



INTERNATIONAL TELECOMMUNICATION UNION

**ITU-T**

**H.323 System  
Implementors'  
Guide**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

(5 August 2005)

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

---

**Implementors' Guide for Recommendations of  
the H.323 System ("Packet-based multimedia  
communications systems"):**

***H.323, H.225.0, H.245, H.246, H.283, H.235, H.341,  
H.450 Series, H.460 Series, and H.500 Series***

---

## **Summary**

This document is a compilation of reported defects identified in the versions of ITU-T Recommendation H.323 and its related Recommendations currently in force. It must 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.

This revision contains all updates submitted upto and including those at Study Group 16 meeting, July/August 2005, in Geneva (TD 175/PLEN). This Implementors' Guide provides corrections and clarifications for implementations of the H.323-series Recommendations approved in May 2003, and supersedes the earlier Implementors' Guide version approved 2004-11.

## Contact Information

ITU-T Study Group 16 / Rapporteur Question 2/16	Paul E. Jones Cisco Systems, Inc. 7025 Kit Creek Road Research Triangle Park, NC 27709. USA	Tel: +1 919 392 6948 Fax: +1 919 392 2177 E-mail: <a href="mailto:paulej@packetizer.com">paulej@packetizer.com</a>
ITU-T Study Group 16 / Rapporteur Question 3/16	Christian Groves Ericsson Australia Pty. Ltd. 37/360 Elizabeth Street Victoria 3000. Australia	Tel: +61 3 9301 6116 Fax: +61 3 9301 1499 E-mail: <a href="mailto:Christian.Groves@ericsson.com">Christian.Groves@ericsson.com</a>
ITU-T Study Group 16 / Rapporteur Question 25/16	Martin Euchner Siemens AG Hofmannstr 51 81359 Munich, Germany	Tel: +49 89 722 5 57 90 Fax: +49 89 722 6 23 66 E-mail: <a href="mailto:martin.euchner@siemens.com">martin.euchner@siemens.com</a>
Editor ITU-T Rec. H.235	Roni Even Polycom Israel Ltd. 94 Derech Em Hamoshavot P.O. Box 3654 Petach-Tikva 49130 Israel	Tel: +972 3 925 1200 Fax: +972 3 921 1571 E-mail: <a href="mailto:roni.even@polycom.co.il">roni.even@polycom.co.il</a>
Editor ITU-T Rec. H.341	Craig Blasberg Cisco Systems, Inc. 7025 Kit Creek Road Research Triangle Park, NC 27709. USA	Tel: +1 919 392 5760x Fax: +1 919 392 6801 E-mail: <a href="mailto:blasberg@cisco.com">blasberg@cisco.com</a>
Editor ITU-T Rec. H.225.0	Faisal Siyavudeen Cisco Systems, Inc. Divyashree Chambers 'B' Wing, No.11 O'Shaugnessey Road, Off Langford Road Bangalore, India – 560027	Tel: +91-80-51033543 Fax: +91-80-22230167 E-mail: <a href="mailto:fsiyavud@cisco.com">fsiyavud@cisco.com</a>
Editor ITU-T Rec. H.225.0 Annex G	Miner Gleason Cisco Systems, Inc. 7025 Kit Creek Road Research Triangle Park, NC 27709. USA	Tel: +1 919 392 8752 Fax: +1 919 392 7065 E-mail: <a href="mailto:mgleason@cisco.com">mgleason@cisco.com</a>
Editor ITU-T Rec. H.245	Mike Nilsson BT Labs Ipswich. United Kingdom	Tel: +44 1 473 645413 Fax: +44 1 473 643791 E-mail: <a href="mailto:mike.nilsson@bt.com">mike.nilsson@bt.com</a>
Editor ITU-T Rec. H.246	Ogishi Hisataka	E-mail: <a href="mailto:ogishi.hisataka@lab.ntt.co.jp">ogishi.hisataka@lab.ntt.co.jp</a>
ITU-T Study Group 16 / Rapporteur Question 5/16 & Editor ITU-T Rec. for H.450. {1,2,3,4,5, 6,9,10,11,12}	Ernst Horvath Siemens Austria Gudrunstrasse 11 A-1101 Vienna. Austria	Tel: +43 5 1707 45897 Fax: +43 5 1707 56992 E-mail: <a href="mailto:ernst.horvath@siemens.at">ernst.horvath@siemens.at</a>
Editor ITU-T Rec. H.450.7	Dave Walker SS8 Networks 135 Michael Cowpland Drive, Suite 200 Kanata, Ontario, K2M 2E9. Canada	Tel: +1 613 592 8450 Fax: +1 613 592 9634 E-mail: <a href="mailto:dwalker@ss8networks.com">dwalker@ss8networks.com</a>
Editor ITU-T Rec. H.450.8	Glen Freundlich Avaya Communication 1300 W. 120th Avenue Westminster, CO 80234. USA	Tel: +1 303 538 2899 Fax: +1 303 538 3007 E-mail: <a href="mailto:ggf@avaya.com">ggf@avaya.com</a>
Editor ITU-T Rec. H.460.1	P. Cordell	E-mail: <a href="mailto:pete@tech-know-ware.com">pete@tech-know-ware.com</a>
Editor ITU-T Rec. H.460.4	Gary Thom	E-mail: <a href="mailto:gthom@delta-info.com">gthom@delta-info.com</a>
Editor ITU-T Rec. H.460.5	Sasha Ruditsky	E-mail: <a href="mailto:sasha@radvision.com">sasha@radvision.com</a>
Editor ITU-T Rec. H.460.6	Bob Gilman	E-mail: <a href="mailto:rrg@avaya.com">rrg@avaya.com</a>
Editor ITU-T Rec. H.460. {2,7,8}	Paul Jones (see above for Q.2/16)	E-mail: <a href="mailto:paulej@packetizer.com">paulej@packetizer.com</a>
Editor ITU-T Rec. H.460.3	Louis Fourie	E-mail: <a href="mailto:lfourie@cisco.com">lfourie@cisco.com</a>
Editor ITU-T Rec. H.460.9	Ernst Horvath (see above for Q.5/16)	E-mail: <a href="mailto:ernst.horvath@siemens.at">ernst.horvath@siemens.at</a>

**Note:** Not all Recommendations indicated above have IG issues in this document. The information above is provided for completeness.

## Table of Contents

<b>1</b>	<b>INTRODUCTION</b> .....	<b>1</b>
<b>2</b>	<b>SCOPE</b> .....	<b>1</b>
<b>3</b>	<b>REFERENCES</b> .....	<b>1</b>
<b>4</b>	<b>NOMENCLATURE</b> .....	<b>3</b>
<b>5</b>	<b>TECHNICAL AND EDITORIAL CORRECTIONS TO H.323 SERIES RECOMMENDATIONS</b> .....	<b>3</b>
5.1	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.323 (2003).....	3
5.2	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.225.0 (2003).....	5
5.3	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.245 (1/2005) .....	9
5.4	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.246 (1998).....	9
5.5	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.235 (2003).....	20
5.6	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.450 SERIES .....	20
5.7	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.341 (1999).....	35
5.8	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.283 (1999).....	44
5.9	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION H.460 SERIES .....	44
5.10	TECHNICAL AND EDITORIAL CORRECTIONS TO ITU-T RECOMMENDATION ANNEX P/H.323 (2003).....	53
<b>6</b>	<b>IMPLEMENTATION CLARIFICATIONS</b> .....	<b>54</b>
6.1	TOKEN USAGE IN H.323 SYSTEMS .....	54
6.2	H.235 RANDOM VALUE USAGE IN H.323 SYSTEMS .....	54
6.3	GATEWAY RESOURCE AVAILABILITY MESSAGES .....	54
6.4	OPENLOGICALCHANNEL IN FASTSTART .....	55
6.5	CLARIFICATION IN Q.931 (1993) .....	55
6.6	GRACEFUL CLOSURE OF TCP CONNECTIONS.....	55
6.7	RACE CONDITION ON SIMULTANEOUS CLOSE OF CHANNELS .....	55
6.8	ACCEPTANCE OF FAST CONNECT .....	55
6.9	SEMANTIC DIFFERENCES BETWEEN LIGHTWEIGHT RRQs AND IRQ/IRR MESSAGES .....	55
6.10	SPECIFYING THE PAYLOAD FORMAT FOR A CHANNEL .....	56
6.11	VERSION DEPENDENCIES IN ANNEXES.....	56
6.12	ROUTING THROUGH SIGNALING ENTITIES AND DETECTING LOOPS .....	56
6.13	PACKETIZATION FOR G.729, G.729A, G.711, AND G.723.1 .....	58
6.14	CHECKING VERSIONS FOR T.38 AND V.150.1.....	58
<b>7</b>	<b>ALLOCATED OBJECT IDENTIFIERS AND PORT NUMBERS</b> .....	<b>58</b>
7.1	ALLOCATED OBJECT IDENTIFIERS .....	58
7.2	ALLOCATED PORT NUMBERS.....	59
<b>8</b>	<b>USE OF E.164 AND ISO/IEC 11571 NUMBERING PLANS</b> .....	<b>59</b>
8.1	E.164 NUMBERING PLAN.....	59
8.2	PRIVATE NETWORK NUMBER .....	62
<b>9</b>	<b>ASN.1 USAGE, GUIDELINES, AND CONVENTIONS</b> .....	<b>63</b>
9.1	NULL, BOOLEAN, AND NULL/BOOLEAN OPTIONAL .....	63
9.2	ASN.1 USAGE IN H.450-SERIES RECOMMENDATIONS.....	63
	<b>ANNEX: H.323 RECOMMENDATION SYSTEM DEFECT REPORT FORM</b> .....	<b>69</b>

# **IMPLEMENTORS' GUIDE FOR ITU-T H.323, H.225.0, H.245, H.246, H.283, H.235, H.341, H.450 SERIES, H.460 SERIES, AND H.500 SERIES RECOMMENDATIONS**

## **1 Introduction**

This document is a compilation of reported defects identified in the versions of ITU-T Recommendation H.323 and its related Recommendations currently in force. It must 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.323-series Recommendations.

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 Q2/16 or Q3/16 Rapporteur. The template for a defect report is located at the end of the Guide. Contact information for these parties is included at the front of the 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 any interested party. Formal membership in the ITU is not required to participate in this process.

## **2 Scope**

This guide resolves defects in the following categories:

- editorial errors
- technical errors, such as omissions and inconsistencies
- ambiguities

In addition, the Implementors' 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 through contributions to the ITU-T.

## **3 References**

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation

- ITU-T Recommendation H.323 (2003), Packet-Based multimedia communications systems
- ITU-T Recommendation H.225.0 (2003), Call signaling protocols and media stream packetization for packet based multimedia communications Systems
- ITU-T Recommendation H.245 (7/2003), Control protocol for multimedia communication
- ITU-T Recommendation H.246 (1998), Interworking of H-Series multimedia terminals with H-Series multimedia terminals and voice/voiceband terminals on GSTN and ISDN

- ITU-T Recommendation H.246 – Annex C (2003), ISDN User Part Function - H.225.0 Interworking
- ITU-T Recommendation H.235 (2003), Security and encryption for H Series (H.323 and other H.245 based) multimedia terminals
- ITU-T Recommendation H.235 Amendment 1 (2004)
- 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
- ITU-T Recommendation H.450.4 (1999), Call hold supplementary service for H.323
- ITU-T Recommendation H.450.5 (1999), Call park and call pickup supplementary services for H.323
- ITU-T Recommendation H.450.6 (1999), Call waiting supplementary service for H.323
- ITU-T Recommendation H.450.7 (1999), Message waiting indication supplementary service for H.323
- ITU-T Recommendation H.450.8 (2000), Name identification supplementary service for H.323
- ITU-T Recommendation H.450.9 (2000), Call Completion Supplementary Services for H.323
- ITU-T Recommendation H.450.10 (2001), Call offer supplementary service for H.323
- ITU-T Recommendation H.450.11 (2001), Call intrusion supplementary services
- ITU-T Recommendation H.450.12 (2001), Call Information Additional Network Feature for H.323
- ITU-T Recommendation H.460.1 (2002), Guidelines for the use of generic extensibility framework
- ITU-T Recommendation H.460.2 (2001), Number Portability interworking between H.323 and SCN networks
- ITU-T Recommendation H.460.3 (2002), Circuit status map within H.323 systems
- ITU-T Recommendation H.460.4 (2002), Call priority designation for H.323 calls
- ITU-T Recommendation H.460.5 (2002), H.225.0 transport of multiple Q.931 IE of the same type
- ITU-T Recommendation H.460.6 (2002), Extended Fast Connect Feature
- ITU-T Recommendation H.460.7 (2002), Digit Maps Within H.323 Systems
- ITU-T Recommendation H.460.8 (2002), Querying for alternate routes within H.323 systems
- ITU-T Recommendation H.460.9 (2002), Support for online QoS-Monitoring report
- ISO/IEC 11571 (1998), Information technology – Telecommunications and information exchange between systems – Private Integrated Services Networks – Addressing
- ITU-T Recommendation Q.931 (1998), ISDN user-network interface layer 3 specification for basic call control
- ITU-T Recommendation H.283, Remote device control logical channel transport

## 4 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
<u>[Begin Correction]</u>	Identifies the start of revision marked text based on extractions from the published Recommendations affected by the correction being described.
<u>[End Correction]</u>	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.

## 5 Technical and Editorial Corrections to H.323 Series Recommendations

### 5.1 Technical and Editorial Corrections to ITU-T Recommendation H.323 (2003)

#### 5.1.1 H.245 Request Mode Initiation

<b>Description:</b>	Figures D.8 through D.11/H.323 show the H.245 procedures necessary to establish a fax call. The endpoints shown are labeled as Originating and Terminating. However, this distinction is not important and any endpoint that detects the fax tone may initiate the procedures, regardless of whether it originated or terminated the call. The following text should be added to Annex D/H.323 to further clarify this.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

[Begin Correction]

#### D.5 Replacing an existing audio stream with a T.38 fax stream

...

Figure D.8 illustrates a successful switchover from voice to fax when a separate H.245 channel is already open for two unidirectional media channels. Note that in this diagram and in ones that follow, originating and terminating endpoints do not necessarily refer to sending and receiving or calling and called endpoints. Any endpoint that detects the fax tone initiates H.245 procedures to switch to fax mode.

[End Correction]

## 5.1.2 Clarifications to H.323 Call Clearing

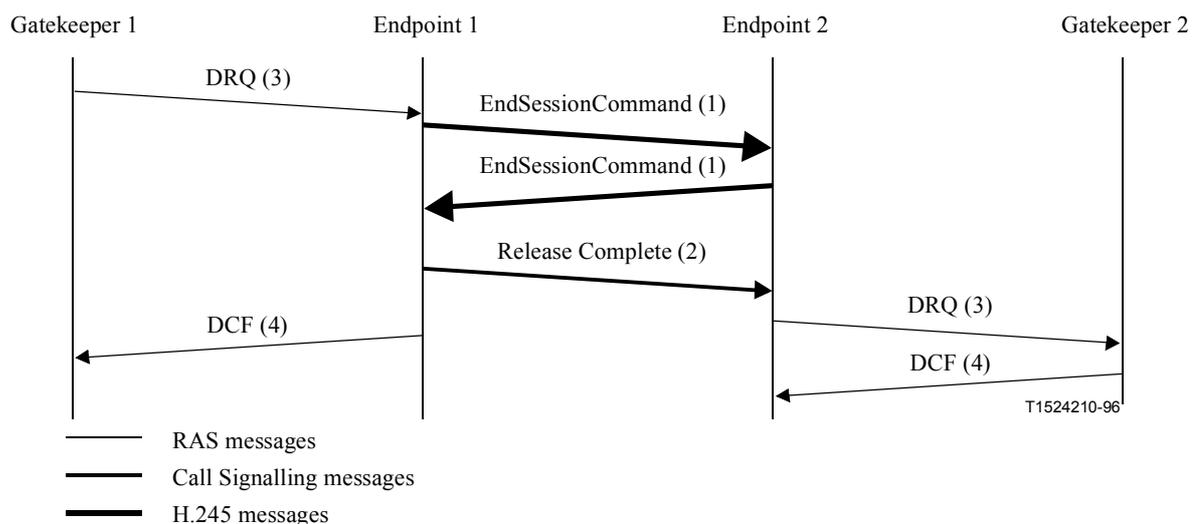
<b>Description:</b>	The procedures in H.323 Section 8.5 that describe call clearing have been a source of recent confusion regarding the messages sent immediately before H.225.0 ReleaseComplete. The text below clarifies the call clearing procedure.
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

*[Begin Correction]*

### 8.5.3 Call clearing by gatekeeper

The Gatekeeper may terminate call by sending a DRQ to an endpoint. See Figure 52. The endpoint shall immediately follow steps A-1) through A-5) or B-1) through B-6) from above and then reply to the Gatekeeper with DCF. The other endpoint, upon receiving **endSessionCommand**, shall follow the procedure described above. Figure 52 shows the direct call model; a similar procedure is followed for the Gatekeeper routed model.

