaving may be used.

If interleaving is used, it shall be a applied to the entire AL-PDU including the control field. As the length of the AL-PDU varies, the dimension of the block interleaver matrix has to be recalculated for each length. Given a AL-PDU of length l_v, the dimensions, the width *a* and the height *b* of the block interleaver can be calculated:

$$a = \max_{\alpha \in \Im,\ l_v \bmod \alpha = 0} \left\{ \alpha \le \sqrt{l_v} \right\}, \quad \text{with } \Im \text{ all integers}$$

$$b = l_v \,/\, a$$

b describes the distance between two before interleaving consecutive bits after interleaving. ~~As the AL-PDU is an integer number of octets, the minimum *b* is *8*.~~

The receiver shall calculate the dimensions of the interleaver with the upper equation and the length of the received AL-PDU l_v. Deinterleaving shall also be applied to the entire AL-PDU.

[End Correction]

8.4 Replacement in Section C.4.1.9 Item 1

Description:	The parameter *lp* is not defined. Therefore it shall be replaced by a defined parameters.

[Begin Correction]

1) Calculate the length of the AL-PDU ~~payload~~ l_v~~l_p~~ according to Section C.4.1.7.1 and the first rate required in the H.245 OpenLogicalChannel message.

[End Correction]

8.5 Replacement in Section C.4.1.9 Item 6

Description:	The parameter *lp* is not defined. Therefore it shall be replaced by a defined parameters.

[Begin Correction]

6) For the first transmission read l_v - l_h~~l_p~~ (AL-PDU payload length) bits from the buffer, starting from the beginning of the buffer and~~,~~ fill these bits into the AL-PDU payload field. The first octet of the buffer is the first octet of the AL-PDU payload field.

[End Correction]

8.6 Replacement to Section C.4.1.9 Item 7

Description:	Clarification of the appropriate H.245 message.

[Begin Correction]

7) ~~If required in the H.245 OpenLogicalChannel message, the Control Field (CF) shall be added at the beginning of the AL-PDU.~~The Control Field (CF) shall not be used if the ARQ mode, signalled by the H.245 message, is set to "noArq".

[End Correction]

8.7 Replacement to Section C.4.1.9

Description:	The old description is not precise enough for an implementation. Therefore this replacement is done for a sufficient description.

[Begin Correction]

ARQII ~~The transmitting entity shall first transmit the first code rate according to the H.245 OpenLogicalChannel message and may choose any AL-PDU payload length. for following incremental retransmissions.~~If $V^{j}(S) = 0$, the encoding procedure step 6 of this chapter shall be performed. Otherwise, the transmitter may choose any AL-PDU payload length, whereby the AL-PDU payload length shall be integral number of octets. This AL-PDU payload shall be read in the consecutive order from the linear buffer.

However if the mother code rate is reached, the transmitter begins transmitting at the beginning of the linear buffer and is still free to choose the code rate, if the maximum number of retransmissions is not reached.

Figure C.6/H.223C illustrates the encoding procedures of the AL1M at the transmit side.

[End Correction]

8.8 Correction of error in Figure C.6/H.223

Description:	Replace Figure C.6/H.223 due to errors of two of the indexes. Also the caption refers to the wrong AL layer.

[Begin Correction]

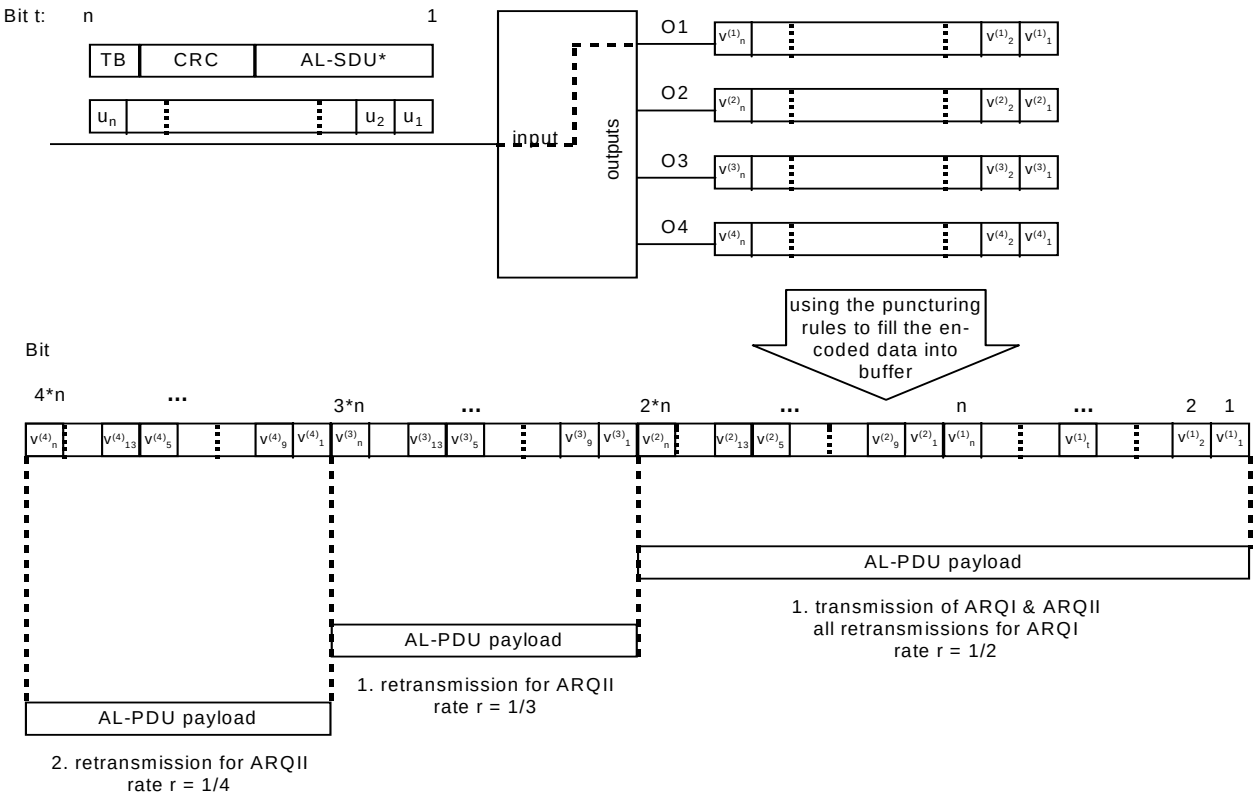


FIGURE C.6/H.223

Encoding procedure of the AL~~3~~1M at the transmitter side

[End Correction]

8.9 Correction of error in Section C.4.1.13

Description:	In Section C.4.1.13.1, there is a description on the range of sequence number field, but the maximum value is not correct. The maximum value is 2^5-1 or 2^{10}-1 for 5-bit or 10-bit field, respectively.

[Begin Correction]

C.4.1.13.1 Definitions

a) Modulo

Each AL-PDU Payload is sequentially numbered modulo 2^5 or 2^{10} and may have the value 0 through 2^5-1 or 2^{10}-1. The length of the sequence number field (SN) is set with the OpenLogicalChannel message of Recommendation H.245.

NOTE - All arithmetic operations on state variables and sequence numbers contained in this section are modulo 2^5 or 2^{10}.

…

[End Correction]

8.10 Correction of error in Section C.4.1.13.6

Description:	The 1-bit receive retransmission number in the CF is RN, as described in Section C.4.1.5.2.

[Begin Correction]

C.4.1.13.6 Receiving SREJ-PDUs

On receipt of a valid SREJ-PDU, the AL1M entity shall act as follows:

a) If the I-PDU, whose N(S) is equal to the N(R) of the SREJ message is still in the send buffer, the AL1M entity shall pass a corresponding AL-PDU to the MUX layer as soon as possible.

When ARQI error protection is used the same AL-PDU payload shall be used for re-transmission.

When ARQII is used, the parity of the send retransmission variable $V^j(S)$ is checked against the 1-bit receive retransmission number ~~N(R)~~RN. If the parity differs, $V^j(S)$ will be decremented by 1. Then the next I-PDU payload, according to the procedure described in C.4.1.9, shall be re-transmitted to the receiver.

No other previously transmitted I-PDUs shall be retransmitted as a result of receiving the SREJ-PDU.

…

[End Correction]

8.11 Correction of error in Section C.4.1.13.8

Description:	In the original text, it is not explicitly stated whether the exception conditions are cleared or not, in case the retransmission I-PDU with N(S) doesn't equal to V(R), while it implicitly says that they are cleared. This change is in order to clarify that.

[Begin Correction]

C.4.1.13.8 Exception condition reporting and recovery

Exception conditions may occur as a result of errors on the physical connection or procedural errors by an AL1M entity.

The error-recovery procedures that are available following the detection of an exception condition by an AL1M entity are defined in this subsection.

a) Receiving invalid AL-PDUs

When a received AL-PDU is invalid, it is either discarded or saved for possible delivery later to the AL1 user.

b) N(S) sequence error

When there are no other outstanding exception conditions, an N(S) sequence error exception condition occurs in the receiving AL1M entity when a valid I-PDU is received containing an N(S) value that is not equal to the V(R) at the receiver. In this case, V(R) shall not be incremented, and one or more SREJ-PDUs, each containing a different N(R), may be transmitted by the AL1M receiving entity to initiate an exception condition recovery for each SREJ-PDU. After passing each SREJ-PDU to the MUX layer, the AL1M entity shall start a local timer. Several factors that affect the length of the timer are given in Appendix IV/V.42. A different timer is maintained for each outstanding SREJ-PDU. Successive SREJ-PDUs are transmitted in the order indicated by their N(R) field.

For each SREJ-PDU that it transmits, the AL1M receiver may pass an empty AL-SDU or an invalid received AL-SDU (previously saved), with an appropriate EI parameter, to the AL1 user via the AL-DATA.indication primitive.

When the retransmitted I-PDU with $N(S) = V(R)$ is received, the exception condition for that I-PDU shall be cleared. The AL1M receiver should pass the associated AL-SDU, together with an appropriate EI parameter, to the AL1 user via the AL-DATA.indication primitive. When the exception condition is cleared, the associated timer shall be stopped and V(R) shall be incremented as many times as necessary so that V(R) represents the send sequence number of the next expected in-sequence I-PDU.

When a retransmitted I-PDU with $N(S) \neq V(R)$ is received, the AL1M receiving unit shall ~~stop the timers associated with~~clear all exception conditions related to previously sent SREJ-PDUs for which retransmission is received, by stopping the associated timers. For each exception condition cleared, the AL1M receiver shall increment V(R) by 1, and may deliver an empty AL-SDU, together with an appropriate EI parameter, to the AL1 user via the AL-DATA.indication primitive, prior to delivering the AL-SDU associated with the received I-PDU.

The information in all other received valid I-PDUs should be delivered to the AL1 user in AL-SDUs, together with an appropriate EI parameter.

…

[End Correction]

8.12 Corrections in Appendix I

Description:	The calculation in the example is wrong.

[Begin Correction]

APPENDIX I

(to Annex C to H.223)

Generator matrixes of the systematic extended BCH

This appendix describes Systematic Extended Bose-Chaudhuri-Hocquenghem (SEBCH) codes and includes the generator matrixes, which are used by the Recommendation H.223/Annex C.

I.1 BCH codes

BCH codes are linear cyclic block codes, hence they can be described using a generator polynomial. However, the easiest way to describe short block codes is using a generator matrix which describes all characteristics of the code. With a generator matrix $\underline{\boldsymbol{G}}$ and a information sequence $\underline{\boldsymbol{i}}$ of length $\boldsymbol{k}$ the code vector $\underline{\boldsymbol{c}}$ of length $\boldsymbol{n}$ can be obtained by:

$$\underline{\mathbf{c}} = \underline{\mathbf{i}} \cdot \underline{\mathbf{G}} = [\ \underline{\mathbf{i}}^{\mathbf{T}} \mid \underline{\mathbf{c}}_{\mathbf{0}}{}^{\mathbf{T}}\ ]^{\mathbf{T}}$$

with $\underline{\boldsymbol{G}} = [\ \underline{\boldsymbol{1}} \mid \underline{\boldsymbol{A}}\ ]$ a $(\boldsymbol{k}\times\boldsymbol{n})$ matrix containing a $(\boldsymbol{k}\times\boldsymbol{k})$ identity matrix in the first $\boldsymbol{k}$ columns/rows to obtain a systematic code. For a primitive BCH code the length of the code $\boldsymbol{n}$ is always $\boldsymbol{n = 2^h - 1}$. For $\boldsymbol{k}$ there are some constraints, not all values are possible.

The third parameter describing a block code besides code length $\boldsymbol{n}$ and information length $\boldsymbol{k}$ is the minimum distance between two code words $\boldsymbol{d}$. If a code has minimal distance $\boldsymbol{d}$, it can correct at most $\boldsymbol{\lfloor (d-1)/2 \rfloor}$ errors or detect $\boldsymbol{(d-1)}$ errors.

I.2 Systematic extended BCH codes

As all linear cyclic block codes can be made systematic, there always exits a systematic BCH code.

As we evaluated earlier, primitive BCH codes always have the length $\boldsymbol{n=2^h-1}$. To make these codes octet aligned, extension has to be applied. The extension of a BCH($\boldsymbol{n, k, d}$) has the length $\boldsymbol{n+1}$. One digit is appended, so that each code word has even weight. The extended BCH code then always has minimal distance $\boldsymbol{d+1}$. Hence we derived from BCH($\boldsymbol{n, k, d}$) a code EXBCH($\boldsymbol{n+1, k, d+1}$). Extended codes are still linear, but no more cyclic. Hence the description using generator polynomials is impossible.

The generator matrix of the extended code from $\underline{\boldsymbol{G}}$ of the mother code can be derived by adding one column which contains the parity check bit of each row. The examples of the generator matrices ~~of the codes used in this proposal~~ are given in Table I.1 and I.2.

I.3 Decoder overview

For decoding BCH codes, usually Berleykamp-Massey algorithm is used. This is an efficient method to determine error locations in the received vector. There are also some approaches to use reliability information for decoding block codes. However, these algorithms yield in high complexity.

One main feature of BCH codes is the possibility to use these codes for error correction and detection at the same time. For example a code with ***d=5*** could correct up to 1 error and detect up to 3 errors in parallel. With the usage of BCH codes only, the decoder has the flexibility to decide how many errors to correct and use the rest of redundancy for error detection. Berleykamp-Massey algorithm can also be used for this.

I.4 Example

In this example we use the SEBCH(16,5,8). The information vector ~~***c***~~ ***i*** is given as:

~~***c***~~ ***i = [1 0 0 1 1]***

By using the generator matrix ***G*** the code word ***c*** can be evaluated by:

c = ***i***·***G*** ***= [1 0 0 1 1 0 1 0 1 1 1 1*** ~~***1***~~ ***0 0 0 0]***

For transmission these bits are filled into octet-aligned fields. The LSB-bit of the vector ***c*** is at its left side, the MSB at its right. The LSB of ***c*** is filled to the lowest numbered bit of the last octet (octet 2) and the MSB of ***c*** to the highest-numbered bit of the first octet (octet 1), see Figure I.1.

Bit:	8	7	6	5	4	3	2	1	Octet
	0	0	0	~~1~~0	1	1	1	1	1
	0	1	0	1	1	0	0	1	2

FIGURE I.1/H.223

Field mapping convention of SEBCH-codes

…

[End Correction]

8.13 Replacement to Section B.3.2.1

Description:	The old description is not precise enough for an implementation. Therefore this replacement is done for a sufficient description.

[Begin Correction]

An optional header for this Annex provides the capability to use the previous MUX-PDU whose header is corrupted due to channel errors. Figure B.3 shows the format of the MUX-PDU when using this option, and Figure B.4 shows the format of the optional header. ~~The optional header contains the packed master signalling and multiplex code of the previous MUX-PDU.~~ The optional header contains the packet marker signalling and multiplex code of the previous header. This header belongs either to a non-stuffing MUX-PDU or a stuffing sequence.

The values of MC' and PM' are as indicated by MC and PM, respectively, in H.223/Level 0. The HEC' field shall be calculated from MC' according to the procedure described in 6.4.1.2/H.223.