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



**Figure 52/H.323 – Gatekeeper initiated call clearing**

Note – The Release Complete message mentioned in this section may actually be preceded by other H.225.0 messages as part of the call termination process. For example, when tunneling another protocol inside H.225.0 it may be necessary to exchange tunneled messages prior to sending the final Release Complete. Also, since H.245 and H.225.0 may operate on separate connections, an H.245 End Session may arrive after a previously transmitted H.225.0 message. As such, implementations should not expect that the H.225.0 message immediately following receipt of an H.245 End Session will necessarily be Release Complete.

*[End Correction]*

## 5.2 Technical and Editorial Corrections to ITU-T Recommendation H.225.0 (2003)

### 5.2.1 Error Codes Mapping is Unspecified

<b>Description:</b>	Mapping from LocationRejectReason and AccessRejectionReason codes to AdmissionRejectReason codes is not specified within H.225.0. The following new section provides this mapping.
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

*[Begin Correction]*

### 7.22 Error Code Mapping

A Gatekeeper that needs to return an **AdmissionReject** message in response to an **AdmissionRequest** from an endpoint, as a result of having received a **LocationReject** or an **H.501 AccessRejection** in response to its sending an **LocationRequest** or **AccessRequest** message, should use the following tables to map the error code that it returns in the **AdmissionReject** message.

**Table 24/H.225.0 – LocationRejectReason to AdmissionRejectReason**

<u>LocationRejectReason</u>	<u>Corresponding AdmissionRejectReason</u>
<u>notRegistered</u>	<u>calledPartyNotRegistered</u>
<u>invalidPermission</u>	<u>invalidPermission</u>
<u>requestDenied</u>	<u>requestDenied</u>
<u>undefinedReason</u>	<u>undefinedReason</u>
<u>securityDenial</u>	<u>securityDenial</u>
<u>aliasInconsistent</u>	<u>aliasesInconsistent</u>
<u>routeCallToSCN</u>	<u>routeCallToSCN</u>
<u>resourceUnavailable</u>	<u>resourceUnavailable</u>
<u>genericDataReason</u>	<u>genericDataReason</u>
<u>neededFeatureNotSupported</u>	<u>neededFeatureNotSupported</u>
<u>hopCountExceeded</u>	<u>noRouteToDestination</u>
<u>incompleteAddress</u>	<u>incompleteAddress</u>
<u>securityWrongSyncTime</u>	<u>securityWrongSyncTime</u>
<u>securityReplay</u>	<u>securityReplay</u>
<u>securityWrongGeneralID</u>	<u>securityWrongGeneralID</u>
<u>securityWrongSendersID</u>	<u>securityWrongSendersID</u>
<u>securityMessageIntegrityFailed</u>	<u>securityMessageIntegrityFailed</u>
<u>securityWrongOID</u>	<u>securityWrongOID</u>
<u>securityDHmismatch</u>	<u>securityDHmismatch</u>
<u>noRouteToDestination</u>	<u>noRouteToDestination</u>
<u>unallocatedNumber</u>	<u>unallocatedNumber</u>

**Table 25/H.225.0 – AccessRejectionReason to AdmissionRejectReason**

<u>AccessRejectionReason</u>	<u>Corresponding AdmissionRejectReason</u>
<u>noMatch</u>	<u>noRouteToDestination</u>
<u>packetSizeExceeded</u>	<u>undefinedReason</u>
<u>security</u>	<u>securityDenial</u>
<u>hopCountExceeded</u>	<u>noRouteToDestination</u>
<u>needCallInformation</u>	<u>undefinedReason</u>
<u>noServiceRelationship</u>	<u>noRouteToDestination</u>
<u>undefined</u>	<u>undefinedReason</u>
<u>neededFeature</u>	<u>neededFeatureNotSupported</u>
<u>genericDataReason</u>	<u>genericDataReason</u>
<u>destinationUnavailable</u>	<u>resourceUnavailable</u>
<u>aliasesInconsistent</u>	<u>aliasesInconsistent</u>
<u>resourceUnavailable</u>	<u>resourceUnavailable</u>
<u>incompleteAddress</u>	<u>incompleteAddress</u>
<u>unknownServiceID</u>	<u>noRouteToDestination</u>
<u>usageUnavailable</u>	<u>undefinedReason</u>
<u>cannotSupportUsageSpec</u>	<u>undefinedReason</u>
<u>unknownUsageSendTo</u>	<u>undefinedReason</u>

---

*[End Correction]*

## 5.2.2 Clarification on Usage of additionalSourceAddresses

<b>Description:</b>	The interworking function between ISUP and H.323 uses Calling Party IE and <b>additionSourceAddresses</b> in H.225.0 Call Signalling Setup message to carry more than one calling party address. The usage by a call signaling routed GK is noted as below.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

### 7.2.2.6 Calling party number

...

H.323 endpoints shall not send multiple Calling Party Number IEs in the same message. Gateways may provide support for interworking with Q.931 SETUP messages that contain multiple Calling Party Number IEs. Gateways that provide such support shall map the first Q.931 Calling Party Number IE to the Calling Party Number IE of the H.225.0 Setup message, and map subsequent Q.931 Calling Party Number IEs to the **additionalSourceAddresses** field of the H.225.0 Setup message. Gatekeepers that route H.225.0 Setup messages initiated by an H.323 endpoint may insert a number in the **additionalSourceAddresses** field before forwarding it to its next recipient.

---

*[End Correction]*

### 5.2.3 Gatekeeper Assignment of Aliases

<b>Description:</b>	The text in H.225.0v4 removed the capability of a Gatekeeper to assign an E.164 address to an endpoint that does not register any itself. The following corrections restore this capability.
---------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

---

#### 7.9.1 RegistrationRequest (RRQ)

...

**terminalAlias** – This optional value is a list of alias addresses, by which other terminals may identify this terminal. This field may be used in addition to or as an alternative to the **terminalAliasPattern** and **supportedPrefixes** fields. If the **terminalAlias** is null, a **terminalAlias** address may be assigned by the gatekeeper, and included in the RCF. If the **terminalAlias** does not contain any **dialedDigits** or **partyNumber** address, a **dialedDigits** or **partyNumber** address may be assigned by the gatekeeper and included in the RCF. If an **email-ID** is available for the endpoint, it should be registered. Note that multiple alias addresses may refer to the same transport addresses. All of the endpoint's aliases that it desires to register shall be included in this list unless the **additiveRegistration** option is specified, in which case the endpoint aliases in an RRQ shall be added to the list of aliases currently registered for the endpoint.

---

*[End Correction]*

---

### 5.2.4 Use of Facility to initiate H.245

<b>Description:</b>	The usage of Facility message to carry h245Address to initiate H.245 procedures is not correctly reflected in the text. The following sections in H.225.0 should be amended as shown below
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

---

#### 7.3.1 Alerting

...

**h245Address** – This is a specific transport address on which the called endpoint or gatekeeper handling the call would like to establish H.245 signalling. This address may also be sent in Call Proceeding, Progress, ~~or~~ Connect, or Facility.

...

#### 7.3.3 Connect

...

**h245Address** – This is a specific transport address on which the called endpoint or gatekeeper handling the call would like to establish H.245 signalling. This address shall be sent if sent earlier in Alerting, Progress, ~~or~~ Call Proceeding, or Facility.

...

#### 7.3.7 Progress

...

**h245Address** – This is a specific transport address on which the called endpoint or gatekeeper handling the call would like to establish H.245 signalling. This address shall be sent if sent earlier in Call Proceeding, Alerting, ~~or~~ Connect, or Facility.

...

---

*[End Correction]*

### 5.2.5 Length Field of UIIE

<b>Description:</b>	The current text describing the length of the UIIE field and the value supplied in Table 7/H.225.0 through Table 18/H.225.0 can be misleading. The text below supplies the correction.
---------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

### 7.3 Q.931 based H.225.0 call signalling message details

Note that the lengths of the information elements specified in the tables below refer to messages that are generated by H.323 terminals only. The size of the User-user information element, not explicitly shown, is understood as the size of the ~~user-data structure in PER encoded H323-UserInformation sequence and does not include the h323-UU-PDU~~. The total size of **H323-UserInformation** is limited to 65 536 octets. Regardless of the specified sizes, messages forwarded from the SCN side may have different (larger) sizes.

---

*[End Correction]*

---

*[Begin Correction]*

[Editorial Note] The correction below to the User-User Information Element should be applied to all tables from Table 7/H.225.0 through to Table 18/H.225.0.

**Table 7/H.225.0 – Alerting**

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	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
Signal	O	2-3

High layer compatibility	FFS	NA
User-user	M	2-131*

---

*[End Correction]*

**5.3 Technical and Editorial Corrections to ITU-T Recommendation H.245 (1/2005)**

**5.4 Technical and Editorial Corrections to ITU-T Recommendation H.246 (1998)**

**5.4.1 Technical and Editorial Corrections to Annex C/H.246 (2003)**

**5.4.1.1 Interworking for Conveying Two Calling Party Numbers**

<b>Description:</b>	<p>ISUP networks can convey 2 calling party numbers: One in the calling party number, and another one in the generic number (which has the qualification “additional calling party number”). H.323 describes H.460.5 to convey multiple IEs. However this does not allow Gatekeepers to add a “network provided number” and is un-necessarily complex for this simple function.</p> <p>The interworking function to use the existing H.225.0 field <b>additionalSourceAddresses</b> to support this functionality is described using the following set of corrections and additions.</p>
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

**C.6.2.1.1 Special arrangement applies**

...

**Setup Received from Gatekeeper**

A) If only the **additionalSourceAddresses** is present, or

If only the Calling Party Number is present, or

If only the **sourceAddress** is present, then Table C.20.1 applies:

**Table C.20.1/H.246 – Calling Party Number**

<u>SETUP→</u>	<u>IAM→</u>
<u><b>additionalSourceAddresses</b> or</u> <u><b>Calling Party Number</b> or</u> <u><b>sourceAddress</b></u>	<u><b>Calling Party Number</b></u>

B) If the Calling Party Number (or the **sourceAddress** in case the Calling Party Number IE is absent) and the **additionalSourceAddresses** are present, then Table C.20.2 applies:

**Table C.20.2/H.246 – Calling Party Number**

<u>SETUP</u> →	<u>IAM</u> →
<u>Calling Party Number</u> or <u>sourceAddress</u>	<u>Generic Number</u> (– <u>additional Calling Party Number</u> )
<u>additionalSourceAddresses</u>	<u>Calling Party Number</u>

Table C.20 applies:

**Table C.20/H.246—Calling Party Number**

<u>SETUP</u> →	<u>IAM</u> →
<u>Calling Party Number</u>	<u>Calling Party Number</u> or <u>Generic Number</u> —(– <u>additional Calling Party number</u> )
<u>SourceAddress</u>	<u>Calling Party Number</u>
NOTE—If a Calling Party number is included in the <b>sourceAddress</b> , then the Calling party number should be sent in the Generic Number.	

The address presentation restricted indicator of the calling party number and generic number parameters shall be set according to the CLIR supplementary service. ~~The H.225.0 Setup IE **presentationIndicator**~~ indicates whether presentation of the **sourceAddress** should be allowed or restricted. If both ~~**presentationIndicator**~~ and the presentation indicator of the Calling Party Number IE are present and are in conflict, the presentation indicator of the Calling Party Number IE shall be used.

---

*[End Correction]*

---

*[Begin Correction]*

---

#### C.6.2.1.2 Special arrangement does not apply

...

Table C.20.1 and Table C.20.2 apply.

**Table C.22/H.246—Connected Party Number**

<u>SETUP</u> →	<u>IAM</u> →
----------------	--------------

<b>Calling Party Number</b>	<b>Calling Party Number</b> or (Note) <b>Generic Number</b> — ( <del>additional Calling Party number</del> )
<b>sourceAddress</b>	<b>Calling Party Number</b>
NOTE — If a Calling Party number is included in the sourceAddress, then the Calling party number should be sent in the Generic Number.	

The address presentation restricted indicator of the calling party number parameter shall be set according to the CLIR supplementary service. ~~The H.225.0 Setup IE **presentationIndicator**— indicates whether presentation of the sourceAddress should be allowed or restricted. If both **presentationIndicator** and the presentation indicator of the Calling Party Number IE are present and are in conflict, the presentation indicator of the Calling Party Number IE shall be used.~~

---

*[End Correction]*

---

*[Begin Correction]*

---

### C.6.2.2 Calling Party Name Restriction (H.450.8)/Calling Line Identification Restriction (CLIR)

If **additionalSourceAddresses** field is not present then ~~The~~ address presentation restricted indicator of the calling party number and of the generic number parameters is coded as described in Table C.23.

NOTE – If the calling user does not have the Calling Party Name Restriction, the address presentation restricted indicator of the calling party number parameter is set to *presentation allowed* (see 4.10/Q.951.x [19]).

**Table C.23/H.246 – Coding of the address presentation restricted indicator of the calling party number and generic number parameters**

Internal data (User profile data)		SETUP→	IAM→
Permanent mode	Temporary mode default setting	Calling party number information element/ User-user information element	Calling party number/ generic number parameter
		Presentation indicator	Address presentation restricted indicator
Yes	Value non-significant	Value non-significant	Presentation restricted
No	Restricted	Presentation restricted	Presentation restricted
		Absent	Presentation restricted
		Presentation allowed	Presentation allowed
	Allowed	Presentation allowed	Presentation allowed
		Absent	Presentation allowed
		Presentation restricted	Presentation restricted

NOTE – The presentationIndicator field in Setup-UUIE carries information identical to the presentation indicator found in the Calling Party Number IE. If both **presentationIndicator** and the presentation indicator of the Calling Party Number IE are present and are in conflict, the presentation indicator of the Calling Party Number IE shall be used. The meaning and use of the presentation indicator is defined in Q.951.x.

If **additionalSourceAddresses** field is present then the address presentation restricted indicator of the calling party number and of the generic number parameters is coded as described in Table C.23bis.

**Table C.23bis/H.246 – Coding of the address presentation restricted indicator of the calling party number and generic number parameters**

<u>Internal data (User profile data)</u>		<u>SETUP→</u>	<u>IAM→</u>
<u>Permanent mode</u>	<u>Temporary mode default setting</u>		
<u>Yes</u>	<u>Value non-significant</u>	<u>Calling party number information element/ User-user information element presentation indicator</u> <u>Value non-significant</u>	<u>generic number presentation indicator</u>  <u>Presentation restricted</u>
		<b><u>additionalSourceAddresses presentation indicator</u></b> <u>Value non-significant</u>	<u>Calling party number presentation indicator</u>  <u>Presentation restricted</u>
<u>No</u>	<u>Restricted</u>	<u>Calling party number information element/ User-user information element presentation indicator</u>	<u>generic number presentation indicator</u>
		<u>Presentation restricted</u>	<u>Presentation restricted</u>
		<u>Absent</u>	<u>Presentation restricted</u>
		<u>Presentation allowed</u>	<u>Presentation allowed</u>
		<b><u>additionalSourceAddresses presentation indicator</u></b>	<u>Calling party number presentation indicator</u>
		<u>Presentation restricted</u>	<u>Presentation restricted</u>
		<u>Absent</u>	<u>Presentation restricted</u>
<u>Presentation allowed</u>	<u>Presentation allowed</u>		

<u>Internal data (User profile data)</u>		<u>SETUP→</u>	<u>IAM→</u>
<u>Permanent mode</u>	<u>Temporary mode default setting</u>		
	<u>Allowed</u>	<u>Calling party number information element/ User-user information element presentation indicator</u>	<u>generic number presentation indicator</u>
		<u>Presentation allowed</u>	<u>Presentation allowed</u>
		<u>Absent</u>	<u>Presentation allowed</u>
		<u>Presentation restricted</u>	<u>Presentation restricted</u>
		<u>additionalSourceAddresses presentation indicator</u>	<u>Calling party number presentation indicator</u>
		<u>Presentation allowed</u>	<u>Presentation allowed</u>
		<u>Absent</u>	<u>Presentation allowed</u>
		<u>Presentation restricted</u>	<u>Presentation restricted</u>
NOTE – The presentationIndicator field in Setup-UUIE carries information identical to the presentation indicator found in the Calling Party Number IE. If both <b>presentationIndicator</b> and the presentation indicator of the Calling Party Number IE are present and are in conflict, the presentation indicator of the Calling Party Number IE shall be used for the ISUP generic number with the qualification “additional calling party number”. The meaning and use of the presentation indicator is defined in Q.951.x.			

---

*[End Correction]*

---

*[Begin Correction]*

---

### **C.7.2.3 Calling Line Identification Presentation (CLIP)/Calling Party Name Presentation (H.450.8)**

If the called user has CLIP, one or two calling party number information elements are sent in the SETUP message by performing the conversion specified in H.460.5[20]tables C.56 and C.60.

#### **SETUP message sent to Terminal or Gateway**

See Table C.56.

**Table C.56/H.246 – CLIP information sent to the called user**

IAM→		SETUP→
Calling party number parameter Address presentation restricted indicator	Generic number parameter with number qualifier set to <i>additional calling party number</i>	CPAP information sent to the calling user
<i>Presentation allowed</i>	Absent	sourceAddress or Calling party number IE (see Table C.57)
<i>Presentation allowed</i>	Present	sourceAddress or Calling party number IE (Notes 1-and-2) (see Table C.578) <b><u>additionalSourceAddresses</u></b> Calling party number IE (Notes 1-and-2) (see Table C.58bis)
<i>Presentation restricted</i> (Note 32)	Value non-significant	sourceAddress or Calling party number IE <u>And</u> <b><u>additionalSourceAddresses</u></b> Option 1: Type of number As received Numbering plan As received Presentation ind. <i>Presentation restricted</i> Screening ind. As received Number digits No digit  Option 2: Type of number <i>Unknown</i> Numbering plan <i>Unknown</i> Presentation ind. <i>Presentation restricted</i> Screening ind. <i>Network provided</i> Number digits No digit

**Table C.56/H.246 – CLIP information sent to the called user (concluded)**

IAM→		SETUP→
<p><i>Address not available</i> or No calling party number parameter</p>	<p>Value non-significant</p>	<p>sourceAddress or Calling party number IE</p>
		<p>Type of number    <i>Unknown</i> Numbering plan    <i>Unknown</i> Presentation ind.    <i>Not available due to interworking</i> Screening ind.    <i>Network provided</i> Number digits    No digit</p>

NOTE 1 – If the "two calling party number delivery option" does not apply:  
 – only one calling party number information element is sent on H.225.0 side. The generic number is used (see Table C.58).

If the "two calling party number delivery option" applies:

- ~~By performing the conversion specified in H.460.5[20], two calling party number information elements are sent on H.225.0 side: one coded according to the generic number parameter (see Table C.58), one according to the calling party number parameter (see Table C.57). The order in which the calling party number information elements appear in the SETUP message is a network option.~~
- The additionalSourceAddresses is sent and is coded according to the calling party number parameter received, and the Calling Party Number IE is coded according to the generic number parameter (see Table C.58)

NOTE 2 – As a national option, the presentation restriction indication received in the calling party number parameter can be overridden for specific calling access categories. In such a case, the same actions are taken as if *presentation allowed* was received, except for the presentation restriction indication, which is passed transparently into the calling party number information element.

**Table C.57/H.246 – Coding of the calling party number information element according to the calling party number parameter**

IAM→	SETUP→
<p>Calling party number parameter</p>	<p>sourceAddress or Calling party number IE</p>
<p>Nature of address indicator <i>National number</i> <i>International number</i></p>	<p>Type of number (Note 1) <i>National number</i> <i>International number</i></p>
<p>Numbering plan indicator <i>ISDN/Telephony numbering plan</i></p>	<p>Numbering plan identification <i>ISDN/Telephony numbering plan</i></p>
<p>Address presentation restricted indicator <i>Presentation allowed</i> <i>Presentation restricted</i></p>	<p>Presentation indicator (Note 2) <i>Presentation allowed</i> <i>Presentation restricted</i></p>
<p>Screening indicator <i>User provided, verified and passed</i> <i>Network provided</i></p>	<p>Screening indicator (Note 3) <i>User provided, verified and passed</i> <i>Network provided</i></p>
<p>Address signals</p>	<p>Number digits</p>

NOTE 1 – As a network option, the type of number may be coded *unknown* when a prefix is added to the number.

NOTE 2 – The Presentation Indicator may be coded as part of the Calling Party Number or as a H.225.0 presentationIndicator IE.

NOTE 3 – The Screening Indicator may be coded as part of the Screening Indicator or as a H.225.0 screeningIndicator IE.

**Table C.58/H.246 – Coding of the calling party number information element according to the generic number parameter**

<b>IAM→</b>	<b>SETUP→</b>
Generic number parameter with number qualifier set to <i>additional calling party number</i>	sourceAddress or Calling party number IE
Nature of address indicator <i>National number</i> <i>International number</i>	Type of number (Note 1) <i>National number</i> <i>International number</i>
Numbering plan indicator <i>ISDN/Telephony numbering plan</i>	Numbering plan identification <i>ISDN/Telephony numbering plan</i>
Address presentation restricted indicator <i>Presentation allowed</i> <i>Presentation restricted</i>	Presentation indicator (Note 2) <i>Presentation allowed</i> <i>Presentation restricted</i>
Screening indicator <i>User provided, not verified</i>	Screening indicator (Note 3) <i>User provided, not verified</i>
Address signals	Number digits
NOTE 1 – As a network option, the type of number may be coded <i>unknown</i> when a prefix is added to the number.	
NOTE 2 – The Presentation Indicator may be coded as part of the Calling Party Number or as a H.225.0 presentationIndicator IE.	
NOTE 3 – The Screening Indicator may be coded as part of the Screening Indicator or as a H.225.0 screeningIndicator IE.	

**Table C.58bis /H.246 – Coding of the additionalSourceAddresses information element according to the calling party number parameter**

<b><u>IAM→</u></b>	<b><u>SETUP→</u></b>
<u>Calling party number parameter</u>	<u>additionalSourceAddresses</u>
<u>Nature of address indicator</u> <u><i>National number</i></u> <u><i>International number</i></u>	<u>Type of number (Note 1)</u> <u><i>National number</i></u> <u><i>International number</i></u>
<u>Numbering plan indicator</u> <u><i>ISDN/Telephony numbering plan</i></u>	<u>Numbering plan identification</u> <u><i>ISDN/Telephony numbering plan</i></u>
<u>Address presentation restricted indicator</u> <u><i>Presentation allowed</i></u> <u><i>Presentation restricted</i></u>	<u>Presentation indicator</u> <u><i>Presentation allowed</i></u> <u><i>Presentation restricted</i></u>

Screening indicator <i>User provided, verified and passed</i> <i>Network provided</i>	Screening indicator <i>User provided, verified and passed</i> <i>Network provided</i>
Address signals	Number digits
NOTE 1 – As a network option, the type of number may be coded <i>unknown</i> when a prefix is added to the number.	

[End Correction]

[Begin Correction]

### SETUP Message sent to Gatekeeper

A) If the calling party number is absent in ISUP then Table C.60.1 applies:

**Table C.60.1/H.246 – Calling Party Number**

<u>←SETUP</u>	<u>←IAM</u>
sourceAddress or Calling Party Number IE	
Type of number <i>Unknown</i>	
Numbering plan <i>Unknown</i>	
Presentation ind. <i>Not available due to interworking</i>	
Screening ind. <i>Network provided</i>	
Number digits <i>No digit</i>	

B) If only the calling party number is present in ISUP, then Table C.60.2 applies:

**Table C.60.2/H.246 – Calling Party Number**

<u>←SETUP</u>	<u>←IAM</u>
<b><u>Calling Party Number</u></b>	<b><u>Calling Party Number</u></b>

C) If the calling party number and the generic number (with the qualification additional Calling Party Number) are present, then Table C.60.3 applies:

**Table C.60.3/H.246 – Calling Party Number**

<u>←SETUP</u>	<u>←IAM</u>

<u>Calling Party Number</u>	<u>Generic Number</u> <u>(– additional Calling Party number)</u>
<u>additionalSourceAddresses</u>	<u>Calling Party Number</u>

See Table C.60.

**Table C.60/H.246 – Calling Party Number**

<b>← SETUP</b>	<b>← IAM</b>
<b>Calling Party Number</b>	<b>Calling Party Number</b> or (Note) <b>Generic Number</b> <b>(– additional Calling Party number)</b>
<b>sourceAddress</b>	<b>Calling Party Number</b>
NOTE – If an additional Calling Party number is included in the Generic Number, then the additional Calling party number should be sent in the Calling Party Number.	

*–[End Correction]*

#### 5.4.1.2 ACM sending without cause when PI=8

<b>Description:</b>	The current text seems to indicate an SS7 ACM message should not be sent when the Progress Indicator had value 8 (in-band information or an appropriate pattern is now available) in a Call Proceeding or Progress message. This is incorrect since the SS7 backward talk path will not be set up without sending ACM and there will be no way for the in-band information to make it to the calling party without sending ACM.
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

*[Begin Correction]*

#### C.7.1.3.2 ACM without cause parameter

The following cases are possible trigger conditions of sending the address complete message (ACM) without cause parameter:

- The destination has determined independently of access indications that the complete called party number has been received.
- Overlap receiving is used on the H.225.0 side and a CALL PROCEEDING is received.
- En bloc* receiving is used on the H.225.0 side and a Progress indicator information element (except with value No. 8, *in-band information or an appropriate pattern is now available*, No. 3, *originating address is non-ISDN*, or No. 4, *call has returned to the ISDN*) is received in a CALL PROCEEDING message or in a PROGRESS message.
- The first ALERTING message is received.

On *speech* or *3.1 kHz* calls, the awaiting answer indication (e.g. ring tone) is sent to the calling party upon receipt of the first ALERTING message.

NOTE 1 – In all cases, it is assumed that no Address Complete Message (ACM) has already been sent.

NOTE 2 – The case of the sending of the Address Complete Message (ACM) when the call is forwarded is not described hereafter: see C.7.2.

---

[End Correction]

### 5.4.1.3 Support for Location Number

<b>Description:</b>	This text adds support for Location Number in H.246 Annex C
---------------------	-------------------------------------------------------------

---

[Begin Correction]

#### C.6.1.1.1 Mandatory parameters

...

##### Location number

This parameter is sent only if the parameter LocationSourceAddress is present inside SETUP message.

– Nature of address indicator:

According to the field PublicTypeOfNumber in the alias address of the LocationSourceAddress parameter.

– Internal network number indicator:

1 routing to internal network number not allowed

– Numbering plan indicator:

001 ISDN (telephony) numbering plan (Recommendation E.164)

– Address presentation restricted indicator:

According to the field presentationIndicator of the LocationSourceAddress parameter.

– Address screening indicator:

According to the field screeningIndicator of the LocationSourceAddress parameter.

– Address signal:

According to the LocationSourceAddress parameter received in the SETUP.

#### C.6.1.1.2 Optional parameters

##### **Calling party number**

See Table C.5.

---

[End Correction]

---

[Begin Correction]

#### C.7.1.1 Sending of the SETUP message

...

##### LocationSourceAddress

This parameter is sent only if the parameter Location Number is present inside IAM message. See Rec. H.460.LocationSourceaddress.

- Address: partyNumber: e.164Number: publicTypeOf Number:

According to the field Nature of address indicator of the location number parameter.

- Address: PartyNumber: e.164Number: publicNumberDigits:

According to the field address signal of the location number parameter.

- PresentationIndicator

According to the field address presentation restricted indicator of the location number parameter.

- ScreeningIndicator

According to the field address screening indicator of the location number parameter.

### **Low layer compatibility**

FFS.

### **High layer compatibility**

FFS.

### **User-to-user**

The user-to-user information element contains the Setup-UUIE defined in the H.225.0 Message Syntax.

### **C.7.1.2 Receipt of the Subsequent Address Message (SAM)**

If *en bloc* sending is used on the H.225.0 side, the SETUP message shall contain all the information required by the called user to process the call (see C.7.1.1).

---

*[End Correction]*

## **5.5 Technical and Editorial Corrections to ITU-T Recommendation H.235 (2003)**

## **5.6 Technical and Editorial Corrections to ITU-T Recommendation H.450 Series**

### **5.6.1 Technical and Editorial Corrections to H.450.1 (1998)**

#### **5.6.1.1 Actions at a Destination Entity**

<b>Description:</b>	Typographical errors have been discovered in section 6.6 of H.450.1 (1998). The text below outlines the necessary changes.
---------------------	----------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

1) Section 6.6, line 6

Change:

"rejectUnrecognizedInvokePdu"

to

"rejectAnyUnrecognizedInvokePdu"

2) Section 6.6, line 12

Change:

"discardAnyUnrecognizedInvokePDU"

to

"discardAnyUnrecognizedInvokePdu"

---

[End Correction]

### 5.6.1.2 Corrections to the ASN.1

<b>Description:</b>	H.225.0 (1999) introduces redundancy with H.450.1 in that both H.225.0 (1999) and H.450.1 have screening and presentation information. To remove the redundancy, it was decided that H.225.0 was the proper place for this information and the redundant elements shall be removed from H.450.1. Below shows the revision to the ASN.1 found in Table 6/H.450.1.
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

[Begin Correction]

```
Addressing-Data-Elements
{ itu-t recommendation h 450 1 version1(0) addressing-data-elements(9) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
IMPORTS      AliasAddress, PartyNumber, PresentationIndicator, Screening Indicator FROM
H323-MESSAGES; -- see H.225.0

...
-- PartyNumber defined in Recommendation H.225.0
-- PublicPartyNumber defined in Recommendation H.225.0
-- PrivatePartyNumber defined in Recommendation H.225.0
-- NumberDigits defined in Recommendation H.225.0
-- PublicTypeOfNumber defined in Recommendation H.225.0
-- PrivateTypeOfNumber defined in Recommendation H.225.0
-- PresentationIndicator defined in Recommendation H.225.0 (v3 and beyond)
-- ScreeningIndicator defined in Recommendation H.225.0 (v3 and beyond)

EndpointAddress ::= SEQUENCE{
    destinationAddress      SEQUENCE OF AliasAddress,
        -- multiple alias addresses may be used to address the same H.323 endpoint
    remoteExtensionAddress  AliasAddress OPTIONAL,
    ...
    destinationAddressPresentationIndicator PresentationIndicator OPTIONAL,
        -- Note 1, 2
    destinationAddressScreeningIndicator ScreeningIndicator OPTIONAL,
    remoteExtensionAddressPresentationIndicator PresentationIndicator OPTIONAL,
        -- Note 1, 2
    remoteExtensionAddressScreeningIndicator ScreeningIndicator OPTIONAL
}
-- Note 1: If this element is not available, presentation allowed shall be
assumed.
-- Note 2: If an H.450 APDU that carries this element EndpointAddress also
-- contains an element PresentationAllowedIndicator, then the setting of the
-- element PresentationAllowedIndicator shall take precedence in case of
-- conflicting presentation information.

...

ScreeningIndicator ::= ENUMERATED {
    userProvidedNotScreened (0),
        number was provided by a remote user
        , and has not been screened by a gatekeeper
    userProvidedVerifiedAndPassed (1),
        number was provided by a user
        equipment (or by a remote network), and has
        been screened by a gatekeeper
    userProvidedVerifiedAndFailed (2),
        not used, value reserved.
    networkProvided (3),
        number was provided by a gatekeeper
    ...
}
```

---

[End Correction]

### 5.6.1.3 Clarifications to ROS APDUs

<b>Description:</b>	The ASN.1 specification of ROS APDUs has caused some uncertainty over the correct encoding of invoke identifiers. A correct encoding is essential for interoperability between different implementations. The text below attempts to clarify the uncertainty.
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Add the following note below Table 4/H.450.1:

---

[Begin Correction]

Note:

In the *Invoke* APDU, the *invokeID* is an INTEGER constrained by a PER-visible constraint (*InvokeIdSet* = 0..65535) and is therefore encoded as a **constrained** INTEGER (16 bits, no length field). In the *ReturnResult* and *ReturnError* APDUs, however, the *invokeID* is encoded as an **unconstrained** INTEGER (with explicit length field) because the applicable constraint (“must be that for an outstanding operation...” ) is not PER-visible. In the *Reject* APDU the *invokeID* is also encoded as an **unconstrained** INTEGER (with explicit length field) since no constraint applies.

---

[End Correction]

### 5.6.1.4 Error Definitions

<b>Description:</b>	Error description as defined in Table 8 need to be clarified. The text below provides the needed clarifications.
---------------------	------------------------------------------------------------------------------------------------------------------

---

[Begin Correction]

## 10.3 General error list

Table 8 contains the definitions of general errors that may be used within H.450.x Recommendations.

Note – Every operation defined in any H.450.x Recommendation lists all permitted error values explicitly. This means that the errors defined here are not automatically part of another H.450.x Recommendation. An H.450.x Recommendation that uses one of the error values below must import it from the module defined below before it can be used for an operation of H.450.x.