[End Correction]

8.14 Replacement to Section B.3.2.3

Description:	The old description is not precise enough for an implementation. Therefore this replacement is done for a sufficient description.

[Begin Correction]

If no information is available the stuffing mode shall be used. The multiplexer shall indicate a Level 2 stuffing mode by inserting a Level 2 synchronization flag followed by a Level 2 header (either the normal Level 2 header or the optional header in Section B.3.2.1, depending on the mode of operation). The MPL field shall be "00000000" and the MC shall be "0000". If the optional header is used in the stuffing operation mode, the optional header contains the packet marker signalling and multiplex code of the previous header. This header belongs either to a non-stuffing MUX-PDU or a stuffing sequence. This stuffing mode may be inserted consecutively an arbitrary number of times.

[End Correction]

9 Implementation clarifications

9.1 Clarification of the mapping procedure of Figure C.7/H.223 of H.223/Annex C

General

The mapping from the temporary matrix to the linear buffer is done by the rules of the puncturing Table C.4/H.223 that describes the exact reading order from the temporary matrix. Table 1 reflects that reading order for the output 2, 3 and 4.

TABLE 1

Reading order for the output 2, 3 and 4 of the temporary matrix of Figure C.7/H.223

column number	1	2	3	4	5	6	7	8
reading order	1	5	3	7	2	6	4	8

Mapping Procedure

The linear buffer is filled in the following way:

1) The first output line of the convolutional encoder is directly written to the linear buffer.

2) The columns of output 2 of the temporary matrix are written to the linear buffer by the use of Table 1. Thus first all the bits in column 1 are read from the top to down and filled to the linear buffer, followed by column 5 and so on. After all columns are read the mapping procedure continues with output 3.

3) The columns of output 3 of the temporary matrix are written to the linear buffer by the use of Table 1. Therefore first all the bits in column 1 are read from the top to down and filled to the linear buffer, followed by column 5 and so on. After all columns are read the mapping procedure continues with output 4.

4) The columns of output 4 of the temporary matrix are written to the linear buffer by the use of Table 1. Therefore first all the bits in column 1 are read from the top to down and filled to the linear buffer, followed by column 5 and so on. After all columns are read the mapping procedure is finished.

9.2 Clarification of the interleaving procedure of chapter C.4.1.8 of H.223/Annex C

The process of block interleaving with the width a and the height b is as follows:

1) Prepare a rectangular buffer with a columns and b rows.

2) The input data is written in to the buffer from the top left to the bottom right, row by row, bit by bit.

3) The output data is read out from the buffer from the top left to the bottom right, column by column, bit by bit.

This is represented with a formula as follows:

x_i: i-th input bit to the interleaver. i=0..N-1,

y_j: j-th output bit from the interleaver. j=0..N-1,

$y_j = x_i$,where $i = (j \bmod b) \cdot a + \lceil j/b \rceil$

N is the number of bits input to the interleaver, and $\lceil x \rceil$ is the maximum integer value which is smaller than or equal to x.

H.223 RECOMMENDATION SERIES DEFECT REPORT FORM

DATE:	
CONTACT INFORMATION **NAME:** **COMPANY:** **ADDRESS:** **TEL:** **FAX:** **EMAIL:**	
AFFECTED RECOMMENDATIONS:	
DESCRIPTION OF PROBLEM:	
SUGGESTIONS FOR RESOLUTION:	

NOTE - Attach additional pages if more space is required than is provided above.

3 Implementor's Guide for the ITU-T H.324 Recommendation series Version 2 - Terminal for low bit-rate multimedia communication

Abstract

This document is a compilation of reported defects identified with the 1997-2000 editions of the ITU-T H.324-series Recommendations. It is intended to be read in conjunction with the Recommendations to serve as an additional authoritative source of information for implementors. The changes, clarifications and corrections defined herein are expected to be included in future versions of affected H.324-series Recommendations.

Contact information

ITU-T Study Group 16/Question 11 Rapporteur	Tom Geary Rockwell Semiconductor Systems 4311 Jamboree Road, MC 510-350 Newport Beach, CA 92660-3095 United States	Tel: +1 714 221 4092 Fax: +1 714 221 6511 Email: tom.geary@rss.rockwell.com
Implementor's Guide Editor	Cor Quist KPN Research P.O. Box 421 2260 AK Leidschendam Netherlands	Tel: +31 70 332 4005 Fax: +31 70 332 6477 Email: C.P.Quist@research.kpn.com
ITU-T Recommendation H.324 Editor	Mickey Nasiri Ericsson Telecom S-125 25 Stockholm Sweden	Tel: +46 8 726 2125 Fax: +46 8 18 7620 Email: mickey@clab.ericsson.se

Document history

Revision	Date	Description
1	8-11 June 1998	Initial version - Reviewed at the Q.11/SG 16 meeting.
2	22 September 1998	Second version - Completed at the ITU-T Study Group 16 Rapporteurs meeting.
3	24 May 1999	Third version - Completed at the ITU-T Study Group 16 Rapporteurs meeting.

Introduction

This document is a compilation of reported defects identified with the 1997-2000 editions of the ITU-T H.324-series Recommendations. It is intended to be read in conjunction with the Recommendations to serve as an additional authoritative source of information for implementors. The changes, clarifications and corrections defined herein are expected to be included in future versions of affected H.324-series Recommendations.

The first version of the guide was produced following the September 1998 ITU-T Study Group 16 meeting. Wide distribution of this document is expected and encouraged.

Scope

This guide resolves defects in the following categories:

- editorial errors;
- technical errors such as omissions or inconsistencies;
- ambiguities.

In addition the Guide may include explanatory text found necessary as a result of interpretation difficulties apparent from the defect reports.

This Guide will not address proposed additions, deletions or modifications to the Recommendations that are not strictly related to implementation difficulties in the above categories. Proposals for new features should be made in the normal way through contributions to the ITU-T.

Policies for updating this document

This document is managed by the ITU-T Study Group 16 Question 11 Rapporteur's Group. It can be revised at any recognized Q.11/16 Rapporteur's Group meeting provided the proposed revisions are unanimously accepted by the members of the group. A revision history cataloguing the evolution of this document is included.

Defect resolution procedure

Upon discovering technical defects with any components of the H.324 Recommendations series, please provide a written description directly to the editors of the affected Recommendations with a copy to the Q.11/16 Rapporteur. The template for a defect report is enclosed. Contact information for these parties is included in this document. Return contact information should also be supplied so a dialogue can be established to resolve the matter and an appropriate reply to the defect report can be conveyed. This defect resolution process is open to anyone interested in H.324-series Recommendations. Formal membership in the ITU is not required to participate in this process.

References

This document refers to the following H.324-series Recommendations:

– ITU-T Recommendation H.324 (1998), *Terminal for low bit-rate Multimedia Communication.*

Nomenclature

In addition to traditional revision marks, the following marks and symbols are used to indicate to the reader how changes to the text of a Recommendation should be applied:

Symbol	Description
[Begin Correction]	Identifies the start of revision marked text based on extractions from the published Recommendations affected by the correction being described.
[End Correction]	Identifies the end of revision marked text based on extractions from the published Recommendations affected by the correction being described.
...	Indicates that the portion of the Recommendation between the text appearing before and after this symbol has remained unaffected by the correction being described and has been omitted for brevity.
--- SPECIAL INSTRUCTIONS --- {instructions}	Indicates a set of special editing instructions to be followed.

Technical and editorial corrections

Clarifications to H.324 Annex C

This section describes the editorial corrections to clarify the level change procedure in the H.324 Annex C. The following text at the end of the Chapter C.7 of H.324 Annex C is needed.

[Begin Correction]

Note that after changing from level 0 to some higher levels, MUX-PDU octet alignment shall be preserved. Therefore, the transmitter shall add so many "0" bits after the level change sequence that the first synchronization flag of the new level will be octet aligned. In the transmitter, the reference for the octet alignment is the first transmitted bit. In the receiver, the reference for the octet alignment is the first bit of the first detected synchronization flag in the initial level set-up procedure.

[End Correction]

Implementation clarifications

This section describes the procedures for using the supplementary services Call Hold and Explicit Call Transfer in H.324/ISDN. Implementation of these procedures is optional.

Procedures for Call Hold (CH)

The two procedures as described below should be used if a terminal supports the Call Hold supplementary service.

1 Invocation procedure for CH

Initial situation: Terminal A is connected to terminal B. Either Terminal A or terminal B has established the call.

Objective: Terminal A wishes to put terminal B on hold.

1) In case Multilink is used, terminal A should remove all but one B-channel connections from the H.Multilink Channel Set according to the Multilink procedures.

2) Terminal A should proceed with phase F of Annex D/H.324. The **EndSessionCommand** message should indicate to the far end that the terminal will be put on hold by signalling **terminalOnHold** in **isdnOptions**.

3) Terminal A should invoke the CH supplementary service by D-channel signalling, requesting the network to put all B-channel connections with terminal B on hold.

2 Retrieval after invocation of CH

Initial situation: Terminal A has terminal B on hold.

Objective: Terminal A wishes to retrieve the call with terminal B.

1) Terminal A should apply D-channel signalling to retrieve all the B-channel connections with terminal B.

2) Terminal A should initiate phase A of Annex D/H.324 starting with the execution of H.Dispatch, because the channel is already established.

3) Terminal A should add the additional B-channel connections to the H.Multilink Channel Set using the Multilink procedures.

NOTE - The CH procedures should only be used if both terminals A and B are H.324/I terminals.

Procedures for Explicit Call Transfer (ECT)

The procedure as described below should be used if a terminal supports the invocation of ECT.

Initial situation: Terminal A is connected to terminal B. Either terminal A or terminal B has established the call.

Objective: Terminal A wishes to put terminal B on HOLD, make a call to terminal C and then connect terminal B to terminal C.

1 Invocation procedure for ECT

1) In case Multilink is used, terminal A should disconnect all but one B-channel connections with terminal B according to the Multilink procedures defined in Annex F/H.324.

2) Terminal A should put terminal B on hold according to the procedures of the CH supplementary service.

3) Terminal A should establish a call with terminal C.

4) ECT should not be activated when terminal A does not succeed in establishing a call with terminal C or when terminal C is not a H.324/I terminal; appropriate indications should be given to the user(s).

5) In case Multilink is used, terminal A should disconnect all but one B-channel connections with terminal C according to the Multilink procedures defined in Annex F/H.324.

6) Terminal A should put terminal C on hold according to the procedures of the CH supplementary service.

7) Terminal A should invoke the ECT supplementary service by D-channel signalling, requesting the network to connect terminal B to C.

NOTE 1 - The procedure for ECT should only be used if all terminals A, B and C are H.324/I terminals. The implementation of ECT in case not all the terminals A, B and C are H.324/I terminals is left for further study.

NOTE 2 - The method used for addressing phone numbers in H.Multilink in case calls are transferred is left for further study.

NOTE 3 - The network provider may restrict the invocation of the ECT supplementary service to either the calling or the called terminal.

H.324 RECOMMENDATION SERIES DEFECT REPORT FORM

DATE:	
CONTACT INFORMATION **NAME:** **COMPANY:** **ADDRESS:** **TEL:** **FAX:** **EMAIL:**	
AFFECTED RECOMMENDATIONS:	
DESCRIPTION OF PROBLEM:	
SUGGESTIONS FOR RESOLUTION:	

NOTE - Attach additional pages if more space is required than is provided above.

4 Implementor's Guide for the ITU-T H.323, H.225.0, H.245, H.246, H.235 and H.450 series Recommendations - Packet-Based Multimedia Communication Systems

Contact Information

ITU-T Study Group 16/ Question 13 Rapporteur	Dale Skran Ascend Communications 620 Tinton Avenue Building A, Second Floor Tinton Falls, NJ. 07724 USA	Tel: +1 (908) 578 3101 Fax: +1 (908) 578 3131 Email: dale.skran@ascend.com
ITU-T Study Group 16/ Question 14 Rapporteur ITU-T Recommendation H.225.0 Editor	Glen Freundlich Lucent Technologies 11900 N. Pecos St. Westminster, CO 80234 USA	Tel: +1 303 538 2899 Fax: +1 303 538 5478 Email: ggf@lucent.com
ITU-T Recommendation H.235 Editor	James Toga MetaTel, Inc. Boston, MA, USA	Tel: +1 781 891 9000 Fax: +1 xxx xxx xxxx Email: jim.toga@metatel.com
ITU-T Recommendation H.245 Editor	Mike Nilsson BT Labs Ipswich, UK	Tel: +44 1473 645413 Fax: +44 1473 643791 Email: mike.nilsson@bt-sys.bt.co.uk
ITU-T Recommendation H.246 Editor	Mark Reid VideoServer, Inc. Burlington, MA, USA	Tel: +1 781 505 2368 Fax: +1 781 505 2101 Email: mreid@videoserver.com
ITU-T Recommendation H.323 Editor Implementor's Guide Editor	Paul E. Jones DataBeam Corporation 230 Lexington Green Circle Lexington, KY 40503 USA	Tel: +1 606 425 3609 Fax: +1 606 425 3528 Email: paul.jones@ties.itu.int
ITU-T Recommendations H.450.1, H.450.2, and H.450.3 Editor	Markku Korpi Siemens AG Munich, Germany	Tel: +49 89 722 34570 Fax: +49 89 722 23977 Email: korpim@sbs.de

Document History

Revision	Date	Description
A	April 1998	Initial version - completed at the ITU-T Study Group 16 Rapporteur's meeting, Yokuska, Japan April 1998
B	July 1998	Changes as a result of Rapporteur's meeting, Cannes June 1998
C	September 1998	Changes as a result of the SG 16 meeting, Geneva September 1998
D	May 1999	Changes as a result of the SG 16 meeting, Santiago May 1999

NOTE ABOUT THIS DRAFT

The following new sections have been for version D:

6.1.6 through 6.1.16

6.2.10 through 6.2.20

6.3.4 through 6.3.5

6.4.2

6.5.8 through 6.5.10

6.6.3

6.6.4

6.6.6

8.1-8.2

The following section has been modified for version D:

6.5.7

TABLE OF CONTENTS

1 Introduction

This document is a compilation of reported defects identified with the 1998 decided editions of the ITU-T H.323-series Recommendations. It is intended to be read in conjunction with the Recommendations to serve as an additional authoritative source of information for implementers. The changes, clarifications and corrections defined herein are expected to be included in future versions of affected H.323-series Recommendations.

2 Scope

This guide resolves defects in the following categories:

- editorial errors;
- technical errors such as omissions or inconsistencies;
- ambiguities.

In addition the Guide may include explanatory text found necessary as a result of interpretation difficulties apparent from the defect reports.

This Guide will not address proposed additions, deletions or modifications to the Recommendations that are not strictly related to implementation difficulties in the above categories. Proposals for new features should be made in the normal way through contributions to the ITU-T.

3 Defect Resolution Procedure

Upon discovering technical defects with any components of the H.323 Recommendations series, please provide a written description directly to the editors of the affected Recommendations with a copy to the Q.13/16 or Q.14/16 Rapporteur. The template for a defect report is enclosed. Contact information for these parties is included in this document. Return contact information should also be supplied so a dialogue can be established to resolve the matter and an appropriate reply to the defect report can be conveyed. This defect resolution process is open to anyone interested in H.323-series Recommendations. Formal membership in the ITU is not required to participate in this process.

4 References

This document refers to the following H.323-series Recommendations:

– ITU-T Recommendation H.323 (1998), *Packet-Based Multimedia Communications Systems.*

– ITU-T Recommendation H.225.0 (1998), *Call Signalling Protocols and Media Stream Packetization for Packet Based Multimedia Communications Systems.*

– ITU-T Recommendation H.245 (1998), *Control Protocol for Multimedia Communication.*

– ITU-T Recommendation H.235 (1998), *Security and Encryption for H Series (H.323 and other H.245 based) multimedia terminals.*

– ITU-T Recommendation H.450.1 (1998), *Generic functional protocol for the support of supplementary services in H.323.*

– ITU-T Recommendation H.450.2 (1998), *Call transfer supplementary service for H.323.*

– ITU-T Recommendation H.450.3 (1998), *Call diversion supplementary service for H.323.*

5 Nomenclature

In addition to traditional revision marks, the following marks and symbols are used to indicate to the reader how changes to the text of a Recommendation should be applied:

Symbol	Description
[Begin Correction]	Identifies the start of revision marked text based on extractions from the published Recommendations affected by the correction being described.
[End Correction]	Identifies the end of revision marked text based on extractions from the published Recommendations affected by the correction being described.
...	Indicates that the portion of the Recommendation between the text appearing before and after this symbol has remained unaffected by the correction being described and has been omitted for brevity.
--- SPECIAL INSTRUCTIONS --- {instructions}	Indicates a set of special editing instructions to be followed.

6 Technical and Editorial Corrections

6.1 Technical and Editorial Corrections to ITU-T Recommendation H.323

6.1.1 Early Call Signalling channel closure

Description: An incomplete description concerning closing of the call signalling channel is contained within section 7.3.1 of H.323.

This information will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.323 document that was submitted for approval in 1998.

In general this change should not effect implementations, it is intended to clarify consistency issues should they occur. The corrected text is shown below.

[Begin Correction]

7.3.1 Call signalling channel routing

...

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

[End Correction]

6.1.2 FastConnect Clarifications

Description: In section 8.1.7 of H.323, Fast Connect Procedures, the text regarding the refusal of the fast connect is not clear and may be confusing to the reader.

The clarified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change should not affect the functionality or local operation of an Gatekeeper or endpoint.

[Begin Correction]

8.1.7 Fast Connect Procedure

...

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

[End Correction]

Description: An inconsistency in the use of channel addressing within the FastStart procedure has been discovered.

This information will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.323 document that was submitted for approval in 1998.

As a part of the "Fast Connect Procedure", it is the responsibility of the called endpoint to decide about the coupling of proposed incoming and outgoing media streams for each SessionId. According to RTP/RTCP specifications, media streams in the same session, must use a common RTCP channel. Additionally, many RTP/RTCP implementations mandate adjacent odd/even port pairs to be allocated.

This requirement mandates that for any and all fastStart structures that are proposed in a SETUP message, those with common sessionID values shall also have common mediaChannelControl values.

This change may affect the functionality or local operation of an implementation.

[Begin Correction]

8.1.7.1 Proposal, Selection, and Opening of Media Channels

...

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

All alternative OpenLogicalChannel structures, that propose a channel for transmission from the called endpoint to the calling endpoint, shall contain the same sessionID and the same mediaChannel value

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

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

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

When accepting a proposed channel for transmission from the calling endpoint to the called endpoint, the called endpoint shall return the corresponding **OpenLogicalChannel** structure to the calling endpoint.~~,~~ The called endpoint shall ~~inserting a~~insert valid mediaChannel and **media**~~**Control**~~**Channel**~~element~~ fields (indicating the RTCP channel going in the same direction) into the **H2250LogicalChannelParameters** element of the **forwardLogicalChannelParameters** structure. The called endpoint shall then prepare to immediately receive media flow according to the parameters specified in **forwardLogicalChannelParameters**. The calling endpoint may begin transmitting media on the accepted and opened channels upon receipt of the Q.931 response containing **fastStart**, and may release any resources allocated to reception on proposed channels that were not accepted.

[End Correction]

6.1.3 Gateway Inbound Calling

Description: An omission in the operation of gateways during inbound calls between the SCN and the IP network.

This information will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.323 document that was submitted for approval in 1998.

The TCS-4/IIS option of requesting a remote LAN extension was omitted.

This change may affect the functionality or local operation of an gateway depending on options implemented.

[Begin Correction]

8.1.8.1 Gateway inbound call set-up

...

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

[End Correction]

6.1.4 Facility Redirection

Description: A clarification in the operation of MC(U)s which host multiple conferences has been added.

This information will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, this information appears missing in the final H.323 document that was submitted for approval in 1998.

The clarifying paragraph is shown below.

This change may affect the functionality or local operation of an MC(U) or endpoints depending on options implemented.

[Begin Correction]

8.4.3.1 Direct Endpoint Call Signalling - Conference Create

...

A2d) If the MC(U) hosts multiple conferences and wishes to provide endpoint 1 with a choice of conferences to join, it can send a Facility message indicating conferenceListChoice and a list of conferences that endpoint 1 may choose from. The list of conferences is sent as part of the Facility-UUIE. For backward compatibility, with version 1 endpoints, conference lists are only provided if the ProtocolIdentifier in endpoint 1's Setup message indicates that it is version 2 or above.

The recipient of this "routeCallToMC" Facility message should consider the previous exchange completed and send a new SETUP message to the MC(U) address with the chosen conference it wishes to join.

[End Correction]

6.1.5 H.323 Annex C - Use of B-HLI Field

Description: An error in the use of the B-HLI field concerning mapping between ATM virtual circuits and logical channel numbers has been discovered.

B-HLI is used by the receiving endpoint to associate the ATM VC with the proper RTP logical channel. The endpoint that initiates the OpenLogicalChannel command is the endpoint that opens the ATM VC. It is possible for the initiating endpoint to select a B-HLI that is already in use by the receiving endpoint. This would cause a failure in the OLC procedure.

Additionally the receiving RTCP port is also specified by the initiating endpoint by implication. H.323 states that the corresponding RTCP data shall flow on a UDP port number equal to the VC Association port number plus 1. It is possible that the resulting port number for RTCP, VC Association port number plus 1, will be in use on the receiving endpoint since the VC Association port number is selected by the initiating endpoint.

Due to the above problems the receiving endpoint should have the choice of selecting the B-HLI.

This change will affect the functionality or local operation of an implementation. Note that there is an associated ASN.1 change within H.245.

[Begin Correction]

C.4.1.1 Broadband High Layer Information

IE Parameter	Value	Notes
Length of B-HLI contents (octets 3-4)	3	One octet type plus two octets VC association port number
High layer information type (octet 5)	"000 0001"	User-specific
High layer information (octets 6-7)	VC association port number	In basic mode, the UDP port number to be used for RTP

It should be noted that the **~~portID~~ portNumber** field in H.245 is only 16 bits in length. For this reason, only 16 bits are used in the B-HLI High Layer Information parameter.

The portNumber field of the OpenLogicalChannel message is used to select the B-HLI. The receiving endpoint uses this B-HLI to associate the ATM VC with the proper RTP logical channel. If the receiving endpoint finds that the given B-HLI is inappropriate it can select a new B-HLI and use the portNumber field of the OpenLogicalChannelAck message to indicate the new value to the initiating endpoint. The selected portNumber field is conveyed

in the B-HLI information element. The format of the B-HLI is specified in the protocol section below. This enables the receiving side to associate the ATM VC with the proper RTP logical channel.

The VC association port number is represented in network byte order in octets 6 and 7 of the B-HLI (i.e. octet 6 holds the MSB and octet 7 holds the LSB).

~~The VC Association port number is used to identify the ATM VC for the RTP media stream. The corresponding RTCP data shall flow on a UDP port number equal to the VC Association port number plus 1.~~

[End Correction]

6.1.6 H.323 Annex C - Indication of ATM capabilities in TransportCapability

Description: Annex C requires that the indication of ATM be indicated in the Terminal Capability Set, but it implies that it is done in the capability exchange procedures. While this is valid when fast start is not used, it is not when fast start procedures are used since they do not use the capability exchange procedures before setting up the channels. The current wording unfortunately prohibits the use of fast start with Annex C. This contribution proposes that the wording be amended to allow for this indication to be provided in fast start as well. More specifically, the current Annex mandates that the Terminal Capability set be used to indicate the ATM capabilities. If fast start is used, the TransportCapability (where the ATM information is included) is not included. It is however included in the OpenLogicalChannel. Annex C shall be amended the text to reflect this. It shall also be clarified that OpenLogicalChannel is used instead of OpenLogicalChannelAck when fast start is used..

The clarified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

[Begin Correction]

C.3.6 Transport Capabilities added to ~~Terminal~~ TransportCapability Set

For operation of H.323 on AAL5, an addition to the ~~Terminal~~ **TransportCapability** set is made in H.245. This includes transport level capabilities such as support for ATM Transfer Capability (DBR, SBR1, SBR2, SBR3, ABT/DT, ABT/IT, ABR) as defined in Recommendation I.371. Terminals that do not send this new capability parameter shall not make use of the new methods described in this Annex. The **TransportCapability** information can be sent as part of the Terminal Capability set exchange in the capability exchange phase. It is also included in the **OpenLogicalChannel**.

...

C.3.7.1 ATM address

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

...

C.3.8.2 Bidirectional logical channels

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

...

[End Correction]

6.1.7 Remote Device Control

Description: With the ongoing development of the new Recommendations of H.282 and H.283 covering device control, the previous text related to this functionality is no longer valid

The clarified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change should not affect the current functionality of an endpoint.

[Begin Correction]

2 Normative references

...

[29] ITU-T Recommendation I.371.1 (1997), *Traffic control and congestion control in B-ISDN: Conformance definitions for ABT and ABR.*

[30] ITU-T Recommendation Q.2961.2 (1997), *Support of ATM Transfer capability in the broadband bearer capability information element.*

[31] ITU-T Recommendation H.282 (1999), *Remote Device Control Protocol for Multimedia Applications.*

[32] ITU-T Recommendation H.283 (1999), *Remote Device Control Logical Channel Transport.*

~~[31] ITU-T Recommendation H.224 (1994), *A real time control protocol for simplex applications using the H.221 LSD/HSD/MLP channels*.~~

~~[32] ITU-T Recommendation H.281 (1994), *A far-end camera control protocol for videoconferences using H.224.*~~

[End Correction]

[Begin Correction]

6.2.7 Data channel

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

Recommendation T.120 is the default basis of data interoperability between an H.323 terminal and other H.323, H.324, H.320, or H.310 terminals. Where any optional data application is implemented using one or more of the ITU-T Recommendations which can be negotiated via H.245, the equivalent T.120 application, if any, shall be one of those provided. ~~A terminal that provides far-end camera control using H.281 and H.224 is not required to also support a T.120 far-end camera control protocol.~~

...

[End Correction]

[Begin Addition]

6.2.7.2 Remote Device Control

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

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

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

[End Addition]

6.1.8 H.323 Protocol Revisions

Description: This section presents a clarification to the use of revisions of specific protocol Recommendations within the H.323 system, Version 2.

The clarified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change may affect the current functionality of an H.323 system.

[Begin Correction]

Summary

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

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

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

...

[End Correction]

6.1.9 Version Numbers in Gatekeeper routed Calls

Description: This section presents a clarification to the use of version numbers in both the H.225.0 and H.245 messages when routing through a Gatekeeper. This clarification is needed in order to verify the correct inter-working between version 1 and version 2 systems.

The clarified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change may affect the functionality or local operation of an H.323 endpoint or Gateway depending on current implementation.

[Begin Addition]

7.3.3 Call Signalling and Control protocol revisions

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

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

b) If the originating endpoint's version number is greater than gatekeeper's, the routed messages shall reflect the version number of the gatekeeper.

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

[End Addition]

6.1.10 Master/Slave Determination

Description: This section presents a clarification to the use of H.245 master-slave determination in the context of FastStart.

Specifically, the strict requirements of the master-slave procedure are not applicable in all cases. The result of this change does not minimize the implementation of this procedure, only the use of the procedure in all cases.

This change will be reflected in H.323 v3.

This change may affect the functionality or local operation of an H.323 endpoint or Gateway depending on options implemented.

[Begin Correction]

8.2 Phase B - Initial communication and capability exchange

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

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

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

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

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

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

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

Following successful completion of the requirements of Phase B ~~this exchange of capabilities~~, the endpoints shall proceed directly to the desired operating mode, normally ~~i.e.~~ Phase C.

[End Correction]

6.1.11 H.245 Tunnelling

Description: The ability to tunnel H.245 PDUs inside H.225.0 call signalling messages provides a high level of flexibility. The h245control element, which is the tunnelling mechanism, allows for multiple H.245 PDUs to be sent in a single H.225.0 message.

The order in which H.245 PDUs are encapsulated and extracted from the tunnel is undefined. In some instances H.245 procedures are being executed in parallel message order becomes significant.

For this to be successful, the encapsulating endpoint must know the order in which the destination will process the encapsulated H.245 PDUs.

The clarified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change may affect the current functionality of an endpoint.

[Begin Correction]

8.2.1 Encapsulation of H.245 Messages within Q.931 Messages

...

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

[End Correction]

6.1.12 Endpoint Registration

Description: This section presents a clarification to the use of RRQ messages received by a Gatekeeper.

Section 7.2.2 of H.323 version 2 refers repeatedly to "previous RRQ". Literal interpretation of these references is incorrect. For example, the second sentence reads, "If a Gatekeeper receives an RRQ having the same alias address and the same Transport Address as a previous RRQ, it shall respond with RCF." This means that if an endpoint re-sends an RRQ to a particular Gatekeeper after receiving an RRJ for its initial RRQ, the Gatekeeper must send the endpoint an RCF.

The clarified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change may affect the current operation of an endpoint or Gatekeeper.

[Begin Correction]

7.2.2 Endpoint Registration

...

The RRQ may be repeated periodically (i.e. at terminal power-up), so the Gatekeeper shall be able to handle multiple requests from the same endpoint. If a Gatekeeper receives an RRQ having the same alias address and the same Transport Address as an active registration ~~previous RRQ~~, it shall respond with RCF. If a Gatekeeper receives an RRQ having the same alias address as an active registration ~~previous RRQ~~ and a different Transport Address, it may confirm the request, if it complies with the Gatekeeper's security policy. Otherwise, it should reject the registration indicating a duplicate registration. If the Gatekeeper receives an

RRQ having the same Transport Address as an active registration ~~previous RRQ~~ and a different alias address, it should replace the translation table entries. The Gatekeeper may have a method to authenticate these changes.

...

[End Correction]

6.1.13 Lightweight Registration

Description: This section presents a clarification to the use of Time To Live timer in association with the keepAlive re-registration.

The new text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change may affect the current functionality of an endpoint.

[Begin Correction]

...

7.2.2 Endpoint Registration

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

...

[End Correction]

[Begin Addition]

7.2.2.1 Use of Lightweight RRQ

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

registration to be extended. After the expiration time, the endpoint must re-register with a Gatekeeper using a full RRQ message.

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

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

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

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

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

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

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

[End Addition]

[Begin Correction]

...

```
RegistrationRejectReason ::= CHOICE
{
    discoveryRequired           NULL,   -- registration permission has aged
    invalidRevision             NULL,
    invalidCallSignalAddress    NULL,
    invalidRASAddress           NULL,   -- supplied address is invalid
    duplicateAlias              SEQUENCE OF AliasAddress,
                                        -- alias registered to another endpoint
    invalidTerminalType         NULL,
    undefinedReason             NULL,
    transportNotSupported       NULL,   -- one or more of the transports
    ...,
    transportQOSNotSupported    NULL,   -- endpoint QoS not supported
    resourceUnavailable         NULL,   -- gatekeeper resources exhausted
```

```
    invalidAlias                NULL,   -- alias not consistent with gatekeeper rules
    securityDenial              NULL,
    fullRegistrationRequired    NULL    -- registration permission has expired
}
```

…

[End Correction]

6.1.14 Gatekeeper – MC Access

Description: In sections of H.323, the assumption appears to be made in a few places within H.323 that a gatekeeper which is routing H.245 signalling has (or has access to) an MC. This is not explicitly stated, and was not intended to be a requirement.

The clarified text will be contained in the revision 3 of H.323.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change should not affect the functionality or local operation of a Gatekeeper or endpoint.

[Begin Correction]

8.1.2 Both endpoints registered with the same Gatekeeper

In the scenario shown in Figure 15, both endpoints are registered to the same Gatekeeper, and the

…

H.245 signalling. The Gatekeeper sends the Connect (10) message to endpoint 1 which may contain the endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.

8.1.3 Only calling endpoint has Gatekeeper

In the scenario shown in Figure 17, endpoint 1 (calling endpoint) is registered with a Gatekeeper, endpoint

…

Channel Transport Address for use in H.245 signalling. The Gatekeeper sends the Connect (8) message to endpoint 1 which may contain the endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.

8.1.4 Only called endpoint has Gatekeeper

In the scenario shown in Figure 19, endpoint 1 (calling endpoint) is not registered with a Gatekeeper,

...