**Table 8/H.450.1 – H.450.1 General Error List**

```
H4501-General-Error-List
  { itu-t recommendation h 450 1 version1(0) general-error-list (1) }

DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
IMPORTS ERROR FROM Remote-Operations-Information-Objects
  { joint-iso-itu-t remote-operations(4) informationObjects(5)
  version1(0) };
```

**Table 8/H.450.1 – H.450.1 General Error List**

<i>-- The following errors are based on the error definitions of Recommendation Q.950.</i>	
<b>userNotSubscribed</b>	<b>ERROR ::= {CODE local:0}</b>
<i>-- is an indication that the user has not subscribed to this service.</i>	
<b>RrejectedByNetwork</b>	<b>ERROR ::= {CODE local:1}</b>
<i>-- is an indication that the requested service is rejected by the network -- (e.g. gatekeeper).</i>	
<b>RrejectedByUser</b>	<b>ERROR ::= {CODE local:2}</b>
<i>-- is an indication that <del>the requested service is provided by the</del> <del>network but that the remote user has rejected this service request.</del></i>	
<b>NnotAvailable</b>	<b>ERROR ::= {CODE local:3}</b>
<i>-- is an indication that <del>the user has subscribed to this service but the</del> <del>requested service is not available under the specific circumstances combined with the basic service or</del> <del>the other services (e.g. operation).</del></i>	
<b>IinsufficientInformation</b>	<b>ERROR ::= {CODE local:5}</b>
<i>-- is an indication that <del>the content of some essential information is missing, operation argument is</del> <del>although the operation argument is formally correct incomplete, or absent entirely.</del></i>	
<b>IinvalidServedUserNumber</b>	<b>ERROR ::= {CODE local:6}</b>
<i>-- is an indication that the requested service cannot be performed -- because of the usage of an invalid served user number.</i>	

**Table 8/H.450.1 – H.450.1 General Error List (concluded)**

<b>IinvalidCallState</b>	<b>ERROR ::= {CODE local:7}</b>
<p>-- is an indication that <del>no match exists between the service request is and</del>  -- <del>incompatible with the valid current H.225.0 call state, this applies also to invalid</del>  -- <del>auxiliary states or an invalid combination of H.225.0 call states and</del>  -- <del>auxiliary states.</del></p>	
<b>BbasicServiceNotProvided</b>	<b>ERROR ::= {CODE local:8}</b>
<p>-- is an indication that the service request <del>refers to an unsupported is directed to a Basic</del>  -- <del>terminal capability. Service which is not provided (e.g. this return error value is used</del>  -- <del>in cases where a supplementary service is to be invoked with a SETUP</del>  -- <del>message but indicating the wrong Basic Service).</del></p>	
<b>NnotIncomingCall</b>	<b>ERROR ::= {CODE local:9}</b>
<p>-- is an indication that the service request has been invoked for an  -- outgoing call, which is not permitted for that service.</p>	
<b>SsupplementaryServiceInteractionNotAllowed</b>	<b>ERROR ::= {CODE local:10}</b>
<p>-- is an indication that the Sservice request is not permitted in  -- combination with either a further requested or active supplementary  -- service.</p>	
<b>RresourceUnavailable</b>	<b>ERROR ::= {CODE local:11}</b>
<p>-- is an indication that the service provider has temporarily no  -- resource available for the provision of the requested service.</p>	
<b>CcallFailure</b>	<b>ERROR ::= {CODE local:25}</b>
<p>-- is an indication that the requested supplementary service was not executable by virtue  -- of a Basic Call Failure. The parameter is included under circumstances where the  -- call failure was remote from the local gatekeeper interface over which the error is to  -- be sent. For example when:  -- a) no H.225.0 RELEASE COMPLETE message is provided locally, or  -- b) the cause information element included in the RELEASE COMPLETE  -- message represents only the reason for local basic call clearing.  -- In these cases the parameter value represents the clearing cause included in the  -- remote clearing procedure.</p>	
<b>PproceduralError</b>	<b>ERROR ::= {CODE local:43}</b>
<p>-- is an indication that a transport message (e.g. SETUP)  -- is received which has one or more operation APDUs which have a valid  -- information content of  -- is not compatible with the procedural context in which it is received content but which are not specified as valid  -- information content of  -- the transport message used.</p>	
<p>END -- of H4501-General-Error-List</p>	

[End Correction]

## 5.6.2 Technical and Editorial Corrections to H.450.2 (1998)

### 5.6.2.1 Editorial Corrections

<b>Description:</b>	Typographical errors have been discovered in sections 11.4.2, 11.5.2, 11.6.2, and 13.4 of H.450.2. The text below outlines the necessary changes.
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------

- 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.




---

[End Correction]

### 5.6.2.2 Clarification of CallIdentifier and ConferenceIdentifier

<b>Description:</b>	<p>A clarification of the setting of H.225.0 elements <b>CallIdentifier</b> and <b>ConferenceIdentifier</b> values in conjunction with H.450.2 transferred calls has been added within a new clause 10.7 "Interactions with H.225.0 parameters".</p> <p><i>Special Note: This section appeared in the May 1999 Implementors' Guide, but stated that the CallIdentifier should be the same for transferred calls. That definition contradicted H.323v2's definition of the CallIdentifier, so this section has been changed to align with H.323v2 and higher.</i></p>
---------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

[Begin Correction]

#### 10.7 Interactions with H.225.0 parameters

The H.225.0 CallIdentifier value of the transferred call shall use a new value, rather than 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 Correction]

### 5.6.2.3 Transfer without Consultation

<b>Description:</b>	<p>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.</p>
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

[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 Correction]

---

### **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 Correction]

## 5.6.3 Technical and Editorial Corrections to H.450.3 (1998)

### 5.6.3.1 Editorial Correction in H.450.3

<b>Description:</b>	Typographical errors have been discovered in H.450.3 clause 12 SDLs.
---------------------	----------------------------------------------------------------------

---

[Begin Correction]

---

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.



---

[End Correction]

### 5.6.3.2 Clarification of the CallIdentifier and ConferenceIdentifier

<b>Description:</b>	<p>A clarification of the setting of H.225.0 elements <b>CallIdentifier</b> and <b>ConferenceIdentifier</b> 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".</p> <p><i>Special Note: This section appeared in the May 1999 Implementors' Guide, but stated that the CallIdentifier should be the same for diverted calls. That definition contradicted H.323v2's definition of the CallIdentifier, so this section has been changed to align with H.323v2 and higher.</i></p>
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

[Begin Correction]

#### 9.9.3 Interactions with H.225.0 parameters

The H.225.0 CallIdentifier of a forwarded call shall use a new value, rather than the value that was used in the forwarding call.

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 Correction]

### 5.6.3.3 Correction to the ASN.1

<b>Description:</b>	<p>A typographical error has been discovered in the ASN.1 definitions presented in H.450.3, Chapter 11.</p>
---------------------	-------------------------------------------------------------------------------------------------------------

---

[Begin Correction]

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

---

[End Correction]

### 5.6.4 Technical and Editorial Corrections to H.450.4 (1999)

#### 5.6.4.1 Change Relating to Interpretation APDU

<b>Description:</b>	<p>In order to align H.450.4 with other H.450-series A modified description of the Call Hold Interpretation APDU (i-apdu) setting has been added in clause 6 of Recommendation H.450.4.</p> <p>This information will be contained in the revision 2 of H.450.4</p>
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Recommendation to be published by the ITU-T. The modified text is shown below.
--------------------------------------------------------------------------------

---

*[Begin Correction]*

## 6 Messages and Information elements

...

When conveying the Invoke APDU of operations **remoteHold** and **remoteRetrieve**, the Interpretation APDU shall be omitted or shall contain the value **rejectAnyUnrecognizedInvokePdu**.

---

*[End Correction]*

### 5.6.4.2 Feature Interaction between H.450.4 and H.450.2

<b>Description:</b>	A modified description of the Call Hold interaction with Call Transfer has been added in clause 9.2.1 of Recommendation H.450.4.  This information will be contained in the revision 2 of H.450.4 Recommendation to be published by the ITU-T. The modified text is shown below.
---------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

#### 9.2.1 Call Transfer (H.450.2)

If prior to Consultation, the first call has been put on hold, the served User endpoint shall decide whether or not to automatically retrieve the held User before Call Transfer is invoked.

- If the served User endpoint decides for the automatic retrieve option, a **retrieveNotific** Invoke APDU (in case of near end call hold) or a **remoteRetrieve** Invoke APDU (in case of remote-end call hold) may either be sent by the served user prior to the message containing the **callTransferInitiate** Invoke APDU or may be sent within the same message containing the **callTransferInitiate** Invoke APDU.

If call transfer fails after retrieval from hold was successful (i.e. if callTransferInitiate Return Error or Reject APDU is received or if timer CT-T3 expires), the served user endpoint may automatically re-invoke SS-Hold.

If remote-end call hold retrieval is unsuccessful, in order to proceed with call transfer the remoteRetrieve Return Error or remoteRetrieve Reject APDU should be disregarded.

- If the served User endpoint decides to not choose the automatic retrieve option, call hold applies to the primary call until call transfer has been completed successfully (i.e. until the primary call is cleared). If transfer fails, the primary call remains being held by User A.

---

*[End Correction]*

## 5.6.5 Technical and Editorial Corrections to H.450.5 (1999)

### 5.6.5.1 Clarification of the CallIdentifier

<b>Description:</b>	<p>A clarification of the setting of H.225.0 element CallIdentifier in conjunction with H.450.5 parked calls has been added within clause 8.3 "Interactions with H.225.0 parameters".</p> <p>This information will be contained in the revision 2 of H.450.5 Recommendation to be published by the ITU-T. The modified text is shown below.</p>
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

### 8.3 Interaction with H.225.0 parameters

The H.225.0 **CallIdentifier** value within a parked call shall use a new value, rather~~be set to~~ the CallIdentifier value that was used in the primary call. For all other SETUP messages carrying SS-PARK or SS-PICKUP related APDUs as defined within this recommendation, new CallIdentifier values shall be used. Note that the CallIdentifier value of the parked/alerting call is preserved during the SS-PARK / SS-PICKUP procedure within the H.450 APDUs.

---

*[End Correction]*

## 5.6.6 Technical and Editorial Corrections to H.450.7 (1999)

### 5.6.6.1 Change Relating to Interpretation APDU

<b>Description:</b>	<p>In order to align H.450.7 with other H.450-series, a modified description of the Message Waiting Indication Interpretation APDU (i-apdu) setting has been added in clause 7.1.1 of Recommendation H.450.7.</p> <p>This information will be contained in the revision 2 of H.450.7 Recommendation to be published by the ITU-T. The modified text is shown below.</p>
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

### 7.1.1 H.450.1 Supplementary Service APDU

...

When conveying the Invoke APDU of operations **mwiActivate**, **mwiDeactivate**, and **mwiInterrogate**, the interpretation APDU shall be omitted or shall contain the value **rejectAnyUnrecognizedInvokePdu**.~~This is implicitly equivalent to specifying an interpretation APDU of rejectAnyUnrecognizedInvokePDU.~~

---

*[End Correction]*

## 5.6.7 Technical and Editorial Corrections to H.450.8 (2000)

### 5.6.7.1 Usage of CalledName and AlertingName

<b>Description:</b>	An editorial error has been found in the H.450.8 (2000) Recommendation in the usage of calledName and alteringName. The following text corrects the errors.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

---

## 7.2 Terminals or MCU as Originating Endpoint

...

A terminal or MCU in receipt of an H.225.0 Connect, Alerting, or Release Complete message containing a connectedName, ~~called~~alteringName, or busyName APDU should not present name information if the Name element indicates namePresentationRestricted.

## 8.2 Terminals or MCU as Terminating Endpoint

A terminal or MCU in receipt of the H.225.0 Setup message may include name information in the Connect, Alerting or Release Complete as described above in 6.2, 6.3 or 6.4. If presentation of the name to the calling party is desirable, the Name element in the alteringName, connectedName, or busyName operation should indicate namePresentationAllowed. If presentation of the name to the called party is to be restricted, the Name element in the ~~called~~alteringName, connectedName, or busyName operation should indicate namePresentationRestricted.

---

*[End Correction]*

---

## 5.6.8 Technical and Editorial Corrections to H.450.12 (2001)

### 5.6.8.1 Technical Correction

<b>Description:</b>	The receipt of a CmnInform APDU at User A's Endpoint is not described. Therefore add the text below at the end of section 7.1.1.1 ANF-CMN invocation.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

---

### 7.1.1.1 ANF-CMN invocation

...

Upon receipt of a CmnInform invoke APDU in any message, the Originating endpoint shall remain in the current state.

---

*[End Correction]*

---

### 5.6.8.2 Add definition of the states CMN-Wait-Response and CMN-Wait-Answer-Response

<b>Description:</b>	The states CMN-Wait-Response and CMN-Wait-Answer-Response are used
---------------------	--------------------------------------------------------------------

only in the SDL diagrams but are not defined anywhere. To avoid confusion, a definition of their meaning is added in section 13.
----------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

---

### 13. Specification and Description Language (SDL) Diagrams for ANF-CMN

...

In the following SDLs the states CMN-Wait-Response and CMN-Wait-Answer-Response are used to describe the behavior of the Endpoints using explicit primitive exchange.

The state CMN-Wait-Response is entered at the Endpoint after a primitive CMNRequest indication is received and the previous state was CMN-Idle.

The state CMN-Wait-Answer-Response is entered at the Endpoint after a primitive CMNRequest indication is received and the previous state was CMN-Wait-Answer.

---

*[End Correction]*

---

#### 5.6.8.3 Redesign the SDL Diagrams, add two missing collision branches and delete an erroneous message symbol

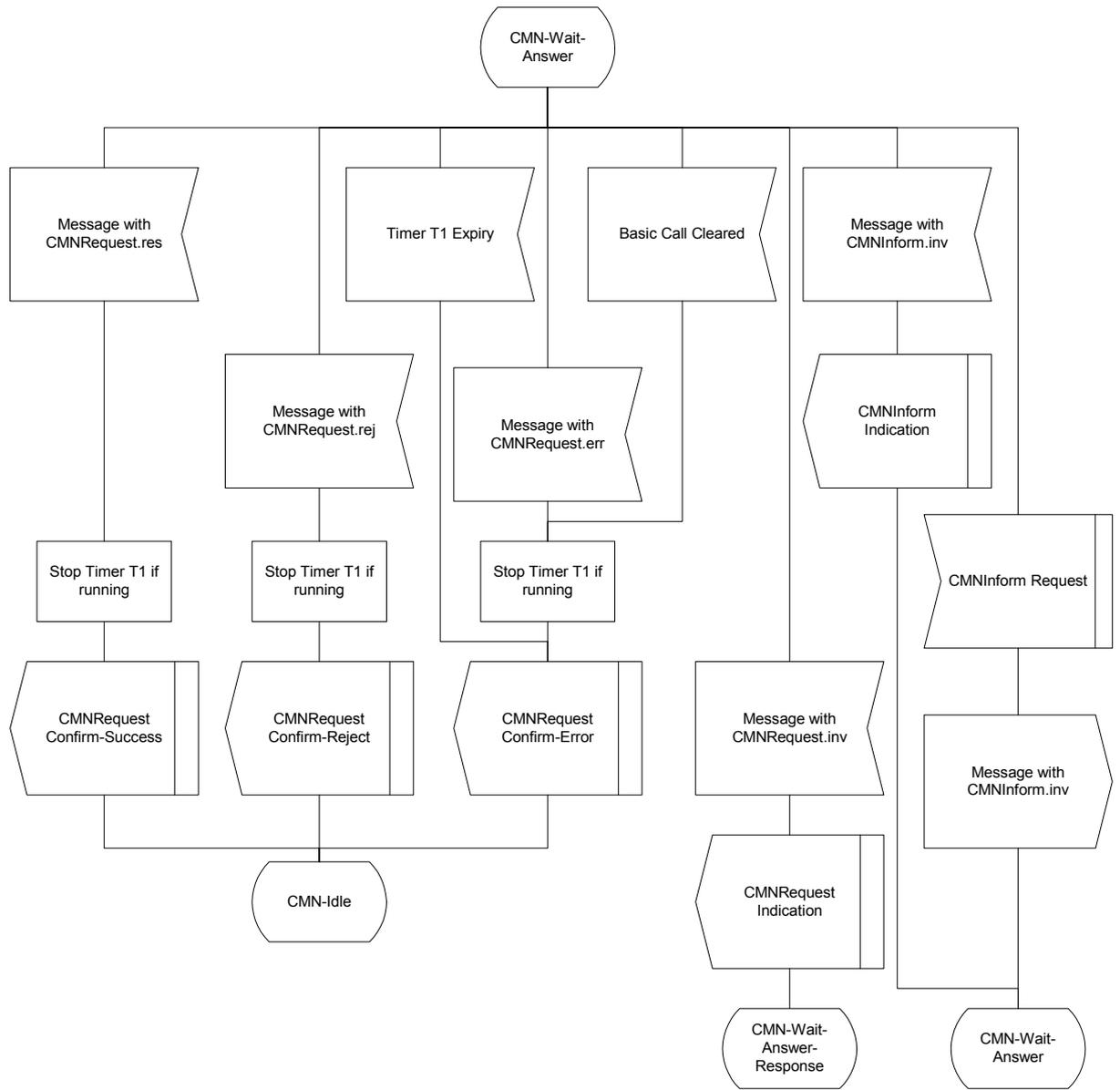
<b>Description:</b>	Two collision branches are missing: add in section 13.1 Figure 8/H.450.12 the possible receipt of a CMNInform request from the application in state CMN-Wait-Answer and in Figure 9/H.450.12 the possible receipt of a CMNRequest request in state CMN-Wait-Response.  In Figure 9/H.450.12 the receipt of a CMNInform Request in state CMN-Wait-Response shall be ignored and the message with CMNInform invoke APDU shall not be forwarded to endpoint B.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