Gatekeeper sends the Connect (13) message to endpoint 1 which may contain the endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.

8.1.5 Both endpoints registered to different Gatekeepers

In the scenario shown in Figure 21, both endpoints are registered to different Gatekeepers, the calling

...

endpoint 1 which may contain the endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper 2 ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.

...

In the scenario shown in Figure 22, both endpoints are registered to different Gatekeepers, the calling

...

Connect (10) message to endpoint 1 which may contain the endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper 1 ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the Gatekeeper chooses to route the H.245 Control Channel or not.

...

In the scenario shown in Figure 23, both endpoints are registered to different Gatekeepers, and both Gatekeepers choose to route the call signalling. Endpoint 1 (calling endpoint) initiates the

...

for use in H.245 signalling. Gatekeeper 2 sends the Connect (16) message to Gatekeeper 1 which may contain the Endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper 2 ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the Gatekeeper 2 chooses to route the H.245 Control Channel or not. Gatekeeper 1 sends the Connect (17) message to Endpoint 1 which may contain the H.245 Control Channel Transport Address sent by Gatekeeper 2, or a Gatekeeper 1 ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the Gatekeeper 1 chooses to route the H.245 Control Channel or not.

8.1.6 Option Called Endpoint Signalling

...

The procedures defined in 8.1.4 and 8.1.5 show that when a called endpoint is registered to a Gatekeeper, a Setup message is initially sent to the called endpoint from the calling endpoint or the

...

for use in H.245 signalling. Gatekeeper 2 sends the Connect (13) message to Gatekeeper 1 which may contain the Endpoint 2 H.245 Control Channel Transport Address, or a Gatekeeper 2 ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the

Gatekeeper 2 chooses to route the H.245 Control Channel or not. Gatekeeper 1 sends the Connect (14) message to Endpoint 1 which may contain the H.245 Control Channel Transport Address sent by Gatekeeper 2, or a Gatekeeper 1 ~~(MC)~~ H.245 Control Channel Transport Address, based on whether the Gatekeeper 1 chooses to route the H.245 Control Channel or not.

8.4.3.1 Direct Endpoint Call Signalling - Conference Create

A4b) Using H.245 master-slave determination procedure, it is determined that endpoint 2 is the master. In the Gatekeeper-Routed model, the master could be in an MC collocated with the Gatekeeper~~the Gatekeeper's MC~~. If the master has an MC, it becomes the Active MC. It may then send the **MCLocationIndication** to the other endpoint(s). The MC may be active in the conference now, or when the user initiates the multipoint conference function, at the choice of the manufacturer.

[End Correction]

6.1.15 Master/Slave Clarification

Description: The last paragraph of the master/slave procedure described in section 8.4.3.4 of H.323 has been determined to be unclear.

The clarified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the clarifying paragraph shown below.

This change should not affect the functionality or local operation of a Gatekeeper or endpoint.

[Begin Correction]

8.4.3.4 Gatekeeper Routed Call Signalling - Conference Create

...

During master-slave determination [A4b], if the Gatekeeper's **terminalType** is greater than the **terminalType** received ~~from an endpoint~~ in the **masterSlaveDetermination** message, the Gatekeeper may attempt to become master for the call. In this case, the Gatekeeper shall immediately send a **masterSlaveDeterminationAck** message to the source of the Master-Slave Determination message indicating that it is a slave and the Gatekeeper performs Master-Slave Determination with the destination entity as defined in section 6.2.8.4. If the Gatekeeper wins that Master-Slave Determination, the MC associated with the Gatekeeper shall be the active MC.~~may replace the endpoint's **terminalType** value with its own before forwarding the message to the destination endpoint.~~ If the Gatekeeper's **terminalType** is not greater than the **terminalType** of the endpoint or the Gatekeeper decides not to replace the endpoint's **terminalType** with its own, the Gatekeeper shall not modify the **terminalType** value and it shall transparently relay all messages of that Master-Slave Determination procedure. ~~In effect, the Gatekeeper is performing the master-slave determination procedure~~

~~with each endpoint. If the Gatekeeper wins the master-slave determination with both endpoints, the MC associated with the Gatekeeper shall be the active MC; otherwise, one of the endpoints shall be the active MC.~~

[End Correction]

6.1.16 Clarification of OLCs within the Context of Fast Start

Description: Section 8.3 of H.323 discusses the procedures of learning the address for the remote endpoint's media channel. However, this section was not properly updated to reflect changes introduced with the introduction of Fast Start.

The modified text will be contained in the revision 3 of H.323 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.323 should be amended with the paragraph shown below.

[Begin Correction]

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

[End Correction]

6.2 Technical and Editorial Corrections to ITU-T Recommendation H.225.0

6.2.1 Use of Connect Acknowledge

Description: In section 7.3.4 Connect Acknowledge states "This message shall not be sent." However, additional text states "Follow Table 3-5/Q.931 as modified below", and includes Table 8/H.225.0 showing the contents of the Connect Acknowledge message. This message is not allowed in H.225.0

An error in the description of the Connect Acknowledge message has been detected. The included table should be deleted.

This information will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.225.0 document that was submitted for approval in 1998. The Connect Acknowledge message is not permitted, the corrected text is shown below.

[Begin Correction]

7.3.4 Connect Acknowledge

~~Follow Table 3-5/Q.931 as modified below.~~

This message shall not be sent.

~~Information element~~	~~H.225.0 status(M/F/O)~~	~~Length in H.225.0~~
~~Protocol discriminator~~	~~M~~	~~1~~
~~Call reference~~	~~M~~	~~3~~
~~Message type~~	~~M~~	~~1~~
~~Display~~	~~O~~	~~2-82~~
~~Signal~~	~~O~~	~~2-3~~
~~User-to-User~~	~~M~~	~~2-131~~

~~Table 8/H.225.0~~

[End Correction]

6.2.2 Information Element Labelling

Description: An error in the labeling of the Information element as described in H.225.0

This information will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.225.0 document that was submitted for approval in 1998.

This change does not affect the functionality or operation of the protocol.

[Begin Correction]

7.3.6 Information

This message may be sent to provide~~supplementary~~ additional information. <u>It may be used to provide information for call establishment (e.g. overlap sending) or miscellaneous call-related information.</u> It may be used to deliver proprietary features.

This message may be sent by an H.323 entity; its processing on receipt is optional.

This message follows Table 3-7/Q.931 with the following modifications:

Information element	H.225.0 status (M/F/O)	Length in H.225.0
Protocol discriminator	M	1
Call reference	M	3
Message type	M	1
Sending complete	O	1
Display	O	2-82
Keypad facility	O	2-34
Signal	O	2-3
Called party number	O	2-35
User-to-User	M	2-131

Table 9/H.225.0

Information Message Content

The user-to-user information element contains the ~~UI~~Information-UUIE defined in the H.225.0 Message Syntax. The ~~U~~Information-UUIE includes the following:

protocolIdentifier - set to the version of H.225 supported

callIdentifier - a globally unique call identifier set by the originating endpoint which can be used to associate RAS signalling with the modified Q.931 signalling used in H.225.0

[End Correction]

6.2.3 Progress Message

Description: An error has been detected concerning the required Information Elements in a Progress message.

This correction will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this appears incorrectly in the final H.225.0 document that was submitted for approval in 1998.

In general this change should not effect implementations, the corrected text is shown below. Table A/H.225.0 indicates the progress indicator IE is optional in the Progress message, but this IE should be marked as mandatory.

[Begin Correction]

7.3.7 Progress

...

Information element	**H.225.0 status (M/F/O)**	**Length in H.225.0**
Protocol discriminator	M	1
Call reference	M	3
Message type	M	1
Bearer capability	O (Note 1)	5-6
Cause	O	2-32
Extended facility	O	8-*
Channel identification	FFS	NA
Facility	O	8-*
Progress indicator	~~O~~M	2-4
Notification Indicator	O	2-*
Display	O	2-82
High layer compatibility	FFS	NA
User-to-User	M	2-131

Table A/H.225.0
Progress

[End Correction]

6.2.4 Missing Field Descriptions

Description: Omitted descriptions of the indicated ASN.1 elements as described in H.225.0 have been detected.

This information will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.225.0 document that was submitted for approval in 1998.

This change does not affect the functionality or operation of the protocol.

[Begin Correction]

7.8.1 GatekeeperRequest (GRQ)

...

algorithmOIDs - indicates the entire set of encryption algorithms supported by the endpoint

7.8.2 GatekeeperConfirm (GCF)

...

algorithmOID - indicates the encryption algorithm required by the Gatekeeper

---

[End Correction]

6.2.5 Use of CallIdentifier in IRQ

Description: An unclear description of the callReferenceValue in the IRQ message of H.225.0 has been detected.

These clarifications will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.225.0 document that was submitted for approval in 1998.

This change should not affect the functionality or operation of the protocol, it may affect the response of an endpoint to this message.

[Begin Correction]

...

7.15 InfoRequest (IRQ)

callReferenceValue - CRV of the call that the query is about. If zero, this message is interpreted as a request for an IRR for each call the terminal is active on. If the terminal is not active on any calls, an IRR shall be sent in response to a CallReferenceValue of 0 with all appropriate fields provided. If callReferenceValue is 0, the endpoint shall ignore callIdentifier - in this case the gatekeeper shall fill callIdentifier with 0.

[End Correction]

6.2.6 H.225.0 Non-Standard Message

Description: An error in the description of the requestSeqNum contained within the Non Standard message has been detected.

This information will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.225.0 document that was submitted for approval in 1998.

The Non Standard message has no well defined or corresponding response message and therefore there can be no correlation between sequence numbers on incoming and outgoing messages from a particular endpoint.

If H.323 applications are to utilize this message and need to be able to detect lost or duplicated messages, the implementation must supply its own sequencing information within the body of the message. The corrected text is shown below.

[Begin Correction]

7.16 Non-Standard Message

The **NonStandardMessage** structure is as follows:

requestSeqNum - this is a monotonically increasing number unique to the sender.~~It shall be returned by the receiver in any response associated with this specific message.~~

...

[End Correction]

6.2.7 Retries & Timeouts for RAC/RAI

Description: Missing timeout values or retry counters were discovered for RAI/RAC in H.225.0 section 7.19.

This information will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.225.0 document that was submitted for approval in 1998.

In general this change should not effect implementations, it is intended to clarify interworking issues should they occur. The corrected text is shown below.

[Begin Correction]

7.19 RAS Timers and Request in Progress (RIP)

...

RAS Message	timeout value (sec)	retry count
GRQ	5	2
RRQ	3	2
URQ	3	1
ARQ	3	2
BRQ	3	2
IRQ	3	1
IRR Note 1	5	2
DRQ	3	2
LRQ	5	2
RAI	3	2

[End Correction]

6.2.8 G.723.1 Audio Packetization

Description: A misleading statement in the description of the G.723.1 packetization is contained within section 13.

This information will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.225.0 document that was submitted for approval in 1998.

In general this change should not effect implementations, it is intended to clarify interworking issues should they occur. The corrected text is shown below.

[Begin Correction]

ANNEX F - Audio Packetization

This annex describes RTP packetization details for audio codecs standardized by the ITU.

This Recommendation specifies a coded representation that can be used for compressing the speech signal component of multi-media services at a very low bit rate. A G.723.1 frame can be one of three sizes: 24 bytes (6.3 kb/s frame), 20 bytes (5.3 kb/s frame), or 4 bytes. These 4-byte frames are called SID frames (Silence Insertion Descriptor) and are used to specify comfort noise parameters. There is no restriction on how 4, 20, and 24 byte frames are intermixed. The least significant two bits of the first octet in the frame determine the frame size and codec type (refer to Table 5/G.723.1 and Table 6/G.723.1 for more information on bit order). It is possible to switch between the two rates at any 30 ms frame boundary. Both (5.3 kb/s and 6.4 kb/s) rates are a mandatory part of the encoder and decoder. This coder was optimized to represent speech with near-toll quality at the above rates using a limited amount of complexity.

All the bits of the encoded bit stream are transmitted always from the least significant bit towards the most significant bit. NOTE - This refers to the order of bits presented to the transport layer and not the order of bits on the wire.

[End Correction]

6.2.9 ANNEX H - H.225.0 Message Syntax (ASN.1)

Description: A number of errors have been detected in the ASN.1 syntax of H.225.0 and are shown in the corrected text below.

This information will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.225.0 document that was submitted for approval in 1998.

The syntax has been changed to appropriately label the information elements as described in the following correcting section.

The Facility message has had a fastStart field added to it. When utilizing the gatekeeper routed model, the gatekeeper will respond to a Setup with a Call Proceeding. When the gatekeeper sends the Setup to an endpoint, the endpoint can respond with Call Proceeding, but the gatekeeper cannot forward the Call Proceeding to the originating endpoint. If the endpoint included fast start information in the Call Proceeding, the gatekeeper can pass this information to the originating endpoint in a Facility message. The Progress message has inappropriate extension markers. These should have been placed at the end of the currently defined message.

The FastStart token as previously defined, did not include the ASN.1 syntax to enforce the presence of required fields within this token. This correction does not change any protocol that is transmitted.

An error in the syntax for security tokens is present in H.225.0 version 2. The typographical error provides a circular reference, which although is not detrimental to implementations, does not provide the intended option.

In order to allow H.323 implementations to utilize the generic H.235 token format in an application specific manner, the ASN.1 is changed to account for the typographical error. Note that this is a required change whether the functionality is used or not.

[Begin Correction]