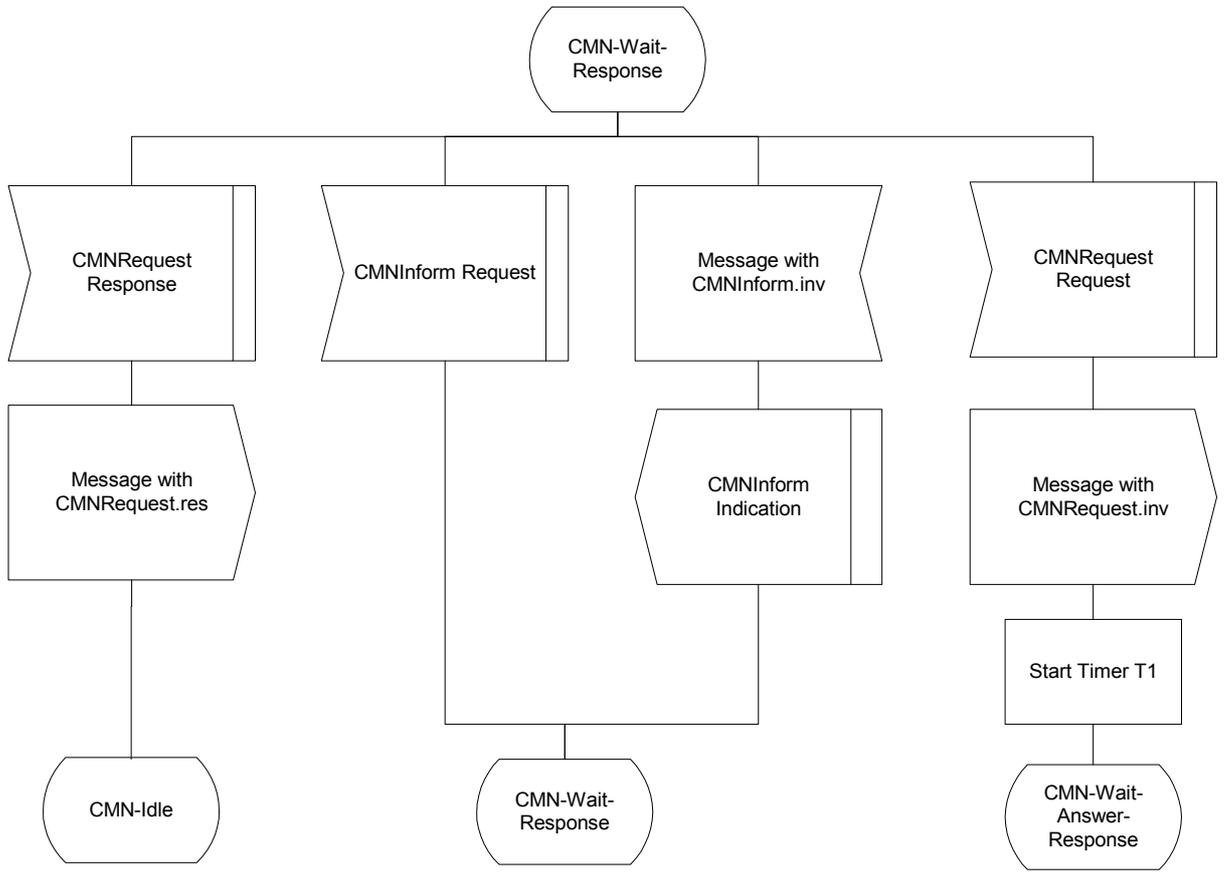
*[Begin Correction]*

---

Editorial - Replace the indicated diagrams by the following:



**Figure 8/H.450.12 – SDL Representation of ANF-CMN at Endpoint A (Part 3)**



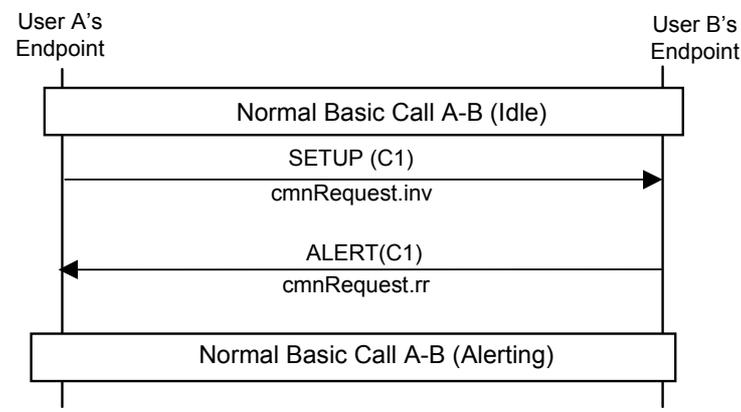
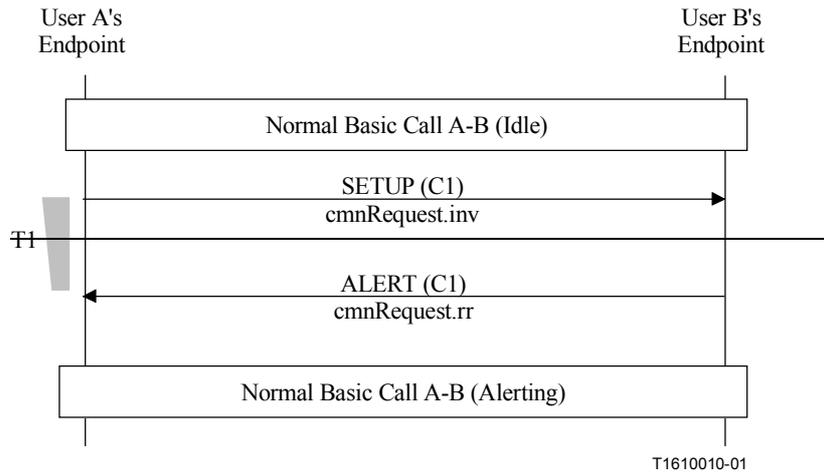
**Figure 9/H.450.12 – SDL Representation of ANF-CMN at Endpoint A (Part 3)**

*[End Correction]*

#### 5.6.8.4 Message Flow of ANF-CMN

<b>Description:</b>	Timer T1 is started if <b>cmnRequest</b> invoke is sent in FACILITY message, but not if it is sent in a SETUP message. However, the message flow diagram in Figure 2/H.450.12 erroneously contains timer T1. The erroneous diagram should be replaced by the corrected diagram as below.
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

[Begin Correction]



[End Correction]

## 5.7 Technical and Editorial Corrections to ITU-T Recommendation H.341 (1999)

### 5.7.1 Corrections to H.341 Annex B-1 H225-MIB

<b>Description:</b>	Each field in <b>CallSignalStatsEntry</b> SEQUENCE referred to the number of messages received ("In") and the number of messages transmitted ("Out"). These counters shall be combined.
---------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

[Begin Correction]

```
CallSignalStatsEntry ::= SEQUENCE {  
    callSignalStatsCallConnectionsIn  
        Counter32,  
    callSignalStatsCallConnectionsOut  
        Counter32,  
    callSignalStatsAlertingMsgsIn  
        Counter32,  
    callSignalStatsAlertingMsgsOut  
        Counter32,  
    callSignalStatsCallProceedingsIn  
        Counter32,  
    callSignalStatsCallProceedingsOut  
        Counter32,  
}
```

```

callSignalStatsSetupMsgsIn
    Counter32,
callSignalStatsSetupMsgsOut
    Counter32,
callSignalStatsSetupAckMsgsIn
    Counter32,
callSignalStatsSetupAckMsgsOut
    Counter32,
callSignalStatsProgressMsgsIn
    Counter32,
callSignalStatsProgressMsgsOut
    Counter32,
callSignalStatsReleaseCompleteMsgsIn
    Counter32,
callSignalStatsReleaseCompleteMsgsOut
    Counter32,
callSignalStatsStatusMsgsIn
    Counter32,
callSignalStatsStatusMsgsOut
    Counter32,
callSignalStatsStatusInquiryMsgsIn
    Counter32,
callSignalStatsStatusInquiryMsgsOut
    Counter32,
callSignalStatsFacilityMsgsIn
    Counter32,
callSignalStatsFacilityMsgsOut
    Counter32,
callSignalStatsInfoMsgsIn
    Counter32,
callSignalStatsInfoMsgsOut
    Counter32,
callSignalStatsNotifyMsgsIn
    Counter32,
callSignalStatsNotifyMsgsOut
    Counter32,
callSignalStatsAverageCallDuration
    Integer32,
callSignalStatsCallConnections
    Counter32,
callSignalStatsAlertingMsgs
    Counter32,
callSignalStatsCallProceedings
    Counter32,
callSignalStatsSetupMsgs
    Counter32,
callSignalStatsSetupAckMsgs
    Counter32,
callSignalStatsProgressMsgs
    Counter32,
callSignalStatsReleaseCompleteMsgs
    Counter32,
callSignalStatsStatusMsgs
    Counter32,
callSignalStatsStatusInquiryMsgs
    Counter32,
callSignalStatsFacilityMsgs
    Counter32,
callSignalStatsInfoMsgs
    Counter32,
callSignalStatsNotifyMsgs
    Counter32
}

```

```

callSignalStatsCallConnectionsIn OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of successful connections in which this entity
         has been a callee."
    ::= { callSignalStatsEntry 1 }
callSignalStatsCallConnectionsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of successful connections in which this entity
         has been a caller."
    ::= { callSignalStatsEntry 2 }
callSignalStatsAlertingMsgsIn OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of alerting messages received by this entity."
    ::= { callSignalStatsEntry 3 }
callSignalStatsAlertingMsgsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of alerting messages sent by this entity."
    ::= { callSignalStatsEntry 4 }
callSignalStatsCallProceedingsIn OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of call proceeding messages received by this
         entity."
    ::= { callSignalStatsEntry 5 }
callSignalStatsCallProceedingsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of call proceeding messages sent by this entity."
    ::= { callSignalStatsEntry 6 }

callSignalStatsSetupMsgsIn OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of setup messages received by this entity."
    ::= { callSignalStatsEntry 7 }
callSignalStatsSetupMsgsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of setup messages sent by this entity."
    ::= { callSignalStatsEntry 8 }
callSignalStatsSetupAckMsgsIn OBJECT-TYPE

```

```

SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION
    "The number of setupAck messages received by this entity."
 ::= { callSignalStatsEntry 9 }
callSignalStatsSetupAckMsgsOut OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION
    "The number of setupAck messages sent by this entity."
 ::= { callSignalStatsEntry 10 }
callSignalStatsProgressMsgsIn OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION
    "The number of progress messages received by this entity."
 ::= { callSignalStatsEntry 11 }
callSignalStatsProgressMsgsOut OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION
    "The number of progress messages sent by this entity."
 ::= { callSignalStatsEntry 12 }

callSignalStatsReleaseCompleteMsgsIn OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION
    "The number of release complete messages received by this
    entity."
 ::= { callSignalStatsEntry 13 }
callSignalStatsReleaseCompleteMsgsOut OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION
    "The number of release complete messages sent by this
    entity."
 ::= { callSignalStatsEntry 14 }
callSignalStatsStatusMsgsIn OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION
    "The number of status messages received by this entity."
 ::= { callSignalStatsEntry 15 }
callSignalStatsStatusMsgsOut OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION
    "The number of status messages sent by this entity."
 ::= { callSignalStatsEntry 16 }
callSignalStatsStatusInquiryMsgsIn OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      currentdeprecated
DESCRIPTION

```

```

        "The number of status inquiry messages received by this
        entity."
    ::= { callSignalStatsEntry 17 }

callSignalStatsStatusInquiryMsgsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of status inquiry messages sent by this
        entity."
    ::= { callSignalStatsEntry 18 }

callSignalStatsFacilityMsgsIn OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of connect messages received by this entity."
    ::= { callSignalStatsEntry 19 }

callSignalStatsFacilityMsgsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of connect messages sent by this entity."
    ::= { callSignalStatsEntry 20 }

callSignalStatsInfoMsgsIn OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of info messages received by this entity."
    ::= { callSignalStatsEntry 21 }

callSignalStatsInfoMsgsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of info messages sent by this entity."
    ::= { callSignalStatsEntry 22 }

callSignalStatsNotifyMsgsIn OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of notify messages received by this entity."
    ::= { callSignalStatsEntry 23 }

callSignalStatsNotifyMsgsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      currentdeprecated
    DESCRIPTION
        "The number of notify messages sent by this entity."
    ::= { callSignalStatsEntry 24 }

callSignalStatsAverageCallDuration OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The average duration of the call in minutes since
        system boot time. "

```

```

 ::= { callSignalStatsEntry 25 }

callSignalStatsCallConnections OBJECT-TYPE
  SYNTAX Counter32
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
    "The number of successful connections ."
  ::= { callSignalStatsEntry 26 }

callSignalStatsAlertingMsgs OBJECT-TYPE
  SYNTAX Counter32
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
    "The number of alerting messages."
  ::= { callSignalStatsEntry 27 }

callSignalStatsCallProceedings OBJECT-TYPE
  SYNTAX Counter32
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
    "The number of call proceeding messages."
  ::= { callSignalStatsEntry 28 }

callSignalStatsSetupMsgs OBJECT-TYPE
  SYNTAX Counter32
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
    "The number of setup messages."
  ::= { callSignalStatsEntry 29 }

callSignalStatsSetupAckMsgs OBJECT-TYPE
  SYNTAX Counter32
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
    "The number of setupAck messages."
  ::= { callSignalStatsEntry 30 }

callSignalStatsProgressMsgs OBJECT-TYPE
  SYNTAX Counter32
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
    "The number of progress messages."
  ::= { callSignalStatsEntry 31 }

callSignalStatsReleaseCompleteMsgs OBJECT-TYPE
  SYNTAX Counter32
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
    "The number of release complete messages."
  ::= { callSignalStatsEntry 32 }

callSignalStatsStatusMsgs OBJECT-TYPE
  SYNTAX Counter32
  MAX-ACCESS read-only

```

```

        STATUS current
        DESCRIPTION
        "The number of status messages."
        ::= { callSignalStatsEntry 33 }

callSignalStatsStatusInquiryMsgs OBJECT-TYPE
    SYNTAX Counter32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
    "The number of status inquiry messages."
    ::= { callSignalStatsEntry 34 }

callSignalStatsFacilityMsgs OBJECT-TYPE
    SYNTAX Counter32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
    "The number of connect messages."
    ::= { callSignalStatsEntry 35 }

callSignalStatsInfoMsgs OBJECT-TYPE
    SYNTAX Counter32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
    "The number of info messages."
    ::= { callSignalStatsEntry 36 }

callSignalStatsNotifyMsgs OBJECT-TYPE
    SYNTAX Counter32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
    "The number of notify messages."
    ::= { callSignalStatsEntry 37 }

```

---

*[End Correction]*

### 5.7.2 Corrections to H.341 Annex B-2 RAS-MIB

<b>Description:</b>	A few editorial errors have been identified in the RAS MIB in H.341. The <b>rasAdmissionCallIdentifier</b> field is inserted twice in the <b>RasAdmissionTableEntry</b> SEQUENCE. The ASN type of <b>rasRegistrationEndpointType</b> field in <b>RasRegistrationTableEntry</b> SEQUENCE should be changed to <b>MmH323EndpointType</b> .
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

```

RAS-MIB DEFINITIONS ::= BEGIN
    IMPORTS
        Counter32, Integer32, OBJECT-TYPE,
        MODULE-IDENTITY, NOTIFICATION-TYPE
            FROM SNMPv2-SMI
        TAddress, RowStatus, TruthValue, DateAndTime
            FROM SNMPv2-TC
        MODULE-COMPLIANCE, OBJECT-GROUP
            FROM SNMPv2-CONF
    ifIndex

```

```

FROM IF-MIB
MmGatekeeperID, MmTAddressTag, MmEndpointID, MmGlobalIdentifier,
MmAliasTag, MmAliasAddress, mmH323Root, MmH323EndpointType,
MmH225Crv, MmCallType
FROM MULTI-MEDIA-MIB-TC;

```

```

...
RasRegistrationTableEntry ::=
SEQUENCE {
rasRegistrationCallSignallingAddressTag
MmTAddressTag,
rasRegistrationCallSignallingAddress
TAddress,
rasRegistrationSrcRasAddressTag
MmTAddressTag,
rasRegistrationSrcRasAddress
TAddress,
rasRegistrationIsGatekeeper
TruthValue,
rasRegistrationGatekeeperId
MmGatekeeperID,
rasRegistrationEndpointId
MmEndpointID,
rasRegistrationEncryption
TruthValue,
rasRegistrationWillSupplyUUIE
TruthValue,
rasRegistrationIntegrityCheckValue
TruthValue,
rasRegistrationTableNumberOfAliases
Integer32,
rasRegistrationTableRowStatus
RowStatus,
rasRegistrationEndpointType
MmH323EndpointTypeInteger32,
rasRegistrationPregrantedARQ
TruthValue,
rasRegistrationIsregisteredByRRQ
TruthValue
}

```

```

...
rasRegistrationEndpointType OBJECT-TYPE
SYNTAX MmH323EndpointTypeInteger32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Terminal type represents the type of H.323 terminal:

```

```

...
RasAdmissionTableEntry ::=
SEQUENCE {
rasAdmissionSrcCallSignallingAddressTag
MmTAddressTag,
rasAdmissionSrcCallSignallingAddress
TAddress,
rasAdmissionDestCallSignallingAddressTag
MmTAddressTag,
rasAdmissionDestCallSignallingAddress
TAddress,
rasAdmissionCallIdentifier
MmGlobalIdentifier,
rasAdmissionConferenceId
MmGlobalIdentifier,

```

```

rasAdmissionRasAddressTag
    MmTAddressTag,
rasAdmissionRasAddress
    TAddress,
rasAdmissionCRV
    MmH225Crv,
rasAdmissionIsGatekeeper
    TruthValue,
rasAdmissionSrcAliasAddressTag
    MmAliasTag,
rasAdmissionSrcAliasAddress
    MmAliasAddress,
rasAdmissionDestAliasAddressTag
    MmAliasTag,
rasAdmissionDestAliasAddress
    MmAliasAddress,
rasAdmissionAnswerCallIndicator
    INTEGER,
rasAdmissionTime
    DateAndTime,
rasAdmissionCallIdentifier
    MmGlobalIdentifier,
rasAdmissionEndpointId
    MmEndpointID,
rasAdmissionBandwidth
    Integer32,
rasAdmissionIRRFrequency
    Integer32,
rasAdmissionCallType
    MmCallType,
rasAdmissionCallModel
    INTEGER,
rasAdmissionSrcHandlesBandwidth
    TruthValue,
rasAdmissionDestHandlesBandwidth
    TruthValue,
rasAdmissionSecurity
    TruthValue,
rasAdmissionSrcWillSupplyUUIE
    TruthValue,
rasAdmissionDestWillSupplyUUIE
    TruthValue,
rasAdmissionTableRowStatus
    RowStatus
}

```

---

*[End Correction]*

### 5.7.3 Support for Expanded Country Code Values in T.35 in H.341 Annex B-3

<b>Description:</b>	T.35 (1999) expanded the available country codes from one octet to two octets. In order to support the expanded country codes going forward, it is recommended that implementers make the following changes to these definitions in H.341 Annex B-3 H323TERMINAL-MIB.
---------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

```

h323TermSystemt35CountryCode OBJECT-TYPE
    SYNTAX INTEGER (0..255)
    MAX-ACCESS read-only
    STATUS current

```

```

DESCRIPTION
"Country code, per T.35 Annex A."
 ::= { h323TermSystemEntry 5 }
h323TermSystemt35CountryCodeExtention OBJECT-TYPE
    SYNTAX INTEGER (0..255)
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
"Assigned nationally, unless the country code
is 255, in which case this value shall contain
the country code found in T.35 Annex B."
 ::= { h323TermSystemEntry 6 }

```

---

*[End Correction]*

## 5.8 Technical and Editorial Corrections to ITU-T Recommendation H.283 (1999)

### 5.8.1 Support for Expanded Country Code Values in T.35

<b>Description:</b>	T.35 (1999) expanded the available country codes from one octet to two octets. In order to support the expanded country codes going forward, it is recommended that implementers take note of the following usage guidelines for fields in H.283.
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

```

...
H221NonStandard ::= SEQUENCE
{
    t35CountryCode    INTEGER(0..255),    -- country, as per T.35 Annex A
    t35Extension      INTEGER(0..255),    -- assigned nationally, unless
    the
    t35CountryCode is binary 1111 1111,    --
    in which case this field shall
    contain the
    country code found
    in T.35
    Annex B
    manufacturerCode  INTEGER(0..65535)  -- assigned nationally
}
...