...

```
H323-UU-PDU ::= SEQUENCE
{
    h323-message-body CHOICE
    {
        setup                   Setup-UUIE,
        callProceeding          CallProceeding-UUIE,
        connect                 Connect-UUIE,
        alerting                Alerting-UUIE,
        ~~userInformationinformation UUInformation-UUIE,~~information     Information-UUIE,
        releaseComplete         ReleaseComplete-UUIE,
        facility                Facility-UUIE,
        ...,
        progress                Progress-UUIE,
        empty                   NULL        -- used when a FACILITY message is sent,
                                            -- but the Facility-UUIE is not to be invoked
                                            -- (possible when transporting supplementary
                                            -- services messages)
    },
    nonStandardData     NonStandardParameter OPTIONAL,
    ...,
    h4501SupplementaryService       SEQUENCE OF OCTET STRING OPTIONAL,
```

```
                                -- each sequence of octet string is defined as one
                                -- H4501SupplementaryService APDU as defined in
                                -- Table 3/H.450.1
        h245Tunnelling          BOOLEAN,
                                -- if TRUE, tunnelling of H.245 messages is enabled
        h245Control             SEQUENCE OF OCTET STRING OPTIONAL,
                                -- each octet string may contain exactly
                                -- one H.245 PDU
        nonStandardControl      SEQUENCE OF NonStandardParameter OPTIONAL
}


~~UI~~Information-UUIE  ::=SEQUENCE
{
        protocolIdentifier      ProtocolIdentifier,
        ...,
        callIdentifier          CallIdentifier,
        tokens                  SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
        fastStart               SEQUENCE OF OCTET STRING OPTIONAL
}

ReleaseCompleteReason ::= CHOICE
{
        noBandwidth             NULL,   -- bandwidth taken away or ARQ denied
        gatekeeperResources     NULL,   -- exhausted
        unreachableDestination  NULL,   -- no transport path to the destination
        destinationRejection    NULL,   -- rejected at destination
        invalidRevision         NULL,
        noPermission            NULL,   -- called party's gatekeeper rejects
        unreachableGatekeeper   NULL,   -- terminal cannot reach gatekeeper for ARQ
        gatewayResources        NULL,
        badFormatAddress        NULL,
        adaptiveBusy            NULL,   -- call is dropping due to LAN crowding
        inConf                  NULL,   -- no address in AlternativeAddress
        undefinedReason         NULL,
        ...,
        facilityCallDeflection  NULL,   -- call was deflected using a Facility message
        securityDenied          NULL,   -- incompatible security settings
        calledPartyNotRegistered NULL,  -- used by gatekeeper when endpoint has
                                        -- preGrantedARQ to bypass ARQ/ACF
        ~~callerNotregistered~~callerNotRegistered  NULL  -- used by gatekeeper when endpoint has
                                        -- preGrantedArq to bypass ARQ/ACF
}

Facility-UUIE ::= SEQUENCE
{
        protocolIdentifier      ProtocolIdentifier,
        alternativeAddress      TransportAddress OPTIONAL,
        alternativeAliasAddress SEQUENCE OF AliasAddress OPTIONAL,
        conferenceID            ConferenceIdentifier OPTIONAL,
        reason                  FacilityReason,
        ...,
        callIdentifier          CallIdentifier,
        destExtraCallInfo       SEQUENCE OF AliasAddress OPTIONAL,
        remoteExtensionAddress  AliasAddress OPTIONAL,
        tokens                  SEQUENCE OF ClearToken OPTIONAL,
        cryptoTokens            SEQUENCE OF CryptoH323Token OPTIONAL,
```

```
    conferences                     SEQUENCE OF ConferenceList OPTIONAL,
    h245Address                     TransportAddress OPTIONAL,
    fastStart                       SEQUENCE OF OCTET STRING OPTIONAL
}

Progress-UUIE ::= SEQUENCE
{
    protocolIdentifier              ProtocolIdentifier,
    destinationInfo                 EndpointType,
    h245Address                     TransportAddress OPTIONAL,
    callIdentifier                  CallIdentifier,
    h245SecurityMode                H245Security OPTIONAL,
    ...,
    tokens                          SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
    fastStart                       SEQUENCE OF OCTET STRING OPTIONAL,
    ...
}


FastStartToken ::= ClearToken (WITH COMPONENTS {..., timeStamp PRESENT, dhkey PRESENT,
generalID PRESENT -- set to 'alias' –- })
EncodedFastStartToken ::= TYPE-IDENTIFIER.&Type (FastStartToken)
CryptoH323Token::= CHOICE
{
    cryptoEPPwdHash SEQUENCE
        {
            alias       AliasAddress, -- alias of entity generating hash
            timeStamp   TimeStamp, -- timestamp used in hash
                        token       HASHED      { EncodedPwdCertToken –- generalID set to 'alias' -- }
    },
    cryptoGKPwdHash SEQUENCE
        {
                gatekeeperId        GatekeeperIdentifier, -- GatekeeperID of GK generating hash
    timeStamp   TimeStamp, -- timestamp used in hash
                token               HASHED      { EncodedPwdCertToken -- generalID set to
        Gatekeeperid -- }
    },
    cryptoEPPwdEncr         ENCRYPTED
                            { EncodedPwdCertToken –- generalID set to Gatekeeperid --},
    cryptoGKPwdEncr         ENCRYPTED
                            { EncodedPwdCertToken –- generalID set to Gatekeeperid --},
    cryptoEPCert        SIGNED { EncodedPwdCertToken –- generalID set to Gatekeeperid -- },
    cryptoGKCert            SIGNED { EncodedPwdCertToken –- generalID set to alias -- },
    cryptoFastStart         SIGNED { EncodedFastStartToken },
    nestedcryptoToken       Crypto~~H323~~Token,
    ...
}



UUIEsRequested ::= SEQUENCE
{
    setup                   BOOLEAN,
    callProceeding          BOOLEAN,
    connect                 BOOLEAN,
```

```
        alerting                        BOOLEAN,
        ~~userInformation~~information  BOOLEAN,
        releaseComplete                 BOOLEAN,
        facility                      BOOLEAN,
        progress                        BOOLEAN,
        empty                           BOOLEAN,
        ...
}
```

[End Correction]

6.2.10 Source Routed IP Addresses

Description: Limitations have been identified in the ipAddress structure within the TransportAddress common message element in H.225.0 for passage through Network Address Translation (NAT) and Firewalls. This field has certain limitations, which may lead to a need for extension in future, revisions, but is not directly extensible due to the lack of an extension marker ("…") in its ASN.1 encoding

This change will be reflected in H.225.0 v3.

This change clarifies a framework within which an extensible version of the IPv4 address structure is available within the current protocol, using the fact that the ipSourceRoute structure is extensible

This change may affect the functionality or local operation of an H.323 entity, depending on the operating environment.

[Begin Correction]

7.6 H.225.0 Common Message Elements

...

The IPv6 address a148:2:3:4:a:b:c:d shall have the 'a1' encoded in the first octet, '48' in the second, '00' in the third, '02' in the fourth and so forth.

A **TransportAddress** of type **ipSourceRoute** in which the **route** SEQUENCE has no entries shall be interpreted as representing the same address as of type **ipAddress** which contains the same values for both **ip** and **port**.

...

[End Correction]

6.2.11 RAS Timer Values and Registration Request

Description: In section 7.19 of H.225.0, the recommended default timer values have been found to be in error in one case and an omission in another. The timer value for RAI was omitted. The timer value for ARQ has been increased to account for the fact that the message itself may result in a nested LRQ/LCF sequence occurring before the ACF/ARJ can be returned.

(A new note needs to be added to RRQ to provide clarification to the text.)

The clarified text will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.225.0 should be amended with the clarifying paragraph shown below.

This change should not affect the function of any H.323 entities. It may provide some differences in perceived behaviour.

[Begin Correction]

7.19 RAS Timers and Request in Progress (RIP)

These are recommended default timeout values for the response to RAS messages and subsequent retry counts if a response is not received. (These values are subject to change with further implementation experience and input.)

RAS Message	timeout value (sec)	retry count
GRQ	5	2
RRQ NOTE 1	3	2
RAI	3	2
URQ	3	1
ARQ	~~3~~5	2
BRQ	3	2
IRQ	3	1
IRR NOTE 2~~1~~	5	2
DRQ	3	2
LRQ	5	2

NOTE 1 - The time-out value should be recalculated based upon both the time-to-live (which may be indicated by the Gatekeeper in the RCF message) and the desired number of retries.

NOTE ~~1~~ 2 - In cases where the gatekeeper is expected to reply to an unsolicited IRR with IACK or INAK, the timeout may occur if no reply to the IRR is received.

[End Correction]

6.2.12 TPKT Description

Description: In section 19.1 of H.225.0, IP usage, the text regarding the usage of TPKT is not clear and may be confusing to the reader.

The clarified text will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.225.0 should be amended with the clarifying paragraph shown below.

This change should not affect the functionality or local operation of an Gatekeeper or endpoint.

[Begin Correction]

19.1 TCP/IP/UDP

Note that UDP can fragment and re-assemble large video packets, but that failure to perform MB packetization may lead to the loss of an entire GOB.

IP multicast should be used for GRQ distribution as opposed to media access layer broadcast.

Unreliable delivery applications	**Call signalling and H.245 channel**
UDP	TPKT ___ ___ TCP
IP	
Link Layer	
Physical Layer	

A TPKT is a packet format as defined in IETF RFC1006. It is used to delimit individual messages (PDUs) within the TCP stream, which itself provides a continuous stream of octets without explicit boundaries. A TPKT consists of a one octet version number field, followed by a one octet reserved field, followed by a two octet length field, followed by the actual data. The version number field shall contain the value "3", the reserved field shall contain the value "0". The length field shall contain the length of the entire packet including the version number, the reserved and the length fields as a 16-bit big-endian word.

[End Correction]

[Begin Correction]

2.0 References

...

[33] Internet Engineering Task Force, 1987 "ISO transport services on top of the TCP: Version 3", RFC 1006, M.T. Rose, D.E. Cass.

[End Correction]

6.2.13 UDP Port Usage

Description: In section 19.1.1.1 of H.225.0, concerning IP and multicast ports, the text regarding the usage of these ports is not clear and may be confusing to the reader.

The clarified text will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.225.0 should be amended with the clarifying paragraph shown below.

This change may affect the functionality or local operation of a Gatekeeper or endpoint.

[Begin Correction]

19.1.1.1 Discovery Using Multicast Address or Well-Known Port

Following the gatekeeper discovery and registration procedures described in Section 7 of H.323, endpoints should use the following multicast address or well known port when attempting to discover the gatekeeper as appropriate for their network configuration:

- UDP Address for multicast communication with gatekeepers: 224.0.1.41
- UDP port for multicast communication with gatekeepers: 1718
- UDP port for unicast RAS communication where 'no other' agreement exists: 1719

Note - That "other agreement" may include registration of an endpoint with a gatekeeper.

~~- Gatekeeper UDP Discovery Multicast Address 224.0.1.41~~

~~- Gatekeeper UDP Discovery Port 1718~~

~~- Gatekeeper UDP Registration and Status Port 1719~~

Note that implementations should pay attention to the scope of the multicast so as to not flood the Internet with discovery messages.

Assuming a Gatekeeper has an IP address for example of 134.134.12.1, the following signalling may occur:

LRQ or GRQ arrives at 134.134.12.1: port 1719

LRQ or GRQ arrives at 134.134.12.1: port 1718 (note that this may occur with v1 GKs)

LRQ or GRQ arrives at 224.0.1.41: port 1718

The Gatekeeper may transmit an LRQ to the following addresses

224.0.1.41: port 1718 (multicast to all GKs)

X.X.X.X: port 1719 (to a specific GK)

Port 1719 should only be used when a request is sent unicast. This allows the receiver to know whether it should send a reject (xRJ) to the sender (it should in all cases).

Port 1718 should only be used when a request is sent multicast. The receiver should respond with the appropriate response, depending on the message. For LRQ no reject required, the receiver does not reply for multicast requests. For GRQ, a directed GRJ should be sent to the source of the GRQ.

[End Correction]

6.2.14 Multiple Destination Aliases

Description: In section 7.11.1, 7.13.1 of H.225.0, **destinationInfo** usage, the text regarding the usage of this field is not clear and may be confusing to the reader.

The clarified text will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.225.0 should be amended with the clarifying paragraph shown below.

ARQ, Setup and LRQ messages all contain "SEQUENCE OF" destination aliases, but no specific behaviour is documented in the standards for what a gatekeeper should do with them. Note that various behaviours are currently implemented by different vendors, with the result that gatekeepers are not as interchangeable as would be desirable, in the important sense that calls which will succeed with one gatekeeper will fail with another which claims obedience to the same standard.

The clarification may alter the behaviour of Gatekeepers.

[Begin Correction]

7.11.1 AdmissionRequest (ARQ)

The ARQ message includes the following:

requestSeqNum - this is a monotonically increasing number unique to the sender. It shall be returned by the receiver in any messages associated with this specific message.

callType - Using this value, gatekeeper can attempt to determine 'real' bandwidth usage. The default value is **pointToPoint** for all calls.I It should be recognized that the call type may change dynamically during the call and that the final call type may not be known when the ARQ is sent.

callModel - if **direct**, the endpoint is requesting the direct terminal to terminal call model. If **gatekeeperRouted**, the endpoint is requesting the gatekeeper mediated model. The gatekeeper is not required to comply with this request.

endpointIdentifier - This is an endpoint identifier that was assigned to the terminal by RCF.

destinationInfo - sequence of alias addresses for the destination, such as E.164 addresses or H323_IDs. When sending the ARQ to answer a call, **destinationInfo** indicates the destination of the call (the answering endpoint). If at least one alias is registered with a gatekeeper and no two aliases in the ARQ are registered to distinct people, the gatekeeper shall recognize the ARQ as referring to the registered identity. In the case of conflicting aliases the admission request should be rejected with cause AliasesInconsistent. If the gatekeeper does not provide this validation, it shall consider the first registered address to be the destination.

...

7.13.1 LocationRequest (LRQ)

The LRQ message includes the following:

requestSeqNum - this is a monotonically increasing number unique to the sender. It shall be returned by the receiver in any messages associated with this specific message.

endpointIdentifier - This is an endpoint identifier that was assigned to the terminal by RCF.

destinationInfo - sequence of alias addresses for the destination, such as E.164 addresses or H323_IDs. If at least one alias is registered with a gatekeeper and no two aliases in the LRQ are registered to distinct people, the gatekeeper shall recognize the LRQ as referring to the registered identity. In the case of conflicting aliases the location request should be rejected with cause AliasesInconsistent. If the gatekeeper does not provide this validation, it shall consider the first registered address to be the destination.

...

ANNEX H - H.225.0 Message Syntax (ASN.1)

...

```
AdmissionRejectReason ::= CHOICE
  {
       calledPartyNotRegistered        NULL,     -- can't translate address
       invalidPermission               NULL,     -- permission has expired
       requestDenied                   NULL,     -- no bandwidth available
       undefinedReason                 NULL,
       callerNotRegistered             NULL,
       routeCallToGatekeeper           NULL,
       invalidEndpointIdentifier       NULL,
       resourceUnavailable             NULL,
```

```
        ...,
        securityDenial                  NULL,
        qosControlNotSupported          NULL,
        incompleteAddress   NULL,
        aliasesInconsistent             NULL        -- multiple aliases in request
                                                    -- identify distinct people
}

                                                    ...
LocationRejectReason ::= CHOICE
{
        notRegistered                   NULL,
        invalidPermission               NULL,       -- exclusion by administrator or feature
        requestDenied                   NULL,       -- can't find location
        undefinedReason                 NULL,
        ...,
        securityDenial                  NULL,
        aliasesInconsistent             NULL        -- multiple aliases in request
                                                    -- identify distinct people
}

                                                    ...
```

[End Correction]

6.2.15 Lightweight Registration

Description: In section 7.9.1 of H.225.0, there is a list of fields to be included in a "lightweight" registration. In some cases, the Gatekeeper receiving this message may have previously removed the terminals entry - in which case it does not have a RAS port to which to send the response.

The clarified text will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.225.0 should be amended with the clarifying paragraph shown below.

This change may affect the functionality or local operation of an endpoint or Gatekeeper.

[Begin Correction]

7.9.1 RegistrationRequest (RRQ)

The RRQ message includes the following:

...

keepAlive - If set to TRUE indicates that the endpoint has sent this RRQ as a "keep alive". An endpoint can send a lightweight RRQ consisting of only rasAddress, keepAlive, endpointIdentifier, gatekeeperIdentifier, tokens, and timeToLive. A gatekeeper in receipt of RRQ with a keepAlive field set to TRUE should ignore fields other than endpointIdentifier,

gatekeeperIdentifier, tokens, and timeToLive. The rasAddress in a lightweight RRQ shall only be used by a gatekeeper as the destination for an RRJ when the endpoint is not registered.

endpointIdentifier - the endpointIdentifier provided by the gatekeeper during the original RCF

willSupplyUUIEs - If set to TRUE, this indicates that the endpoint will supply Q.931 message information in IRR messages if requested by the gatekeeper.

[End Correction]

6.2.16 Unsolicited IRRs with pregranted admission

Description: When using pregrantedARQ, a gatekeeper is unable to instruct a terminal to send unsolicited IRRs. A gatekeeper can still request IRR messages from the terminal by sending IRQ with call reference value of 0.

The unsolicited IRR mechanism is additionally enabled with the parameter "irrFrequencyInCall" in the preGrantedARQ field of the RegistrationConfirm message.

The clarified text will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.225.0 should be amended with the clarifying paragraph shown below.

This change may affect the functionality or local operation of an endpoint or Gatekeeper.

[Begin Correction]

7.9.2 RegistrationConfirm (RCF)

...

preGrantedARQ - Indicates events for which the gatekeeper has pre-granted admission. This allows for faster call setup times in environments where admission is guaranteed through means other than the ARQ/ACF exchange. Note that even if these fields are set to TRUE, an endpoint can still send an ARQ to the gatekeeper for reasons such as address translation, or the endpoint does not support this modified signalling mode. If the **preGrantedARQ** sequence is not present, then ARQ signalling shall be used in all cases. The flags are:

makeCall - If the **makeCall** flag is TRUE then the gatekeeper has pre-granted permission to the endpoint to initiate calls without first sending an ARQ. If the **makeCall** flag is FALSE, the endpoint shall always send ARQ to get permission to make a call.

useGKCallSignalAddressToMakeCall - If the **makeCall** and **useGKCallSignalAddressToMakeCall** flags are both set to TRUE, then if the endpoint does not send an ARQ to the gatekeeper to make a call, the endpoint shall send all H.225 call signalling to the gatekeeper call signalling channel.

answerCall - If the **answerCall** flag is TRUE then the gatekeeper has pre-granted permission to the endpoint to answer calls without first sending an ARQ. If the **answerCall** flag is FALSE, the endpoint shall always send ARQ to get permission to answer a call.

useGKCallSignalAddressToAnswer - If the **answerCall** and **useGKCallSignalAddressToAnswer** flags are both set to true, then when an endpoint does not send an ARQ to the gatekeeper to answer a call, the endpoint shall ensure that all H.225.0 call signalling comes from the gatekeeper. If an endpoint has been instructed to use the gatekeeper when answering, but it does not know whether an incoming call has come from the gatekeeper (which may involve looking at the transport address), the endpoint shall issue ARQ irrespective of the state of the **useGKCallSignalAddressToAnswer** flag.

irrFrequencyInCall - Indicates the frequency in seconds of IRR messages sent to gatekeeper when the endpoint is in one or more calls. If it is not present, the gatekeeper does not want unsolicited IRR messages. When the endpoint is sending these IRR messages, the call reference value shall be made unique for the terminal, as it would have been generated in an Admission Request. However, this is not a "normal" crv, and can not be reused for further communication (DRQ, IRQ or BRQ). The call identifier shall be the same as used in the call signalling channel messages for the related call.

[End Correction]

[Begin Correction]

15 ANNEX H – H.225.0 Message Syntax (ASN.1)

...

```
RegistrationConfirm ::= SEQUENCE --(RCF)
{
    requestSeqNum               RequestSeqNum,
    protocolIdentifier          ProtocolIdentifier,
    nonStandardData             NonStandardParameter OPTIONAL,
    callSignalAddress           SEQUENCE OF TransportAddress,
    terminalAlias               SEQUENCE OF AliasAddress OPTIONAL,
    gatekeeperIdentifier        GatekeeperIdentifier OPTIONAL,
    endpointIdentifier          EndpointIdentifier,
    ...,
    alternateGatekeeper         SEQUENCE OF AlternateGK OPTIONAL,
    timeToLive                  TimeToLive OPTIONAL,
    tokens                      SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens                SEQUENCE OF CryptoH323Token OPTIONAL,
    integrityCheckValue         ICV OPTIONAL,
    willRespondToIRR            BOOLEAN,
    preGrantedARQ               SEQUENCE
    {
```

```
        makeCall                                BOOLEAN,
        useGKCallSignalAddressToMakeCall        BOOLEAN,
        answerCall                              BOOLEAN,
        useGKCallSignalAddressToAnswer          BOOLEAN,
        ...,
        irrFrequencyInCall                      INTEGER (1..65535) OPTIONAL
    } OPTIONAL
}
```

[End Correction]

6.2.17 SETUP message

Description: The introduction of the pregranted admission with H.225.0 version 2 has left the gatekeeper in some calling scenarios with less information than needed, to identify which endpoint is placing the call.

When placing gatekeeper-routed calls, endpoints should populate the endpointIdentifier structure with the identifier given in the RegistrationConfirm message.

The clarified text will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.225.0 should be amended with the clarifying paragraph shown below.

This change may affect the functionality or local operation of an endpoint or Gatekeeper.

[Begin Correction]

7.3.10 Setup

...

mediaWaitForConnect – If TRUE, indicates that the recipient of the Setup message shall not transmit media until sending the Connect message.

canOverlapSend – If TRUE, indicates that the sender of Setup shall support overlap sending.

endpointIdentifier - This is an endpoint identifier that was assigned to the terminal in the RCF message. This field shall be present when the SETUP is sent towards the gatekeeper where the endpoint is registered, and shall not be present when the setup is sent to any other entity.

[End Correction]

[Begin Correction]

15 ANNEX H – H.225.0 Message Syntax (ASN.1)

...

```
Setup-UUIE ::= SEQUENCE
{
    protocolIdentifier              ProtocolIdentifier,
    h245Address                     TransportAddress OPTIONAL,
    sourceAddress                   SEQUENCE OF AliasAddress OPTIONAL,
    sourceInfo                      EndpointType,
    destinationAddress              SEQUENCE OF AliasAddress OPTIONAL,
    destCallSignalAddress           TransportAddress OPTIONAL,
    destExtraCallInfo               SEQUENCE OF AliasAddress OPTIONAL,        -- Note 1
    destExtraCRV                    SEQUENCE OF CallReferenceValue OPTIONAL,  -- Note 1
    activeMC                        BOOLEAN,
    conferenceID                    ConferenceIdentifier,
    conferenceGoal                  CHOICE
    {
        create                      NULL,
        join                        NULL,
        invite                      NULL,
        ...,`
        capability-negotiation          NULL,
        callIndependentSupplementaryService         NULL
    },
    callServices                    QseriesOptions OPTIONAL,
    callType    CallType,
    ...,
    sourceCallSignalAddress         TransportAddress OPTIONAL,
    remoteExtensionAddress          AliasAddress OPTIONAL,
    callIdentifier                  CallIdentifier,
    h245SecurityCapability          SEQUENCE OF H245Security OPTIONAL,
    tokens                          SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens                    SEQUENCE OF CryptoH323Token OPTIONAL,
    fastStart                       SEQUENCE OF OCTET STRING OPTIONAL,
    mediaWaitForConnect             BOOLEAN,
    canOverlapSend                  BOOLEAN,
    endpointIdentifier              EndpointIdentifier OPTIONAL
}
```

[End Correction]

6.2.18 Support for SET Devices

Description: After H.225.0 (1998) was decided, it became apparent that an EndpointType field was needed to support the SET series of devices that were under development. This amendment represents a backward-compatible addition to the current ASN.1 and H.225.0 text that will be included in H.225.0 Version 3.

[Begin Correction]

7.6 H.225.0 Common Message Elements

...

The **EndpointType** structure conveys information about the H.323 element at the end of the signalling link. The H.323 element would complete one or more of the **gatekeeper**, **gateway**, **mcu**, or **terminal** message elements. If the H.323 element has an MC, then the **mc** Boolean would be true. Presence of the **set** component indicates that the entity is a Simple Endpoint Type (SET) device as defined in H.323 Annex F among others. The bit positions in the set component indicate the type of SET device; their meaning is defined in Annex F and other Recommendations that specify SET device types.

[End Correction]

[Begin Correction]

```
EndpointType ::= SEQUENCE
{
    nonStandardData     NonStandardParameter OPTIONAL,
    vendor              VendorIdentifier OPTIONAL,
    gatekeeper          GatekeeperInfo OPTIONAL,
    gateway             GatewayInfo OPTIONAL,
    mcu                 McuInfo OPTIONAL,           -- mc must be set as well
    terminal            TerminalInfo OPTIONAL,
    mc                  BOOLEAN,                    -- shall not be set by itself
    undefinedNode       BOOLEAN,
    ...,
    set                 BIT STRING (SIZE(32)) OPTIONAL
                                                    -- shall not be used with mc, gatekeeper
                                                    -- code points for the various SET devices
                                                    -- are defined in the respective SET
                                                    -- Annexes
}
```

[End Correction]

6.2.19 Use of Alternate Gatekeepers

Description: Issues have been raised with the implementation of alternate gatekeepers that warrant clarification. These clarifications are outlined below.

The clarified text will be contained in the revision 3 of H.225.0 Recommendation to be published by the ITU-T. However, the current text in revision 2 of H.225.0 should be amended with the clarifying paragraph shown below.

The below correction shall be applied to sections 7.8.3, 7.9.3, 7.10.3, 7.11.3, 7.12.3, 7.13.37.14.3, and 7.15.3

[Begin Correction]

.**altGKInfo** - optional information about alternative gatekeepers. If this information is supplied, an endpoint should retransmit the request to one of the alternate gatekeepers listed. If an alternate gatekeeper rejects the request, the endpoint shall accept the rejection. If an alternate gatekeeper does not respond, the endpoint may send the request to another alternate in the list.

[End Correction]

[Begin Correction]

7.6 H.225.0 Common Message Elements

...

The **AltGKInfo** structure is used to provide information about alternate gatekeepers:

– **alternateGatekeeper** - sequence of prioritised alternateGatekeeper for gatekeeperIdentifer and rasAddress for client to retry the request.

– **altGKisPermanent** - TRUE if all future RAS signals should be redirected to an address from alternateGatekeeper, FALSE if only the message that caused the Reject should be redirected

A gatekeeper may send an endpoint a list of alternate gatekeepers in various messages. When communicating with its gatekeeper, an endpoint that implements the alternate gatekeeper mechanism shall replace any previously received list of alternate gatekeepers with the most recently received list of alternate gatekeepers. It is possible for an alternate gatekeeper to send a list of alternate gatekeepers. If an endpoint sends a request to an alternate gatekeeper that will potentially become its permanent gatekeeper, it shall accept the new list of alternate gatekeepers. Otherwise, if the alternate gatekeeper will not potentially become its permanent gatekeeper, any list of alternate gatekeepers received shall be ignored. A gatekeeper may potentially become an endpoint's permanent gatekeeper if either the current gatekeeper becomes unresponsive or if the altGKisPermanent flag is set to TRUE in the **AltGKInfo** structure.

[End Correction]

6.2.20 Support for Caller Identification

Description: H.225.0 (1996) and H.225.0 (1998) indicated that Octet 3a of the CallingParty IE in H.225.0 shall not be present. However, it has become clear that this restriction was made in error and corrected text is shown below. This amendment will also be made to H.225 version 3.

[Begin Correction]

7.1 Use of Q.931

Each H.225.0 endpoint shall be able to interpret and generate the information elements mandated in the following for the respective Q.931 and H.450 messages. It may interpret and generate the optional information elements as defined below as well. It also may interpret other information elements of Q.931, or other Q-series or H.450 protocols. The endpoints shall be able to ignore unknown information elements contained in a Q.931 or H.450 message without disturbing operation. Procedures for receiving unrecognized "comprehension required" information elements shall apply according to clause 5.8.7.1 of Q.931. Endpoints shall ignore optional information elements with content error contained in a Q.931 message without disturbing operation.

[End Correction]

[Begin Correction]

7.2.2.6 Calling Party Number

...

Octet #3a

~~– Shall not be present.~~

– Encoded following the values and rules of Table 4-11/Q.931.

...

Connected Number

– Encoded following clause 5.4.1 of Q.951.

Connected Sub-address

– Encoded following clause 5.4.2 of Q.951.

Also in support of caller ID services, the Connected Number and Connected Sub-address information elements should be allowed in the Connect message. The Connected Number IE shall be encoded following clause 5.4.1 of Q.951. The Connected Sub-address IE shall be encoded following clause 5.4.2 of Q.951.

[End Correction]

[Begin Correction]

7.3.3 Connect

...

Information element	**H.225.0 status(M/F/O)**	**Length in H.225.0**
Protocol discriminator	M	1
Call reference	M	3
Message type	M	1
Bearer capability	O (Note 1)	5-6
Extended facility	O	8-*
Channel identification	FFS	NA
Facility	O	8-*
Progress indicator	O	2-4
Notification Indicator	O	2-*
Display	O	2-82
Date/Time	O	8
High layer compatibility	FFS	NA
Low layer compatibility	FFS	NA
User-to-User	M	2-131
Connected number	O	2-*
Connected sub-address	O	2-23

[End Correction]

6.3 Technical and Editorial Corrections to ITU-T Recommendation H.245

6.3.1 H.2250LogicalChannelAckParameters

Description: A missing field in the LogicalChannelAck corresponding the ATM virtual circuit issues raised in section 0 of this document.

The corrected ASN.1 is shown below.

[Begin Correction]

...

```
H2250LogicalChannelAckParameters        ::=SEQUENCE
{
    nonStandard                 SEQUENCE OF NonStandardParameter OPTIONAL,
    sessionID                       INTEGER(1..255) OPTIONAL,
    mediaChannel                TransportAddress OPTIONAL,
    mediaControlChannel         TransportAddress OPTIONAL,          -- forward RTCP channel
    dynamicRTPPayloadType       INTEGER(96..127) OPTIONAL,
```

```
                                                    -- used only by the master or MC
        ...,
        flowControlToZero           BOOLEAN,
        portNumber                  INTEGER (0..65535) OPTIONAL
}
```

...

[End Correction]

6.3.2 H.320/H.323 Continuous Presence

Description: A minor inconsistency has been discovered in the Recommendation H.245 concerning H.320 continuous presence operation.

The H.245 equivalent continuous presence BAS codes were not included in H.245v3 so continuous presence processing cannot be translated through an H.320-H.323 gateway. To correct this, the following ASN.1 should be included with H.245v3.

This information will be contained in the revision 4 of H.245 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.245v3 document that was submitted for approval in 1998.

[Begin Correction]

...

```
 TerminalYouAreSeeingInSubPictureNumber ::= SEQUENCE
 {
       terminalNumber              TerminalNumber,
       subPictureNumber    INTEGER (0..255),
       …
 }

 VideoIndicateCompose ::= SEQUENCE
 {
       compositionNumber   INTEGER (0..255),
       …
 }

ConferenceIndication       ::=CHOICE
{
       sbeNumber                   INTEGER (0..9),         -- same as H.230 SBE Number

       terminalNumberAssign        TerminalLabel,          -- same as H.230 TIA

       terminalJoinedConference    TerminalLabel,          -- same as H.230 TIN

       terminalLeftConference      TerminalLabel,          -- same as H.230 TID

       seenByAtLeastOneOther       NULL,                   -- same as H.230 MIV
       cancelSeenByAtLeastOneOther NULL,                   -- same as H.230 cancel MIV
```

```
        seenByAll                   NULL,                   -- like H.230 MIV
        cancelSeenByAll             NULL,                   -- like H.230 MIV

        terminalYouAreSeeing        TerminalLabel,          -- same as H.230 VIN

        requestForFloor             NULL,                   -- same as H.230 TIF

        ...,
        withdrawChairToken          NULL,                   -- same as H.230 CCR
                                                            -- MC-> chair
        floorRequested              TerminalLabel,          -- same as H.230 TIF
                                                            -- MC-> chair
    terminalYouAreSeeingInSubPictureNumber TerminalYouAreSeeingInSubPictureNumber,
    videoIndicateCompose            VideoIndicateCompose
}

                                        ...
ConferenceCapability    ::=SEQUENCE
{
        nonStandardData             SEQUENCE OF NonStandardParameter OPTIONAL,
        chairControlCapability      BOOLEAN,
        ...
    VideoIndicateMixingCapability       BOOLEAN

}

                                        ...
```

[End Correction]

6.3.3 Conference definitions

Description: Minor omissions concerning conference related definitions have been discovered.

This information will be contained in the revision 4 of H.245 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.245v3 document that was submitted for approval in 1998.

These changes should not affect implementations.

[Begin Correction]

B.13.7 Conference Indications

terminalYouAreSeeingInSubPictureNumber shall be defined as H.230 VIN2. subPictureNumber is defined as N as indicated in Figures 2-4/H.243.

videoIndicateCompose shall be defined as H.230 VIC. compositionNumber is defined as M in Table 4/H.243.

[End Correction]

[Begin Correction]

B.2.2.9 Conference Capabilities

videoIndicateMixingCapability shall be defined as H.230 VIM

[End Correction]

6.3.4 Terminal Capabilities

Description: The Fast Connect procedure allows endpoints to determine certain capabilities of peer endpoints in the absence of the full disclosure provided by the H.245 capabilities exchange procedure. This should be reflected in H.245 section 7.2.1, (terminal capability messages) Overview and section 7.3.1, Open Logical Channel under the description of the dataType parameter, where the following text should be added

[Begin Correction]

7.3.1 Open Logical Channel

...

Terminals capable only of unidirectional (transmit or receive) operation on media types which make use of bi-directional channels shall send capabilities only for the supported direction of operation. The reverse direction shall use the nullData type, for which no capability is necessary. Transmit-only terminals should send transmit capabilities, but terminals should not assume that the absence of transmit capabilities implies that transmit-only operation is not possible.

If an endpoint has not indicated non-null receive capabilities through a Terminal Capability Set message, capabilities not previously indicated by the endpoint may be contained in the dataType parameter. In this case, the receiving entity shall consider that the sending endpoint has the specified capabilities, and furthermore, simultaneous capabilities may be indicated by the dataType of any logical channels associated with this request through the use of the associatedSessionID parameter.

...

[End Correction]

[Begin Correction]

7.2.1 Overview

...