```

---

*[End Correction]*

## 5.9 Technical and Editorial Corrections to ITU-T Recommendation H.460 Series

### 5.9.1 Technical and Editorial Corrections to H.460.1 (2002)

#### 5.9.1.1 Encoding rules

<b>Description:</b>	Currently, H.460.1 requires each individual recommendation to specify the encoding that it will use for parameters in the raw format. It would be better to specify the default encoding in H.460.1 and let individual recommendations change it only if needed. The textual changes are shown
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	below.
--	--------

*[Begin Correction]*

---

## 7.2 Encoded in Raw Method

...

If the feature is defined using ASN.1, then it is recommended that the basic aligned variant of the PER encoding rules be used. However, irrespective of this, ~~the~~ encoding rules that are used, if different from the above, shall be explicitly stated in the specification of the feature.

*[End Correction]*

---

## 5.9.2 Technical and Editorial Corrections to H.460.2 (2001)

### 5.9.2.1 Typographical Error in Section 4.1

<b>Description:</b>	A typographical error has been discovered in that the parameter <b>qorPortedNumber</b> in the ASN.1 is referred to as <b>qorPortedAddress</b> in Section 4.1 of H.460.2 (2001). The text below outlines the necessary change.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

*[Begin Correction]*

---

## 4.1 Messages and Signaling

...

- 5) When a Gatekeeper receives an ARQ or LRQ and determines that the destination number is ported out of the network and it may wish to invoke number portability Query on Release (QoR) procedures (as specified in Annex C/Q.769.1). In such cases, the Gatekeeper must respond with ARJ or LRJ that contains a reject reason of **genericDataReason**. The Gatekeeper should include the **genericData** of the ARJ/LRJ that contains the **NumberPortabilityGenericData** with the **numberPortabilityRejectReason**. The **numberPortabilityRejectReason** now will have a value of **qorPortedNumberAddress** (=1). This maps to the ISUP release cause value = #14 (QoR: ported number) as specified in Addendum 1/Q.850.

*[End Correction]*

---

### 5.9.2.2 Cardinality of Number

<b>Description:</b>	A typographical error has been discovered in that the parameter <b>qorPortedNumber</b> in the ASN.1 is referred to as <b>qorPortedAddress</b> in Section 4.1 of H.460.2 (2001). The text below outlines the necessary change.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

*[Begin Correction]*

---

## 5 H.225.0 Generic Data Usage

...

Generic Extensibility Type	Fields	Field name	Value
EnumeratedParameter			
GenericIdentifier	id	standard	1
Contents	content	raw	ASN.1 PER encoding of the <b>NumberPortabilityInfo</b>
<u>Parameter Cardinality</u>			<u>Once and Only Once</u>

*[End Correction]*

---

### 5.9.3 Technical and Editorial Corrections to H.460.6 (2002)

#### 5.9.3.1 Close All Channels

<b>Description:</b>	The intent of the Close All Media Channels request described in section 4.1.2 is to close all open media channels and cancel all available sessions, as described in section 4.5. To this end, text in sections 4.1.2 and 4.5.2 should be changed as follows.
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

*[Begin Correction]*

---

#### 4.1.2 Close All Channels

This parameter may be used by a party to request that the receiving endpoint close all open media channels and cancel all available sessions. Support for this parameter is optional, and shall be negotiated during EFC feature negotiation.

...

#### 4.5.2 Requesting Close-All-Channels

An endpoint or a third party may request that the other endpoint close all open media channels and cancel all available sessions by sending a **genericData** element with the EFC featureID and parameter 2 present in any convenient call signalling message (e.g., FACILITY). The receiving endpoint is expected to silently close all open channels without any response (e.g., without issuing any Null-OLCs.)

*[End Correction]*

---

#### 5.9.3.2 Signaling of EFC Support in supportedFeatures

<b>Description:</b>	It is held that signalling of EFC in <b>supportedFeatures</b> by the originating party is unnecessary. The text in section 4.2 should be corrected as below.
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------

## 4.2 Invocation of Extended Fast Start

An originating party shall indicate its desire to use EFC when it issues a SETUP message. The SETUP shall contain a request for EFC support in the **desiredFeatures** element, or a requirement for EFC support in the **neededFeatures** element. ~~The **supportedFeatures** element shall indicate support for EFC as well.~~ The EFC feature is symmetric, hence requestor support for the feature may be inferred from a request for EFC, and the **supportedFeatures** element need not be included to indicate support for EFC. In addition, the SETUP message shall include a **genericData** element specifying EFC Proposal (parameter 1) and a **fastStart** element containing one or more proposals. That is, EFC procedures shall include the standard Fast Connect procedures.

---

[End Correction]

### 5.9.3.3 Prevention of Race Condition in Master/Slave Determination

<b>Description:</b>	There is a possible race condition that may occur, depending on the order in which an endpoint processes <b>fastStart</b> elements versus tunnelled H.245 master/slave negotiation messages embedded in the same H.225.0 message. Thus, it is suggested that the following paragraph be added to the end of section 4.2.1.
---------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

[Begin Correction]

#### 4.2.1 Master/Slave Determination

Parties supporting Extended Fast Connect should use the H.245 tunnel to carry out master/slave negotiation. For the initial Fast Connect exchange, the caller (sender of the SETUP with proposals) shall be considered the slave, and the called party (acceptor of proposals) shall act as the master. Although this convention will suffice for simple A-to-B calls, it can lead to complications in more complex call scenarios.

Different implementations may process **fastStart** elements and tunnelled H.245 messages in different orders. EFC proposals or acceptances shall not be included in any H.225.0 message that carries an H.245 **MasterSlaveDeterminationAck** message that conveys a change in master/slave status. Doing so could lead to temporary confusion about which party is master and how to respond to the EFC elements.

---

[End Correction]

### 5.9.3.4 Remote Endpoint Type and Version after Re-routing

<b>Description:</b>	An endpoint may not be aware of the H.323 protocol version number supported by the remote endpoint, especially if the call gets re-routed one or more times. In some cases it might be helpful for the endpoint to have this information. The following additions should be made to H.460.6 document.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 5.5 EFC Third-party Pause and Rerouting

EFC supports third-party pause and rerouting, as described in H.323 Annex F for SETs, when used by a routing gatekeeper. The third party (the gatekeeper in the example in Figure 5) may idle the caller's transmit and/or receive channels via **Null-OLCs**, then supply the caller's proposal **fastStart** to a new party (e.g., in a SETUP). The acceptance **fastStart** will appear to the caller as a redirection or reconfiguration, as illustrated in Figure 5.

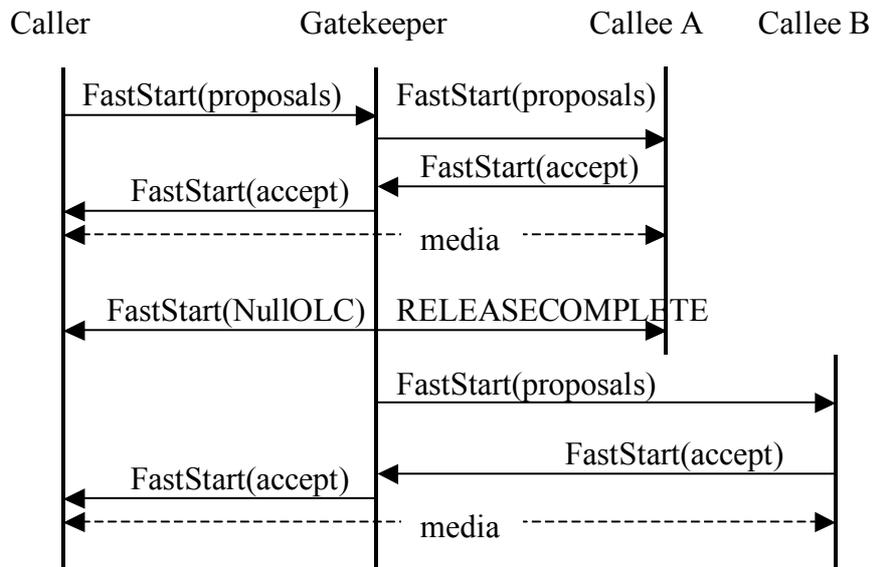


Figure 5/H.460.6 - Third-Party Redirection

In the above diagram, the Gatekeeper, or the entity that re-routes a call should send a Facility message containing the **destinationInfo** field upon completion of the re-routing to the entity that gets re-routed, i.e., Caller. An endpoint should examine this message for the H.225.0 version information at any point that a Facility message is received containing this field.

After coming out of the “paused” state an endpoint should examine the version-id fields in TCS messages to determine the H.245 version supported by the remote endpoint.

In addition, an endpoint interested in knowing the version of the remote endpoint should send a Status Inquiry message and wait for the receipt of the Status message to determine the version of the H.225.0 in use when it exits the paused state when the above Facility message is not received within a reasonable amount of time. The length of this time is left to the implementation.

#### 5.9.3.5 Termination of Extended Fast Connect

<b>Description:</b>	It is held that extended fast connect shall not be terminated when an H.245 address is present in a message returned by the called party. Instead, EFC shall be terminated when a connection is established to the H.245 address. To this end, text in section 4.2 should be changed as follows.
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 4.2 Invocation of Extended Fast Connect

...

Extended Fast ~~Connect Start~~ requires that H.245, if used, shall be tunnelled in the H.225.0 signalling channel. If a calling party offers Extended Fast ~~Connect Start~~-support in SETUP, and the called party returns a message including **h245Address** before returning an EFC response, ~~(thereby requesting a separate H.245 connection)~~, the calling party shall presume that EFC is not supported and may proceed with establishment of the requested connection. Including **h245address** in messages does not by itself terminate EFC, but establishing a connection to an H.245 address will cause termination of EFC.

---

[End Correction]

### 5.9.3.6 Clarification on simultaneous use of EFC acceptance and acceptance fastStart

<b>Description:</b>	EFC acceptance and acceptance <b>fastStart</b> can be simultaneously included in any message up to and including Connect. Having both the fields in the same message can cause undue processing on the recipient and must be avoided when it is possible. To this end, text in section 4.2 should be changed as follows.
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

[Begin Correction]

## 4.2 Invocation of Extended Fast Connect

...

Note that the EFC acceptance and acceptance **fastStart** can be returned in any message up to and including the CONNECT message, but the identical acceptance should not be repeated in any subsequent message.

---

[End Correction]

### 5.9.3.7 Clarification on orientation of logical channels

<b>Description:</b>	Orientation of logical channels proposed using EFC are not related to the direction of the call. The following paragraph should be added to Section 4.3 to clarify this.
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

[Begin Correction]

## 4.3 Opening New Sessions

Opening new media sessions proceeds just like standard Fast Connect, except that either party may invoke EFC at any time to propose new media sessions by sending a message (e.g., FACILITY) containing a proposal **fastStart** element with one or more **OLCs** for one or more **sessionIDs**, along with a **genericData** element indicating “EFC Proposal”. As for standard Fast Connect, multiple **OLCs** with the same **sessionID** are considered to be alternative proposals for a single media stream. The other party may reply with a **fastStart** element containing **OLCs** for the accepted channels and sessions. A slave party will supply a non-zero **sessionID** for any media channels it

proposes. Parties may use EFC to propose and open any number of sessions. Session IDs may take any valid value and need not be limited to the “well-known” values of 1, 2, or 3.

For each logical channel, the EFC proposal establishes the orientation of the forward and reverse logical channels: the forward logical channel carries media from the proposer to the acceptor, and the reverse logical channel carries media from the acceptor to the proposer; the order is not determined from the direction of the call as a whole.

As in standard Fast Connect, once a proposed alternative is selected by another party, the issuing endpoint may suspend any reception of media on the other alternatives. Nevertheless, it shall be prepared for the other party to replace the initially-selected alternative with another (see section 4.7, below).

...

---

*[End Correction]*

## 5.9.4 Technical and Editorial Corrections to H.460.7 (2002)

### 5.9.4.1 Compound Type Parameter Usage

<b>Description:</b>	The contents of the compound parameter in Section 6.4, Table 9 are not well defined. The text below clarifies its usage.
---------------------	--------------------------------------------------------------------------------------------------------------------------

---

*[Begin Correction]*

## 6.4 Digit map string parameters

...

**Table 1/H.460.7 – Type of Number Associated Digit Maps**

Parameter name:	<b>ToN Associated Digit Map</b>
Parameter description:	This compound type conveys Digit Map associated with a particular Type of Number
Parameter identifier type:	Standard
Parameter identifier value:	5
Parameter type:	Compound
Parameter cardinality:	Zero or more

Within the **compound type** defined in Table 9, the parameters defined in Table 2 and Table 11 shall be included to convey one or more Digit Map strings for a particular Type of Number:

**Table 2/H.460.7 – Type of Number Parameter**

Parameter name:	<b>Type of Number (ToN)</b>
Parameter description:	This parameter indicates the type of number
Parameter identifier type:	Standard
Parameter identifier value:	1
Parameter type:	Number8
Parameter valid values:	1 International number 2 National number 3 Network specific number 4 Subscriber number 6 Abbreviated number
Parameter cardinality:	Once

The Digit Map strings comprising the Digit Map associated with a Type of Number are conveyed as additional parameters within the **compound** type of the Type of Number Associated Digit Maps parameter shown in Table 449. This is shown in Table 11.

**Table 3/H.460.7 - Digit Map strings for ToN Parameter**

Parameter name:	<b>Digit Map Strings for ToN</b>
Parameter description:	This parameter contains a single Digit Map string
Parameter identifier type:	Standard
Parameter identifier value:	2
Parameter type:	Text
Parameter cardinality:	One or more

The syntax of the **text** field, which holds a single Digit Map string, is described in section 10.

The order of the Digit Map strings in the **parameters** field has no significance.

---

*[End Correction]*

#### 5.9.4.2 Duplicate Parameters

<b>Description:</b>	Section 6.3 (Table 3) and Section 6.5 (Table 12) both define a parameter with ID 2. They can be used in the same place (in an RCF), so it is not possible to distinguish between them. The parameter identifier value should be changed to 2 as below.
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 6.5 URL parameter

...

**Table 4/H.460.7 - URL Parameter**

Parameter name:	<b>Digit Map URL</b>
Parameter description:	This parameter contains a URL to Digit Map information accessible via HTTP
Parameter identifier type:	Standard
Parameter identifier value:	<u>26</u>
Parameter type:	Alias
Parameter cardinality:	Zero or one

## 5.9.5 Technical and Editorial Corrections to H.460.8 (2002)

### 5.9.5.1 CallTerminationCause Parameter Usage

<b>Description:</b>	Table 3 defines a parameter for "Call Termination Cause" but does not explicitly define its format. The text below clarifies this.
---------------------	------------------------------------------------------------------------------------------------------------------------------------

## 6 Querying For Alternate Routes

...

TABLE 5

Parameter to contain the Call Termination Cause

Parameter name:	Call Termination Cause
Parameter description:	The <b>CallTerminationCause</b> structure contains the reason for the previously failed call attempt
Parameter identifier type:	Standard
Parameter identifier value:	2
Parameter type:	Raw
Parameter cardinality:	Zero or one

The contents of the raw parameter shall be encoded the same as **CallTerminationCause** as defined in the ASN.1 in Annex H/H.225.0.

[End Correction]

### 5.10 Technical and Editorial Corrections to ITU-T Recommendation Annex P/H.323 (2003)

#### 5.10.1 Allowance for multiple Audio and VBD Streams

<b>Description:</b>	Inconsistencies exist between the current Annex P text and the text found in V.150.1 related to the same functionality for SDP-based systems. In particular, the text for SDP-based systems allows for any number of audio streams and multiple VBD streams. The text below provides the correction.
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

[Begin Correction]

While there are no strict limitations on the number of streams that may be contained within any MPS channel, the MPS channel used for MoIP shall contain ~~no zero or more than one~~ audio streams, ~~no one or more than one~~ VBD streams, no more than one SSE stream, and no more than one SPRT stream. If the SPRT stream is opened as a separate channel, the MPS channel shall not also include an SPRT stream. In addition, there ~~may~~ shall be ~~one different~~ payload types for ~~normal~~ audio, ~~one for the~~ VBD stream, ~~one for the~~ SSE stream, and ~~one for the~~ SPRT streams within the MPS. It is possible that more than four payload types may be utilized for ~~those~~ audio, VBD, SSE, and SPRT ~~four~~ streams. For example, if the VBD stream is protected with Forward Error Correction (FEC), and if those FEC packets are contained within a Redundancy Encoding packet,

there may be not just one payload type value for the VBD stream, but three: one used in the RTP header to signify that the packet contains a redundantly encoded payload, one for the primary payload (the VBD data), and one for the FEC data carried as the secondary encoding.

To optionally protect ~~the~~ a VBD stream, an endpoint may utilize forward error correction and/or redundancy encoding. A stream that utilizes forward error correction shall be signalled via the **fec** field of the **DataType** structure within the **MultiplePayloadStreamElement** structure. A stream that utilizes redundancy encoding shall be signalled via the **redundancyEncoding** field in the **DataType** structure within the **MultiplePayloadStreamElement** structure.

---

*[End Correction]*

## 6 Implementation Clarifications

### 6.1 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.

### 6.2 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.

### 6.3 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).

#### **6.4 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.

#### **6.5 Clarification in Q.931 (1993)**

Table 4-3/Q.931 (1993) (Information Element Identifier Coding) shows that the Progress Indicator IE identifier is 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.

#### **6.6 Graceful Closure of TCP Connections**

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

#### **6.7 Race Condition on Simultaneous Close of Channels**

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 shall issue a Release Complete simultaneously. Endpoints should be prepared for this potential race condition.

#### **6.8 Acceptance of Fast Connect**

When an endpoint accepts the Fast Connect procedure, it may select from the proposed channels as specified in section 8.1.7.1/H.323. The Recommendation clearly specifies what fields shall be modified by the endpoint to accept both the forward and the reverse channels. An endpoint shall not modify any fields other than those specified in 8.1.7.1/H.323 when returning the proposed channels.

Newer versions of H.245 may introduce new fields into the **OpenLogicalChannel** sequence or one of the structures contained therein, as well as new procedures. An older endpoint is obviously not required to decode such new fields or to return such new fields when accepting any proposal. Implementers should consider the consequences of transmitting a newer H.245 OLC to an older endpoint. For the purposes of Fast Connect, the calling endpoint shall assume that the called endpoint's version of H.245 is the minimum version of H.245 necessary to be complaint with an H.323 device that advertises the version of H.225.0 transmitted in the messages from the called endpoint (refer to the "Summary" section of H.323).

#### **6.9 Semantic Differences between Lightweight RRQs and IRQ/IRR Messages**

The lightweight RRQ and the IRR message serve two different functions with an H.323 system. While both are a means of allowing the Gatekeeper to discover that an endpoint is alive, they also each serve separate, unique functions.

The lightweight RRQ is intended to prevent a registration with a Gatekeeper from expiring. The message is generated by the endpoint and does not require the Gatekeeper to poll each endpoint on a regular interval. This message is also a means of allowing the Gatekeeper to provide updated registration information, such as a new list of Alternate Gatekeepers, after the initial registration.

Version 1 of H.323 did not have the concept of a lightweight RRQ, so the IRQ/IRR exchange is the only mechanism available to determine endpoint status of Version 1 devices. However, the lightweight RRQ may be a better choice for determining endpoint status for Version 2 and higher devices.

The IRQ/IRR exchange allows the Gatekeeper to poll the endpoint periodically to discover if the endpoint is still alive. However, an IRR is also intended to convey details about current active calls. This can be used by the Gatekeeper to discover calls that have terminated, which may happen if the endpoint fails to properly send a DRQ message for a call. The IRR message also provides specific details about active calls.

## 6.10 Specifying the Payload Format for a Channel

Implementers should be conscientious of the fact that there are possibly multiple payload formats defined for media formats. For example, two payload formats are defined for H.263—one is defined for the Recommendation H.263 (1996) and one for Recommendation H.263 (1998). Other payload formats may be defined for existing codecs or revisions of those codecs. For interoperability, it is strongly advised that implementers provide the **mediaPacketization** element of the **h2250LogicalChannelParameters** sequence in the **OpenLogicalChannel** message so that there is no ambiguity as to which payload format is being used.

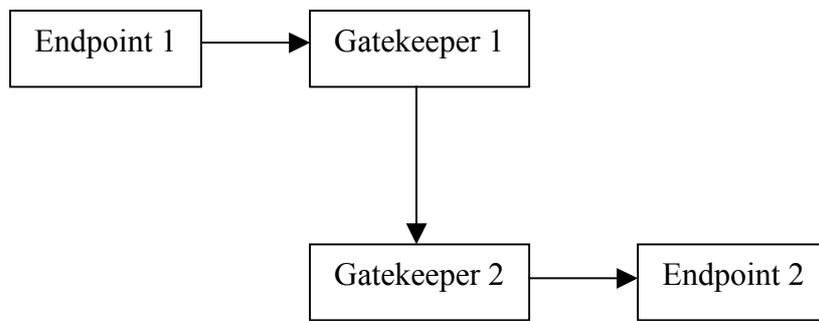
## 6.11 Version Dependencies in Annexes

It was noted that the Annexes to H.323 often fail to indicate the minimum version of H.323 and H.245 required for the Annex. This table is an attempt to clarify the version relationships:

<i>H.323 Annex</i>	<i>Minimum H.323 Version</i>	<i>Minimum H.245 Version</i>
<i>Annex Dv1 (1998)</i>	1998 (Version 2)	1998 (Version 4)
<i>Annex Dv2 (2000)</i>	2000 (Version 4)	2000 (Version 7)
<i>Annex E</i>	1998 (Version 2)	N/A
<i>Annex F</i>	1998 (Version 2)	N/A
<i>Annex G</i>	1998 (Version 2)	1998 (Version 4)
<i>Annex J</i>	1998 (Version 2)	N/A
<i>Annex M.1</i>	2000 (Version 4)	N/A
<i>Annex M.2</i>	2000 (Version 4)	N/A
<i>Annex P</i>	2000 (Version 4)	2003 (Version 9)
<i>Annex R</i>	2000 (Version 4)	N/A

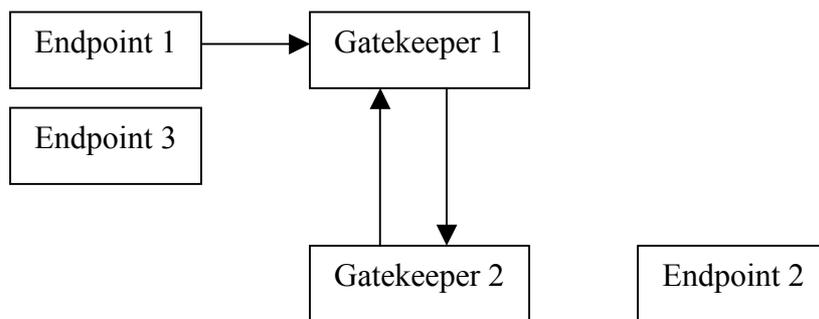
## 6.12 Routing through Signaling Entities and Detecting Loops

In some call scenarios, a call may be routed through a signaling entity multiple times. For example, a call from Endpoint 1 (EP1) may be routed through Gatekeeper 1 (GK1) and Gatekeeper 2 (GK2) to Endpoint 2 (EP2) as shown in Figure 1.



**Figure 1 - Call placed through multiple gatekeepers**

If EP2 redirects the call to a third endpoint, such as Endpoint 3 (EP3), signaling entities such as GK1 and GK2 should be prepared to handle such call rerouting. For this example, assume that EP2 returned a Facility message with a **reason** of **callForwarded** upon receiving a Setup message. Rather than propagate that response back to EP1, GK2 may choose to handle the call forward operation. GK2 would send a Release Complete to EP2 and begin rerouting the call. Suppose that GK2 sends an LRQ message to GK1 for EP3 and that GK1 replies with its address so that that calls routed to EP3 are routed through it. GK2 would then send a Setup message for this call to GK1 as shown in Figure 2.



**Figure 2 - Gatekeeper 2 re-routes call back to Gatekeeper 1**

When GK1 receives the Setup message from GK2, it may inadvertently mistake the call as "bogus", since the Call Identifier will match an already existing call within the Gatekeeper. Implementers should consider this type of call scenario and be prepared to receive incoming calls that contain Call Identifiers for calls that are already being routed through the routing entity. The routing entity should examine not only the Call Identifier, but also the destination address of the call (the call signaling address, aliases, or Called Party Number of the destination). In this case, the call is routed through GK1 with a destination address of EP2 is rerouted by GK2 to GK1, but with a destination address of EP3. In this way, the GK1 will properly handle call routing and rerouting, as well as prevent loops in the call signaling path.

In this example, there was a dependency on the H.323v2 Call Identifier. Unfortunately, H.323 version 1 systems did not have Call Identifiers. For this reason, these loop detection and rerouting procedures are not possible. Nonetheless, it is advisable for routing entities to make an effort to prevent loops properly. For example, if the entities in Figure 2 were version 1 devices, the GK1 may examine the source address, destination address, and Conference Identifier (CID) of the call. The first time the call is presented to the Gatekeeper, the destination address is EP2, just as before. However, when GK re-routes the call back to GK1, the destination address is EP3. In this way, GK1 may allow proper rerouting of the call to EP3.

The logic for Version 1 devices seems similar to that for Version 2 and higher devices, but there are issues when EP2 and EP3 are MCUs, for example. Suppose that EP2 is an MCU that is directing

all calls to EP3. The first time a call is redirected to GK1, GK1 may realize that this is, indeed, a call redirection as described above. However, when the second call is redirected, GK1 has no means of distinguishing between the first redirected call and the second: the source address *may* be the same, the destination address is the same as the previously rerouted call (EP3), and the Conference ID is the same. So in this case, GK1 may have no choice but to assume that a loop has occurred and release the offending call. Although this is unfortunate, H.323v2 and higher systems do not suffer from this problem. What is important, though, is that loop detection is possible—even with version 1 systems.

### 6.13 Packetization for G.729, G.729a, G.711, and G.723.1

The delay associated with codec processing and packetization should be kept as short as possible. To accomplish this objective when G.729 or G.729A is used, two frames per packet should be considered as the maximum packet size. Similarly, G.711 may be used with packet sizes of 10 ms (80 frames) or 20 ms (160 frames) to achieve this objective. Finally, when G.723.1 is used, only one frame should be included in each packet. The 30 ms frame size of G.723.1 results in speech collection and coding delay of at least 60 ms, contributing to difficulty of interactive communications.

### 6.14 Checking versions for T.38 and V.150.1

It is important that devices properly negotiate the version of the T.38 or V.150.1 to be used and agree to use the same version. At the present time there are few guidelines for version negotiation. Until the guidelines are developed the following note applies:

Devices supporting multiple versions of T.38 and V.150.1 may offer multiple proposals in Fast Connect, each with a different version specified. A device shall not accept a proposal for a version that it does not support.

## 7 Allocated Object Identifiers and Port Numbers

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

### 7.1 Allocated Object Identifiers

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

Note that object IDs below that are allocated below the arc { itu-t(0) recommendation(0) } are show with an abbreviated prefix of "0 0" below.

{ 0 0 h(8) 2250 version(0) [v] }	H225.0 version numbers
Assigned values of v: 1-4	
{ 0 0 h(8) 2250 annex(1) g(7) version(0) [v] }	H225.0 Annex G version numbers
Assigned values of v: 1-2	
{ 0 0 h(8) 2250 annex(1) g(7) usage(1) [u] }	H225.0 Annex G usage tags
Assigned values of u: none	
{ 0 0 h(8) 245 version(0) [v] }	H245 version numbers

Assigned values of <i>v</i> : Please refer to Table D.1/H.245	
{ 0 0 h(8) 245 generic-capabilities(1) video(0) [ <i>c</i> ] }	Generic video capabilities
Assigned values of <i>c</i> : Please refer to Table D.1/H.245	
{ 0 0 h(8) 245 generic-capabilities(1) audio(1) [ <i>c</i> ] }	Generic audio capabilities
Assigned values of <i>c</i> : Please refer to Table D.1/H.245	
{ 0 0 h(8) 245 generic-capabilities(1) data(2) [ <i>c</i> ] }	Generic data capabilities
Assigned values of <i>c</i> : Please refer to Table D.1/H.245	
{ 0 0 h(8) 245 generic-capabilities(1) control(3) [ <i>c</i> ] }	Generic control capabilities
Assigned values of <i>c</i> : Please refer to Table D.1/H.245	
{ 0 0 h(8) 245 generic-capabilities(1) multiplex(4) [ <i>c</i> ] }	Generic multiplex capabilities
Assigned values of <i>c</i> : Please refer to Table D.1/H.245	
{ 0 0 h(8) 283 generic-capabilities(1) 0 }	H.283 Capability
{ iso (1) identified-organization (3) icd-ecma (0012) private-isdn-signalling-domain (9) }	Identifies QSIG as the tunneled protocol within an H.225.0 Call Signalling Channel

<b>7.2 Allocated Port Numbers</b>
-----------------------------------

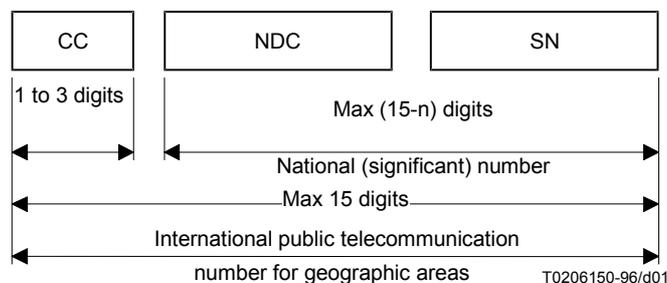
The following IP port numbers have been allocated for various components of H.323:

- 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 Signalling

## 8 Use of E.164 and ISO/IEC 11571 Numbering Plans

<b>8.1 E.164 numbering plan</b>
---------------------------------

ITU-T Recommendation defines E.164 numbers the following way for geographic areas:



CC Country Code for geographic areas  
 NDC National Destination Code (optional)  
 SN Subscriber Number  
 n Number of digits in the country code

NOTE – National and international prefixes are not part of the international public telecommunication number for geographic areas.

### Figure – International public telecommunication number structure for geographic areas

Similar descriptions are also defined for non-geographic areas. Recommendation E.164 further defines country codes (CC) for all the countries and regions of the world.

An international E.164 number always starts with a country code and its total length is always 15 digits or less. More importantly, it does not include any prefixes that are part of a dialing plan (for example, "011" for an international call placed in North America, or "1" for a long-distance call), nor does it include "#" or "\*". The number "49 30 345 67 00" is an E.164 number with CC=49 for Germany. A national number is the international number stripped of the country code, "30 345 67 00" in this case. The subscriber number is the national number stripped of the national destination code, "345 67 00" in this case.

An E.164 number has global significance: any E.164 number can be reached from any location in the world. A "dialed digit sequence", however, only has significance within a specific domain.

Within a typical private numbering plan in an enterprise, for example, a prefix, such as "9", may indicate that a call goes "outside", at which point the local telephone company's dialing plan takes over. Each telephone company or private network is free to choose its own dialing plan. It is also free to change it as it pleases—and frequently does so (adding new area codes, for example).

In a typical geographically determined network where users input telephone numbers manually and where users do not travel too much, having different dialing plans everywhere is usually a problem. However, when a user travels, the user must determine the other network's numbering plan in order to place calls. When computer systems perform the dialing automatically, the user is usually required to customize the dialing software for every region or network.