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

If an endpoint has not indicated non-null receive capabilities through a Terminal Capability Set message, capabilities not previously indicated by the endpoint may be contained in the dataType parameter of an Open Logical Channel message. In this case, the receiving entity shall consider that the sending endpoint has the specified capabilities, and furthermore, simultaneous capabilities may be indicated by the dataType of any logical channels associated with the Open Logical Channel request through the use of the associatedSessionID parameter.

[End Correction]

6.3.5 Clarification of OLCs within the Context of Fast Start

Description: Sections 7.3.1 and 7.3.2 of H.245 discuss the Open Logical Channel procedures. However, these sections were not properly updated to reflect changes introduced with the introduction of Fast Start.

The modified text will be contained in the revision 6 of H.245 Recommendation to be published by the ITU-T. However, the current text in revision 3 of H.323 should be amended with the paragraph shown below.

[Begin Correction]

7.3.1 Open Logical Channel

...

The mediaChannel indicates a transportAddress to be used for the logical channel. When the transport is unicast, mediaChannel~~It~~ is not present in the OpenLogicalChannel ~~message when the transport is unicast~~ forwardLogicalChannelParameters, but may be present in the reverseLogicChannelParameters. If the transportAddress is multicast, the master is responsible for creating the multicast transport address and shall include the address in the OpenLogicalChannel message. A slave entity that wishes to open a new multicast channel will provide zeroes in the multicast transportAddress field. The master will create and provide the multicast transportAddress in the OpenLogicalChannelAck message for the slave entity. Note that the MC will use the communicationModeCommand to specify the details about all the RTP Sessions in the conference.

[End Correction]

[Begin Correction]

7.3.2 Open Logical Channel Acknowledge

...

The mediaChannel indicates a transportAddress to be used for the logical channel. It shall be present in the OpenLogicalChannelAck message when the transport is unicast except where the OpenLogicalChannel request specified a reverse unicast mediaChannel. If the transportAddress is multicast, the master is responsible for creating the multicast transport address and shall include the address in the OpenLogicalChannel message. A slave entity that wishes to open a new multicast channel will provide zeroes in the multicast transportAddress field. The master will create and provide the multicast transportAddress in the OpenLogicalChannelAck message for the slave entity. Note that the MC will use the communicationModeCommand to specify the details about all the RTP Sessions in the conference.

[End Correction]

6.4 Technical and Editorial Corrections to ITU-T Recommendation H.246

6.4.1 Annex A Corrections

Description: A minor inconsistency has been discovered in the Recommendation H.246 Annex A section A.5.2.4.1.

The commands MCV and Cancel-MCV are listed with a H.245 equivalent of broadcastMe and cancelBroadcastMe. The H.245 equivalent of these messages should have been listed as the ConferenceCommands broadcastMyLogicalChannelNumber and cancelBroadcastMyLogicalChannel. (Note: There is also a H.245 ConferenceRequest to broadcastMyLogicalChannelNumber that provides for a response.)

This information will be contained in the revision 2 of H.246 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.246 document that was submitted for approval in 1998.

This change should not affect behaviour in any way.

[Begin Correction]

A.5.2.4.1 Multipoint Control C&I

H.230 command/indication	**H.245 equivalent**
MCV	Send **broadcast~~Me~~MyLogicalChannel**
Cancel-MCV	Send **cancelBroadcast~~Me~~MyLogicalChannel**

[End Correction]

Description: A minor inconsistency has been discovered in the Recommendation H.246 Annex A section A.5.2.4.4.

The H.245 equivalent continuous presence BAS codes were not included in H.245v3 so continuous presence processing cannot be translated through a H.320-H.323 gateway. To correct this, commands are added to H.245 and the following corrected translations amend H.246.

This information will be contained in the revision 2 of H.246 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.246 document that was submitted for approval in 1998.

This change should may affect behaviour of gateways.

[Begin Correction]

A.5.2.4.4 Video Selection and Notification C&I

H.230 command/indication	**H.245 equivalent**
VIN	Send **terminalYouAreSeeing**
VCB/Cancel-VCB	Send **makeTerminalBroadcaster/ CancelMakeTerminalBroadcaster**
VCS/Cancel-VCS	Send **sendThisSource/ CancelSendThisSource**
VCR	Send **videoCommandReject**
~~VIN2~~	~~FFS~~Send **~~terminalYouAreSeeingInSubPictureNumber~~**

VIN2	~~FFS~~Send **terminalYouAreSeeingInSubPictureNumber**
VIC	~~FFS~~send **videoIndicateCompose**
VIM	~~FFS~~send **videoIndicateMixingCapability**

[End Correction]

6.4.2 Reference to ATM Forum document

Description: To help clarify the usage of H.246 with respect to ATM, a reference to an ATM Forum document has been proposed. This reference shall appear in next H.246 publication from the ITU.

[Begin Correction]

1 Scope

...

Voice/Voiceband terminals on GSTN use the appropriate national standards for call control and G.711 or analogue signals for voice. Voice/Voiceband terminals on ISDN use the appropriate national variant of Q.931 for call control and G.711 for voice.

Interworking of H.323 over ATM with H.323 over non-ATM IP networks is possible through the use of an H.323-H.323 gateway. Transport of H.323 media streams over ATM is described in AF-SAA-0124.000.

[End Correction]

[Begin Correction]

2 Normative References

...

- ATM Forum Technical Committee, AF-SAA-0124.000, *Gateway for H.323 Media Transport Over ATM*, 1999.

[End Correction]

6.5 Technical and Editorial Corrections to ITU-T Recommendation H.235

6.5.1 Key Escrow usage

Description: A minor inconsistency has been discovered in the Recommendation H.235 Section 6.6.1.

This information will be contained in the revision 2 of H.235 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.235 document that was submitted for approval in 1998.

This change does not affect behaviour or implementations in any way.

[Begin Correction]

6.6.1 Key Escrow

Although not specifically required for operation, this recommendation contains provision for entities utilizing the H.235 protocol to support ~~key recovery~~ the facility known as trusted third party (TTP) within the signalling elements.

[End Correction]

6.5.2 H.235 Control Channel references

Description: A typographical error has been discovered in section 8 of the Recommendation H.235.

This information will be contained in the revision 2 of H.235 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.235 document that was submitted for approval in 1998.

This change does not affect intended behaviour or implementations in any way, the uncorrected text is misleading and in error.

[Begin Correction]

8.2 Unsecured H.245 Channel Operation

Alternatively, the H.245 channel may operate in an unsecured manner and the two entities open a secure logical channel with which to perform authentication and/or shared-secret derivation. For example TLS or IPSEC may be utilized by opening a logical channel with the datatype containing a value for **~~encryption~~Datah235Control**. This channel could then be used to derive a shared secret which protects any media session keys or to transport the **EncryptionSync**.

[End Correction]

6.5.3 Multipoint procedure section reference

Description: A minor section reference has been discovered in the Recommendation H.235 Section 9.

This information will be contained in the revision 2 of H.235 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.235 document that was submitted for approval in 1998.

[Begin Correction]

9 Multipoint Procedures

9.1 Authentication

Authentication shall occur between an endpoint and the MC(U) in the same manner that it would in a point to point conference. The MC(U) shall set the policy concerning level and stringency of authentication. As stated in section~~0~~ 6.6 the MC(U) is trusted; existing endpoints in a conference may be limited by the authentication level employed by the MC(U). New **ConferenceRequest**/**ConferenceResponse** commands, allow endpoints to obtain the certificates of other participants in the conference from the MC(U). As outlined in H.245 procedures, endpoints in a multipoint conference may request other endpoint certificates via the MC, but may not be able perform direct cryptographic authentication within the H.245 channel.

...

[End Correction]

6.5.4 Introduction to Authentication

Description: The introductory text (paragraph 1) to Section 10 of revision 1 of H.235 Recommendation has been determined to be unclear and potentially misleading.

This text will be corrected in the revision 2 of H.235 Recommendation to be published by the ITU-T. However, this text appears in the final H.235 document that was submitted for approval in 1998.

This change should not effect implementations or operations, but readers are encouraged to utilize the following text.

[Begin Correction]

10.1 Introduction

Authentication is in general based either on using a shared secret (you are authenticated properly if you know the secret) or on public key based methods with certifications (you prove your identity by possessing the correct private key). A shared secret and the subsequent use of symmetric cryptography requires a prior contact between the communicating entities. A prior face-to-face or secure contact can be replaced by generating or exchanging the shared secret key with methods based on public key cryptography, e.g. by Diffie-Hellman key exchange. The communication parties in the key generation and exchange have to be authenticated for example by using digitally signed messages; otherwise the communication parties cannot be sure with whom they share the secret.

This Recommendation presents authentication methods based on subscription, i.e. there must be a prior contact for sharing a secret, and authentication methods where public key cryptography is directly used in authentication or it is used for generating the shared secret.

~~There are two types of authentication that may be utilized. The first type is symmetric encryption based that requires no prior contact between the communicating entities. The second type is based on the ability to have some prior shared secret (further referenced as 'subscription' based). Two forms of subscription-based authentication are provided; password and certificate.~~

[End Correction]

6.5.5 Diffie-Hellman exchange with optional Authentication

Description: Two errors have been discovered in the labeling of parameters of arguments in the Diffe-Hellman exchange described in the Recommendation H.235 Section 10.2. Additionally, the note concerning authentication is to be clarified.

This information will be contained in the revision 2 of H.235 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.235 document that was submitted for approval in 1998.

Phase 1: As this correction affects implementations, which utilize this mechanism to provide authentication during the Diffe-Hellman exchange. Note that if these optional parameters are not utilized (denoted by italics below and in the original recommendation) no implementation changes are needed.

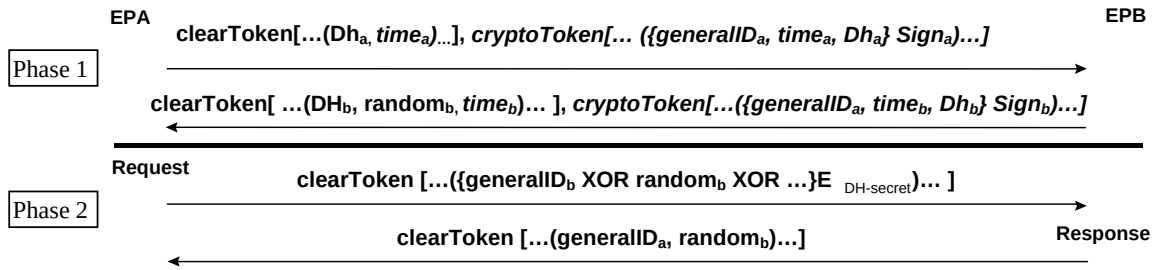
Phase 2: The identifier (generalID) passed from in the second exchange (e.g. Response) should be that of the recipient of the Response message (e.g. EPA).

[Begin Correction]

10.2 Diffie-Hellman with optional Authentication

…

NOTE - If the messages are exchanged over an insecure channel, then digital signatures (or other message origin authentication method) must be used in order to authenticate the parties between whom the secret will be shared. An optional signature element may also be provided these are illustrated in ***italics*** below.



…

[End Correction]

6.5.6 Introduction to Subscription based Authentication

Description: The introductory text (paragraph 1) to Section 10.3 of revision 1 of H.235 Recommendation has been determined to be unclear and potentially misleading. The included text should be added as a new, final paragraph.

This text will be corrected in the revision 2 of H.235 Recommendation to be published by the ITU-T. However, this text appears in the final H.235 document that was submitted for approval in 1998.

This change should not effect implementations or operations, but readers are encouraged to utilize the following text.

[Begin Correction]

10.3.1 Introduction

…

NOTE - In all cases where timestamps are generated and passed as part of a security exchange, implementers should take the following precautions. The time stamp granularity should be fine enough that it is guaranteed to increment with each message. If this is not guaranteed, replay attacks are possible. (e.g. if the timestamp only increments by the minute, then an endpoint 'C' can spoof endpoint 'A' within duration of one minute after endpoint 'A' has sent a message to endpoint 'B').

…

[End Correction]

6.5.7 Password with Hashing

Description: The text to Section 10.3.3 of revision 1 of H.235 Recommendation has been determined to be unclear with respect to parameters that are passed in the exchange of messages. The included text should be added as a new, final paragraph.

This text will be corrected in the revision 2 of H.235 Recommendation to be published by the ITU-T. However, this text appears in the final H.235 document that was submitted for approval in 1998.

This change should not effect implementations or operations, but readers are cautioned to take into account the following text.

[Begin Correction]

10.3.3 Password with Hashing

...

Note 3 - The **cryptoHashedToken** structure is used to pass the parameters used in this exchange. Included in this structure are the 'clear' versions of parameters needed to compute the hashed value. Implementers should include the timestamp in the **hashedVals** and should ***not*** include the password. (E.g. both the password and the 'generalID' should be known *a priori* by the recipient).

Note 4 - The hashing function shall be applied to the **EncodedGeneralToken** structure that includes at least the ID, timestamp and password fields. The password value should NOT be passed in the ClearToken.

...

[End Correction]

6.5.8 Corrections to ANNEX A

Description: An omission in the ASN.1 syntax for H.235 has been discovered. Specifically, an identifier is missing from the ClearToken structure in the case where the ClearToken structure is placed directly into the message.

This information will be contained in the revision 2 of Recommendation H.235 to be published by the ITU-T. However, this information appears incorrectly in the final H.235 document that was submitted for approval in 1998.

The absence of this identifier will not allow multiple ClearTokens included in a single RAS message to be associated with individual uses. Additionally, ClearTokens may be defined for different uses that have the same format and these need to be differentiated by the **tokenOID**.

[Begin Correction]

```
ClearToken      ::= SEQUENCE -- a `token' may contain multiple value types.
{
    tokenOID                OBJECT IDENTIFIER,
    timeStamp               TimeStamp OPTIONAL,
    password                Password OPTIONAL,
    dhkey                   DHset OPTIONAL,
    challenge               ChallengeString OPTIONAL,
    random                  RandomVal OPTIONAL,
    certificate             TypedCertificate OPTIONAL,
    generalID               Identifier OPTIONAL,
    nonStandard             NonStandardParameter OPTIONAL,
    ...
}

--  An object identifier should be placed in the tokenOID field when a
--  ClearToken is included directly in a message (as opposed to being
--  encrypted). In all other cases, an application should use the
--  object identifier { 0 0 } to indicate that the tokenOID value is not present.
```

[End Correction]

6.5.9 Corrections to ANNEX B

Description: A number of typographical errors have been discovered in Annex B. Their corrected values are shown below.

This information will be contained in the revision 2 of Recommendation H.235 to be published by the ITU-T. However, this information appears incorrectly in the final H.235 document that was submitted for approval in 1998.

[Begin Correction]

2 Signalling and Procedures

...

One purpose of H.225.0 exchanges as they relate to H.323 security, is to provide a mechanism to set up the secure H.245 channel. Optionally, authentication may occur during the exchange of H.225.0 messages. This authentication may be certificate or password based, utilizing encryption and/or hashing (i.e. signing). The specifics of these modes of operation are described in sections ~~(0~~ (4.2-4.3) ~~04.2-4.3)~~

...

[End Correction]

[Begin Correction]

4.1 Introduction

This annex will not explicitly provide any form of message privacy between gatekeepers and endpoints. There are two types of authentication that may be utilized. The first type is symmetric encryption based that requires no prior contact between the endpoint and Gatekeeper. The second type is subscription based and will have two forms, password or certificate. All of these forms are derived from the procedures shown in sections ***[change these to document cross-references]*** 10.2, 10.3.2, 10.3.3 and 10.3.4. In this annex, the generic labels (EPA and EPB) showed in the aforementioned sections will represent the Endpoint and Gatekeeper respectively.

...

[End Correction]

[Begin Correction]

4.2 Endpoint-Gatekeeper Authentication (Non-Subscription Based)

This mechanism may provide the Gatekeeper with a cryptographic link that a particular endpoint, which previously registered, is the same one that issues subsequent RAS messages. It should be noted that this might not provide any authentication of the Gatekeeper to the endpoint, unless the optional signature element is included. The establishment of the identity relationship occurs when the terminal issues the **GRQ** as outlined in H.323 section ***[change to cross-reference]* 7.2.1**. The Diffie-Hellman exchange shall occur in conjunction with the **GRQ** and **GCF** messages as shown in the first phase of section 0. This shared secret key shall now be used on any subsequent **RRQ**/**URQ** from the terminal to the gatekeeper. If a Gatekeeper operates in this mode and receives a **GRQ** without a token containing the *DHset* or an acceptable algorithm value, it shall return a **securityDenial** reason code in the **DRJ**.

Terminal (**xRQ**):

1) The terminal shall provide all of the information in the message as described in the appropriate H.225.0 sections.

2) The terminal shall encrypt the **GatekeeperIdentifier** (as returned in the **GCF**) using the shared secret key that was negotiated. This shall be passed in a~~**cryptoToken**~~ **clearToken (see section 10.2)** as the **generalID**.

The 16 bits of the **random** and then the **requestSeqNum** shall be XOR'd with each 16 bits of the **GatekeeperIdentifier**. If the **GatekeeperIdentifier** does not end on an even 16 boundary, the last 8 bits of the **GatekeeperIdentifier** shall be XOR'd with the least significant octet of the random value and then **requestSeqNum.** The **GatekeeperIdentifier** shall be encrypted using the selected algorithm in the **GCF** (~~integrity~~algorithmOID) and utilizing the entire shared secret.

The following example illustrates this procedure:

RND16: 16 bit value of the Random Value

SQN16: 16 bit value of requestSeqNum

BMPX: the Xth BMP character of GatekeeperIdentifier

BMP1' = (BMP1) XOR (RND16) XOR (SQN16)

BMP2' = (BMP2) XOR (RND16) XOR (SQN16)

BMP3' = (BMP3) XOR (RND16) XOR (SQN16)

BMP4' = (BMP4) XOR (RND16) XOR (SQN16)

BMP5' = (BMP5) XOR (RND16) XOR (SQN16)

:

:

BMPn' = (BMPn) XOR (RND16) XOR (SQN16)

...

[End Correction]

[Begin Correction]

5.1 Gateway

As stated in section ***[change to cross reference]* 6.6**, an H.323 Gateway should be considered a trusted element. This includes protocol gateways (H.323-H.320 etc.…) and security gateways (proxy/firewalls). The media privacy can be assured between the communicating endpoint and the gateway device; but what occurs on the far side of the gateway should be considered insecure by default.

[End Correction]

6.5.10 Corrections to Appendix I

Description: A typographical error has been discovered with respect to a section reference of encryption key generation.

This information will be contained in the revision 2 of H.235 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.235 document that was submitted for approval in 1998.

The referenced section was incorrectly labelled to a non-existent section heading.

[Begin Correction]

4.2 Password

...

The encryption key is constructed from the user's password using the procedure described in section 1~~3.3.3.341~~0.3.2 of H.235. The resulting octet "string" is then used as the DES key to encrypt the **challenge**.

...

[End Correction]

6.6 Technical and Editorial Corrections to ITU-T H.450-series Recommendations

6.6.1 H.450.1 Editorial Corrections

Description: Typographical errors have been discovered in clause 6.6 of H.450.1.

This information will be contained in the revision 2 of H.450.1 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.450.1 document that was submitted for approval in 1998. These changes do not affect behaviour or implementations in any way.

1) Editorial, Clause 6.6, line 6

Change:

"rejectUnrecognizedInvokePdu"

to

"reject<u>Any</u>UnrecognizedInvokePdu"

2) Editorial, Clause 6.6, line 12

Change:

"discardAnyUnrecognizedInvokePDU"

to

"discardAnyUnrecognizedInvokeP<u>du"</u>

6.6.2 H.450.2 Editorial Corrections

Description: Typographical errors have been discovered in H.450.2 clauses 11.4.2, 11.5.2, 11.6.2 and 13.4.

This information will be contained in the revision 2 of H.450.2 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.450.2 document that was submitted for approval in 1998. These changes do not affect behaviour or implementations in any way.

1) Editorial - Clause 11.4.2, line 4 c)

Change:

"The CTSetup.request primitive is used to request call establishment from TRTSE."

to

"The CTSetup.request primitive is used to request call establishment to TRTSE""

2) Editorial - Clause 11.4.2, line 5 d)

Change:

"The CTSetup.confirm primitive is used to indicate success of call establishment to TRTSE."

to

"The CTSetup.confirm primitive is used to indicate success of call establishment from TRTSE."

3) Editorial - Clause 11.5.2, line 6 e)

Change:

"The CTIdentify.indication primitive is used to request a call identification."

to

"The CTIdentify.indication primitive is used to indicate a call identification."

4) Editorial - Clause 11.5.2, line 11,12 j)

Change:

"The CTComplete.request primitive may be used by GKs to request sending of call transfer information to the transferred-to user."

to

"The CTComplete.request primitive may be used by GKs to request sending of call transfer information to the transferred-to endpoint."

5) Editorial - Clause 11.5.2, line 13,14 k)

Change:

"The CTComplete.indication primitive is used to indicate call transfer information to the transferred-to endpoint."

to

"The CTComplete.indication primitive is used to indicate call transfer information to the transferred-to user."

6) Editorial - Clause 11.6.2, line 2

Change:

"CT-T1 - Timer CT-T1 shall operate at the TRGSE during state CT-Await-Identify-Response. Its purpose is to protect against the absence of response to the CTIdentify.request"

to

"CT-T1 - Timer CT-T1 shall operate at the TRGSE during state CT-Await-Identify-Response. Its purpose is to protect against the absence of response to the CTIdentify.invoke."

7) Editorial – Clause 13.4, FIGURE 25 (sheet 2 of 3, 4th branch) of H.450.2

(i.e. FIGURE 22/H.450.2 (sheet 2 of 3, 4th branch) of H.450.2 (2/98) publication)

Change "T4 Timeout" to "CT-T4 Timeout".

In addition, the type of symbol was mistake. Time-Out event is an internal event.

6.6.3 H.450.2 Clarification of CallIdentifier and ConferenceIdentifier

Description: A clarification of the setting of H.225.0 elements CallIdentifier and ConferenceIdentifier values in conjunction with H.450.2 transferred calls has been added within a new clause 10.7 "Interactions with H.225.0 parameters".

This information will be contained in the revision 2 of H.450.2 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.450.2 document that was submitted for approval in 1998. These changes do not affect behaviour or implementations in any way.

[Begin Addition]

10.7 Interactions with H.225.0 parameters

The H.225.0 CallIdentifier value of the transferred call shall be set to the value that was used in the primary call.

The H.225.0 ConferenceIdentifier of a transferred call may use a new value. However, the ConferenceIdentifier of an existing conference (multipoint conference) shall not be altered.

[End Addition]

6.6.4 H.450.2 Transfer without Consultation

Description: An exceptional procedure for a transferred endpoint B actions has been added in clause 8.2.1 to allow call transfer without consultation to take place successfully even if the transferred-to endpoint C does either not support H.450.2 or not support H.450 at all. Furthermore, clause 6 was enhanced to allow a different Interpretation APDU setting.

This information will be contained in the revision 2 of H.450.2 Recommendation to be published by the ITU-T.

[Begin Correction]

6 Messages and Information elements

...

When conveying the invoke APDU of operation callTransferSetup, the Interpretation APDU shall contain value clearCallIfAnyInvokePduNotRecognized in case of Transfer with Consultation. In case of Call Transfer without Consultation, the Interpretation APDU shall be set to value discardAnyUnrecognizedInvokePdu.

[End Correction]

[Begin Addition]

8.2.1 Transfer without Consultation with transferred-to endpoint C not supporting H.450.2

a) When receiving a CONNECT message from endpoint C (that does not include a response to the callTransferSetup Invoke APDU) while being in state CT-Await-Setup-Response, the transferred endpoint B should continue as if a callTransferSetup Return Result APDU would have been received. This allows endpoint B to successfully continue with the Call Transfer procedures (including appropriate internal call transfer state handling and clearing of the primary call to the transferring endpoint A). This exceptional procedure enables successful Call Transfer even if the transferred-to endpoint C does not support H.450 at all.

b) When a RELEASE COMPLETE message as a response to a SETUP message containing callTransferSetup Invoke APDU is received in endpoint B on the transferred call attempt, possibly containing callTransferSetup Return Error or Reject APDU, then endpoint B may retry call establishment to endpoint C using a normal basic call. Upon receiving the CONNECT message from endpoint C, endpoint B may continue with the procedures as described in a) above.

Note that this procedure may apply if endpoint C supports H.450.1 but no H.450.2 and if endpoint B has not selected the recommended Interpretation APDU value discardAnyUnrecognizedInvokePdu but has set the value to clearCallIfAnyInvokePduNotRecognized.

[End Addition]

6.6.5 H.450.3 Editorial Corrections

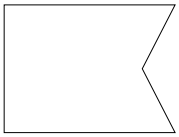
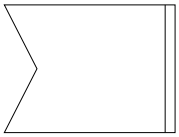
Description: Typographical errors have been discovered in H.450.3 clause 12 SDLs.

This information will be contained in the revision 2 of H.450.3 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.450.3 document that was submitted for approval in 1998. These changes do not affect behaviour or implementations in any way.

Editorial - Clause 12 SDL FIGURES 21 (most right branch), 22 (most right branch), 23 (most right branch), 28 (sheet 1 of 4, second right branch) of H.450.3
(i.e. FIGURES 19, 20, 21 and 24 (sheet 1 of 4) of H.450.3 of H.450.3 (2/98) published).

The type of symbol was mistake. Time-Out event is an internal event.

Note - The text within the referred symbols remains unchanged.

change  to 

6.6.6 H.450.3 Clarification of CallIdentifier and ConferenceIdentifier

Description: A clarification of the setting of H.225.0 elements CallIdentifier and ConferenceIdentifier values in conjunction with H.450.3 forwarded calls has been added within a new clause 9.9.3 "Interactions with H.225.0 parameters".

This information will be contained in the revision 2 of H.450.3 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.450.3 document that was submitted for approval in 1998. The additional text is shown below.

[Begin Addition]

9.9.3 Interactions with H.225.0 parameters

The H.225.0 CallIdentifier of a forwarded call shall be set to the value that was used in the forwarding call. The CallIdentifier value shall be preserved also during multiple call diversions.

The H.225.0 ConferenceIdentifier of a forwarded call may use a new value. However, the ConferenceIdentifier of an existing conference (multipoint conference) shall not be altered.

[End Addition]

6.6.7 H.450.3 ASN.1 Correction

Description: A typographical error has been discovered in the ASN.1 definitions presented in H.450.3, Chapter 11.

This information will be contained in the revision 2 of H.450.3 Recommendation to be published by the ITU-T. However, this information appears incorrectly in the final H.450.3 document that was submitted for approval in 1998.

The corrected, included element is shown below.

[Begin Correction]

H225InformationElement FROM H225-~~Generic~~generic-parameters-definition

...

[End Correction]

7 Implementation Clarifications

7.1 Token Usage in H.323 systems

There has been some confusion on the usage of individual **CryptoH323Tokens** as passed in RAS messages. There are two main categories of CryptoH323Tokens; those used for H.235 procedures and those used in an application specific manner. The use of these tokens should be according to the following rules:

- All H.235 defined (e.g. cryptoEPPwdHash, cryptoGKPwdHash, cryptoEPPwdEncr, cryptoGKPwdEncr, cryptoGKCert, and cryptoFastStart). shall be utilized with the procedures and algorithms as described in H.235.
- Application specific or proprietary use of tokens shall utilize the nestedcryptoToken for their exchanges.
- Any nestedcryptoToken used should have a tokenOID (object identifier) which unambiguously identifies it.

7.2 H.235 Random Value Usage in H.323 systems

The random value that is passed in xRQ/xCF sequence between endpoints and Gatekeepers may be updated by the Gatekeeper. As described in section 4.2 of H.235 this random value may be refreshed in any xCF message to be utilized by a subsequent xRQ messages from the endpoint. Due to the fact that RAS messages may be lost (including xCF/xRJ) the updated random value may also be lost. The recovery from this situation may be the reinitializing of the security context but is left to local implementation.

Implementations that require the use of multiple outstanding RAS requests will be limited by the updating of the random values used in any authentication. If the updating of this value occurs on every response to a request, parallel requests are not possible. One possible solution, is to have a logical 'window' during which a random value remains constant. This issue is a local implementation matter.

7.3 Gateway Resource Availability Messages

The Resources Available Indication (RAI) is a notification from a gateway to a gatekeeper of its current call capacity for each H-series protocol and data rate for that protocol. The gatekeeper responds with a Resources Available Confirmation (RAC) upon receiving a RAI to acknowledge its reception. A Gatekeeper should ignore any RAI notifications (e.g. send no RAC) upon receiving a RAI which contains bogus information (i.e. a bad endpointIdentifier).

7.4 OpenLogicalChannel in fastStart

In the H.225.0 ASN.1, fastStart is defined as SEQUENCE OF OCTET STRING OPTIONAL. The text definition states "This uses the OpenLogicalChannel structure defined in H.245…" Each OCTET STRING in fastStart is to contain the OpenLogicalChannel structure, not an entire request message.

7.5 Clarification in Q.931

Table 4-3/Q.931 (Information Element Identifier Coding) shows that the Progress Indicator IE identifier (like an opcode) shows 0x1e, but Figure 4-29/Q.931 (octet layout of Progress Indicator IE) shows the identifier as 0x1f. Note that the identifier should be 0x1e.

7.6 Graceful closure of TCP connection

When a TCP connection is closed, the graceful closure procedure documented in section 3.5 of RFC 793 should always be used.

7.7 Race condition on simultaneous close of channel

Section 8.5 of H.323 describes the procedures that an endpoint follows to terminate a call. It should be noted that as prescribed in Step 6, both endpoints might issue a Release Complete simultaneously. Endpoints should be prepared for this potential race condition.

8 Allocated Object Identifiers and Port Numbers

Information in this section is provided for informational purposes and convenience. This section does not supercede nor replace proper references in H.225.0, H.225, H.235, or other Recommendations.

8.1 Allocated Object Identifiers

The following object identifiers have been allocated for protocols associated with H.323. Any future object IDs which are allocated should be indexed here to prevent duplication.

Note that all object IDs below are allocated below the object ID { itu-t(0) recommendation(0) } which has been abbreviated as "0.0" below.

{ 0 0 h(8) 2250 version(0) *[v]* } Assigned values of *v*: 1-3	H225.0 version numbers
{ 0 0 h(8) 2250 annex(1) g(7) version(0) *[v]* } Assigned values of *v*: 1	H225.0 annex G version numbers
{ 0 0 h(8) 2250 annex(1) g(7) usage(1) *[u]* } Assigned values of *u*: none	H225.0 annex G usage tags
{ 0 0 h(8) 245 version(0) *[v]* } Assigned values of *v*: 1-6	H245 version numbers
{ 0 0 h(8) 245 generic-capabilities(1) video(0) *[c]* } Assigned values of *c*: is14496-2(0)	Generic video capabilities
{ 0 0 h(8) 245 generic-capabilities(1) audio(1) *[c]* } Assigned values of *c*: none	Generic audio capabilities
{ 0 0 h(8) 245 generic-capabilities(1) data(2) *[c]* } Assigned values of *c*: none	Generic data capabilities
{ 0 0 h(8) 245 generic-capabilities(1) control(3) *[c]* } Assigned values of *c*: Logical-channel-bit-rate-management(0)	Generic control capabilities
{ 0 0 h(8) 245 generic-capabilities(1) multiplex(4) *[c]* } Assigned values of *c*: none	Generic multiplex capabilities
{ 0 0 h(8) 283 generic-capabilities(1) 0}	H.283 Capability

8.2 Allocated Port Numbers

The following IP port numbers have been allocated:

1300 TLS secured call signalling

1718 Multicast RAS Signalling

1719 Unicast RAS Signalling

1720 TCP call signalling

2099 Annex G/H.225.0 Signalling

2517 Annex E/H.323 call signalling

H.323 RECOMMENDATION SERIES DEFECT REPORT FORM

DATE:	
CONTACT INFORMATION **NAME:** **COMPANY:** **ADDRESS:** **TEL:** **FAX:** **EMAIL:**	
AFFECTED RECOMMENDATIONS:	
DESCRIPTION OF PROBLEM:	
SUGGESTIONS FOR RESOLUTION:	

NOTE - Attach additional pages if more space is required than is provided above.