Because of these issues with varying dialing plans and automated dialing, it is essential to be able to refer to an absolute "telephone number" instead of "what you have to dial to reach it from a specific location." Proper usage of E.164 numbers can resolve these issues. Many systems use E.164 numbers instead of dialed digits: for example, a PBX may gather the dialed digits from a user on a telephone and then initiate a call to the local phone company using an E.164 number in the Called Party Number information element in Q.931. When completing the Called Party Number IE, specifying the numbering plan as "ISDN/telephony numbering plan (Recommendation E.164)" indicates an E.164 number. Specifying the type of number as "unknown" and the specifying the numbering plan as "unknown" indicates dialed digits.

The following are a set of definitions from E.164:

#### number

A string of decimal digits that uniquely indicates the public network termination point. The number contains the information necessary to route the call to this termination point.

A number can be in a format determined nationally or in an international format. The international format is known as the International Public Telecommunication Number which includes the country code and subsequent digits, but not the international prefix.

### **numbering plan**

A numbering plan specifies the format and structure of the numbers used within that plan. It typically consists of decimal digits segmented into groups in order to identify specific elements used for identification, routing and charging capabilities, e.g. within E.164 to identify countries, national destinations, and subscribers.

A numbering plan does not include prefixes, suffixes, and additional information required to complete a call.

The national numbering plan is the national implementation of the E.164 numbering plan.

### **dialing plan**

A string or combination of decimal digits, symbols, and additional information that define the method by which the numbering plan is used. A dialing plan includes the use of prefixes, suffixes, and additional information, supplemental to the numbering plan, required to complete the call.

### **address**

A string or combination of decimal digits, symbols, and additional information which identifies the specific termination point(s) of a connection in a public network(s) or, where applicable, in interconnected private network(s).

### **prefix**

A prefix is an indicator consisting of one or more digits, that allows the selection of different types of number formats, networks and/or service.

### **international prefix**

A digit or combination of digits used to indicate that the number following is an International Public Telecommunication Number.

### **country code (CC) for geographic areas**

The combination of one, two or three digits identifying a specific country, countries in an integrated numbering plan, or a specific geographic area.

### **national (significant) number [N(S)N]**

That portion of the number that follows the country code for geographic areas. The national (significant) number consists of the National Destination Code (NDC) followed by the Subscriber Number (SN). The function and format of the N(S)N is nationally determined.

### **national destination code (NDC)**

A nationally optional code field, within the E.164 number plan, which combined with the Subscriber's Number (SN) will constitute the national (significant) number of the international public telecommunication number for geographic areas. The NDC will have a network and/or trunk code selection function.

The NDC can be a decimal digit or a combination of decimal digits (not including any prefix) identifying a numbering area within a country (or group of countries included in one integrated numbering plan or a specific geographic area) and/or network/services.

### **national (trunk) prefix**

A digit or combination of digits used by a calling subscriber, making a call to a subscriber in his own country but outside his own numbering area. It provides access to the automatic outgoing trunk equipment.

### **subscriber number (SN)**

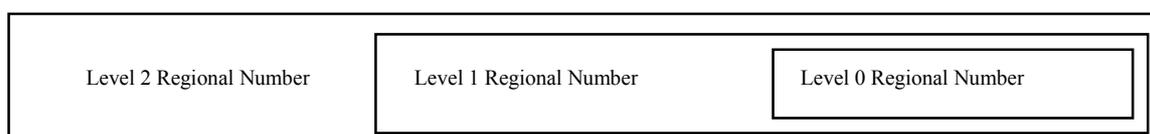
The number identifying a subscriber in a network or numbering area.

## **8.2 Private Network Number**

Private Network Numbers are used in private or virtual private telephony networks, e.g., a corporate network of PBXs and virtual private lines.

ISO/IEC 11571 defines Private Network Number (PNP) as having up to three regional levels.

A PNP Number shall comprise a sequence of x decimal digits (0,1,2,3,4,5,6,7,8,9) with the possibility that different PNP Numbers within the same PNP can have different values of x. The maximum value of x shall be the same as for the public ISDN numbering plan, see ITU-T Recommendation E.164.



**Figure – H.323 - Structure of a PNP Number with three levels of regions**

A level n Regional Number (RN) shall have significance only within the level n region to which it applies. When that number is used outside that level n region, it shall be in the form of an RN of level greater than n. Only a Complete Number shall have significance throughout the entire PNP.

A typical example in North America would be a 4-digit "extension" as the Level 0 Regional Number: a 3-digit "location code" combined with the 4 digit "extension" would form the Level 1 Regional Number. The Level 2 Regional Number would be nil.

A prefix could also be used to signal which regional number is used, and would not be part of the regional number per se, but only part of the dialing plan. Again, a typical example would be the use of digit "6" to access a Level 1 Regional Number, and no digit for a Level 0 Regional Number.

The following are a set of definitions from ISO/IEC 11571:

### **Private Numbering Plan (PNP)**

The numbering plan explicitly relating to a particular private numbering domain, defined by the PISN Administrator of that domain.

### **PNP Number**

A number belonging to a PNP.

### **Region**

The entire domain or a sub-domain of a PNP. A region does not necessarily correspond to a geographical area of a PISN.

### **Region Code (RC)**

The leading digits of a PNP Number which identify a region. The RC may be omitted to yield a shortened form of a PNP Number for use internally to that region.

### **Regional Number (RN)**

A particular form of a PNP Number which is unambiguous in the region concerned.

## Complete Number

A number which is unambiguous in the entire PNP, i.e. which corresponds to the highest regional level employed in that PISN.

## 9 ASN.1 Usage, Guidelines, and Conventions

### 9.1 NULL, BOOLEAN, and NULL/BOOLEAN OPTIONAL

Throughout the ASN.1 used in H.323-series documents, the reader will see the types NULL and BOOLEAN used, along with the modifier OPTIONAL in some cases. People have questioned when NULL should be used or when BOOLEAN should be used and what the semantic differences are.

The BOOLEAN type allows a TRUE or FALSE value to be conveyed in the protocol. When used in conjunction with OPTIONAL, it actually allows three values to be conveyed through the protocol: TRUE, FALSE, and *absent*. The question is what does *absent* mean? In some instances, the absence of a BOOLEAN OPTIONAL means should be interpreted as FALSE, while in other cases, it should be interpreted as "I don't care" or "I don't know"—but not always. For example, the **additiveRegistration** field in the RRQ of H.225.0 Version 4 is defined as a BOOLEAN OPTIONAL. When present, it clearly indicates that the endpoint supports the feature or does not support the feature. However, absence of this field shall also be interpreted as FALSE. The reason is that an older endpoint would not know anything about the field and would obviously not be able to include it. Moreover, they certainly do not support the feature. Another example is the **originator** field in the **perCallInfo** sequence. When present, the meaning is quite clear: the caller is the originator or the terminator of the call. However, if the field is not present, it may mean that the endpoint does not know or cannot supply this information for some reason.

The NULL type is often used to select one of several CHOICE options. NULL carries no particular value, as it merely indicates presence. In selecting the conference goal in a Setup message, for example, the goal CHOICEes are simply NULL types to allow the endpoint to indicate a selection. Another common use of NULL is with the OPTIONAL modifier. A NULL OPTIONAL type allows an endpoint to indicate support for a feature, for example. It is similar in semantics to a BOOLEAN in that the presence of a NULL field indicates TRUE and absence of the NULL field indicates a FALSE. As an example, the **fastConnectRefused** field in the Alerting message is a NULL OPTIONAL. Absence of the field is interpreted as FALSE—Fast Connect is not (yet) refused. Presence of the field, though, clearly indicates refusal of Fast Connect. So why was BOOLEAN not used as the type for this field? It would not have made the encoding any clearer, because the field is past the extension marker (ellipsis). A version 1 and 2 device, for example, would not know to send this field, so there would be three values to consider if BOOLEAN were used: TRUE, FALSE, and *absent*.

Ideally, a field will convey no more values than makes sense. In most cases, these types indicate only two possible values: TRUE/present or FALSE/absent. However, there may be cases where three values are intended and the reader should refer to the appropriate Recommendation to determine if, indeed, there is significance in tri-state fields.

### 9.2 ASN.1 Usage in H.450-Series Recommendations

This section summarizes the use of ASN.1 in the current H.450.x recommendations. This information is provided for implementers of the H.450.x protocols, as well as authors of new H.450.x Recommendations.

### 9.2.1 ASN.1 version and encoding rules

The ASN.1 code in H.450.x is based on the 1994 version of X.680-683, including the amendments on “*Rules of extensibility*”.

The *basic aligned variant of packed encoding rules* (PER) is used as specified in X.691 (1995).

### 9.2.2 Tagging

All modules defined in Recommendations H.450.x use the *tag default* AUTOMATIC TAGS.

The ROS APDUs (see below) are defined in H.450.1 as *tagged types* within the CHOICE type ROS. No other type defined in H.450.x is a *tagged type*, i.e. all *sets*, *sequences* and *choices* (except ROS) are automatically tagged.

### 9.2.3 Basic ASN.1 Types

The following types occur in ASN.1 definitions of H.450.x:

BMPString, NumericString	NULL
BOOLEAN	OBJECT IDENTIFIER
CHOICE	OCTET STRING
<i>CLASS (see below)</i>	<i>Open type (see below)</i>
ENUMERATED	SEQUENCE
GeneralizedTime	SEQUENCE OF
INTEGER	SET OF

No use is currently foreseen for the following basic types (needs consideration on a case-by-case basis):

CHARACTER STRING	ObjectDescriptor
EMBEDDED PDV	REAL
EXTERNAL	UTCTime
GeneralString, GraphicString, PrintableString, TeletexString (T61String), UniversalString, VideotexString, VisibleString (ISO646String)	

Use of the following basic types in future recommendations H.450.x should not be precluded (needs consideration on a case-by-case basis):

BIT STRING	Selection Type (out of a CHOICE)
IA5String	SET
INSTANCE OF	TYPE-IDENTIFIER (see X.681)

Note: Some of these types are already used by other recommendations in the H.323 universe, e.g. BIT STRING and TYPE-IDENTIFIER in H.235.

## 9.2.4 Value sets, subtyping and constraints used in H.450.x:

H.450.x recommendations use *size constraints* (strings, set-of and sequence-of) and *value range constraints* (integers). In H.450.1 *inner subtyping* (“WITH COMPONENTS”) is used occasionally.

The use of *value sets*, *single values*, *contained subtypes* and *permitted alphabets* should be possible if needed by future services. The *type constraint* (for restricting an *open type*) may be useful, too.

Explicit set arithmetic (UNION, INTERSECTION, EXCEPT, ALL EXCEPT) is currently not used on subtype specifications.

## 9.2.5 Object classes, parameterization, general constraints, and ROS

H.450.1 defines a *remote operations service* (ROS) based on X.880. ROS uses *object classes* (X.681), *parameterization* (X.683) and *constraints* (X.682) for its generic part.

Two object classes OPERATION and ERROR are defined and then used to define four PDU types (*Invoke*, *ReturnResult*, *ReturnError* and *Reject*) as sequences containing individual parts of these classes. The first three PDU types contain an optional *open type* component which is tied by a *table constraint* (“at (@)”) notation) to the code value identifying the particular operation or error.

For each supplementary service the actual operations and errors are then defined as *object instances* of the generic classes OPERATION and ERROR in the corresponding Rec. H.450.x. Each operation and error is identified uniquely (within the context of the H.450.x series) by a code value (type INTEGER). A list of currently assigned operation and error values is contained in section 10.8 below.

Each supplementary service defines an *object set* containing all operations defined for that service.

## 9.2.6 Extensibility and non-standard information

Wherever meaningful, an *extension marker* (ellipsis “...”) is included in the definitions.

All operations, and some errors, include placeholders for non-standard (e.g. manufacturer-specific) information. This non-standard information can either be of type *NonStandardParameter* (imported from H.225.0) or of type *Extension*, which is defined in H.450.1 and consists of an *object identifier* followed by an *open type*. The definition of the Extension type uses an *object class* (EXTENSION) with *parameterization* and *constraints* similar to the ROS definition.

Usually there is space for more than one addition of non-standard information in an operation. Additions of both types (NonStandardParameter and Extension) can be mixed in any order.

## 9.2.7 List of Operation and Error Codes

Table 10.1: ASN.1 Operation values used in H.450 series

Value number	Value name	Defined in standard:
0	callingName	H.450.8
1	calledAlertingName	H.450.8
2	connectedName	H.450.8
3	busyName	H.450.8
7	callTransferIdentify	H.450.2
8	callTransferAbandon	H.450.2
9	callTransferInitiate	H.450.2

10	callTransferSetup	H.450.2
11	callTransferActive	H.450.2
12	callTransferComplete	H.450.2
13	callTransferUpdate	H.450.2
14	subaddressTransfer	H.450.2
15	activateDiversionQ	H.450.3
16	deactivateDiversionQ	H.450.3
17	interrogateDiversionQ	H.450.3
18	checkRestriction	H.450.3
19	callRerouting	H.450.3
20	divertingLegInformation1	H.450.3
21	divertingLegInformation2	H.450.3
22	divertingLegInformation3	H.450.3
23	cfnrDivertedLegFailed	H.450.3
27	ccnrRequest	<del>Draft</del> H.450.9
28	ccCancel	<del>Draft</del> H.450.9
29	ccExecPossible	<del>Draft</del> H.450.9
31	ccRingout	<del>Draft</del> H.450.9
32	ccSuspend	<del>Draft</del> H.450.9
33	ccResume	<del>Draft</del> H.450.9
34	<u>callOfferRequest</u>	<u>H.450.10</u>
40	ccbsRequest	<del>Draft</del> H.450.9
43	<u>callIntrusionRequest</u>	<u>H.450.11</u>
44	<u>callIntrusionGetCIPL</u>	<u>H.450.11</u>
45	<u>callIntrusionIsolate</u>	<u>H.450.11</u>
46	<u>callIntrusionForcedRelease</u>	<u>H.450.11</u>
47	<u>callIntrusionWOBRequest</u>	<u>H.450.11</u>
49	<u>cfbOverride</u>	<u>H.450.10</u> (re-used in H.450.11)
80	mwiActivate	H.450.7
81	mwiDeactivate	H.450.7
82	mwiInterrogate	H.450.7
84	<u>cmnRequest</u>	<u>H.450.12</u>
85	<u>cmnInform</u>	<u>H.450.12</u>
100	divertingLegInformation4	H.450.3
101	holdNotific	H.450.4

102	retrieveNotific	H.450.4
103	remoteHold	H.450.4
104	remoteRetrieve	H.450.4
105	callWaiting	H.450.6 (re-used in H.450.10, H.450.11)
106	cpRequest	H.450.5
107	cpSetup	H.450.5
108	groupIndicationOn	H.450.5
109	groupIndicationOff	H.450.5
110	pickrequ	H.450.5
111	pickup	H.450.5
112	pickExe	H.450.5
113	cpNotify	H.450.5
114	cpickupNotify	H.450.5
115	remoteUserAlerting	H.450.10 (re-used in H.450.11)
116	callIntrusionSilentMonitor	H.450.11
117	callIntrusionNotification	H.450.11

**Table 10.2: ASN.1 Error Values used in H.450 series**

<b>Value number</b>	<b>Value name</b>	<b>Defined in standard:</b>
0	userNotSubscribed	H.450.1
1	rejectedByNetwork	H.450.1
2	rejectedByUser	H.450.1
3	notAvailable	H.450.1
5	insufficientInformation	H.450.1
6	invalidServedUserNumber	H.450.1
7	invalidCallState	H.450.1
8	basicServiceNotProvided	H.450.1
9	notIncomingCall	H.450.1
10	supplementaryServiceInteractionNotAllowed	H.450.1
11	resourceUnavailable	H.450.1
12	invalidDivertedNumber	H.450.3
14	specialServiceNumber	H.450.3
15	diversionToServedUserNumber	H.450.3
24	numberOfDiversionsExceeded	H.450.3

25	callFailure	H.450.1
31	notActivated	H.450.7
43	proceduralError	H.450.1
1000	temporarilyUnavailable	H.450.3, <u>H.450.11</u>
1004	invalidReroutingNumber	H.450.2
1005	unrecognizedCallIdentity	H.450.2
1006	establishmentFailure	H.450.2
1007	notAuthorized	H.450.3, <u>H.450.11</u>
1008	unspecified	H.450.2, H.450.3
<u>1009</u>	<u>notBusy</u>	<u>H.450.11</u>
1010	shortTermRejection	<del>Draft</del> H.450.9
1011	longTermRejection	<del>Draft</del> H.450.9
1012	remoteUserBusyAgain	<del>Draft</del> H.450.9
1013	failureToMatch	<del>Draft</del> H.450.9
1018	invalidMsgCentreId	H.450.7
2000	callPickupIdUnvalid	H.450.5
2001	callAlreadyPickedUp	H.450.5
2002	undefined	H.450.4, H.450.5, H.450.7, (re-used in H.450.9, <u>H.450.11</u> , <u>H.450.12</u> )

**Annex: H.323 Recommendation System Defect Report Form**

<b>DATE:</b>	
<b>CONTACT INFORMATION</b>  <b>NAME:</b> <b>COMPANY:</b> <b>ADDRESS:</b>  <b>TEL:</b> <b>FAX:</b> <b>EMAIL:</b>	
<b>AFFECTED RECOMMENDATIONS:</b>	
<b>DESCRIPTION OF PROBLEM:</b>	
<b>SUGGESTIONS FOR RESOLUTION:</b>	

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

\_\_\_\_\_