



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T H.248 Sub-series Implementors' Guide

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(24 November 2006)

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

**Implementors' Guide for the H.248 Sub-series of
Recommendations (Media Gateway Control
Protocol)**

Summary

This document is a compilation of reported defects identified in the ITU-T H.248 sub-series of 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.248 sub-series Recommendations.

This revision contains all updates submitted up to and including those at Study Group 16 meeting in November 2006.

This document was approved by ITU-T Study Group 16 on 24 November 2006 and obsoletes the earlier version of this Implementors' Guide approved on 13 April 2006. Please note that the Implementors' Guides for H.248.1 Version 1 and Version 2 are published as *separate* documents.

Change Log

(All changes that were included in corrigenda, amendments or revisions to the recommendations in the H.248 subseries are omitted here.)

V19 (Melbourne, February 2005)

Added new section for H.248.20, renumbering other sections as appropriate.

New:

- 6.12 Annex C and SDP parameters
- 6.13 Case Sensitivity of Profile Names
- 6.14 Profile Negotiation
- 6.15 Conflict between H.248.1 Version 2 Corrigendum 1 and H.248.8
- 7.1 Clarification of “At-Most-Once” Functionality
- 9.1 Media Values

V20 (Geneva, August 2005)

New:

- 6.16 AuditCapability of Signals
- 6.17 Media Type Mismatch
- 6.18 Notify Avalanche
- 6.19 Topology Reply
- 6.20 Statistics and Sub-lists
- 7.1 Probe Order Typo
- 9.2 New Error Code – Too many transactions
- 12.1 Alerting Confusion
- 13.1 Metering Pulses at Signal Replacement

V21 (Geneva, November 2005)

Removed items pertaining to H.248.1 Version 2, as they are incorporated into the new H.248.1 Version 2 IG. Added new section for H.248.1 Version 3. Removed existing items pertaining to H.248.8, as they are incorporated in the H.248.8 (09/2005) revision. Added new section for H.248.9. Renumbered existing sections.

New:

- 6.1 Clarification of ASN.1 definition of topologyDirectionExtension
- 6.2 Correction of ASN.1
- 6.3 Correction of Annex E.14.6 Procedures
- 6.4 Clarification of statistic reset capability
- 6.5 Clarification of delay calculation
- 6.6 Protocol version negotiation
- 9.1 Error text for error code 449

- 10.1 Clarification on terminating PlayRecord successfully via MGC command
- 10.2 Correction in type-ahead handling

V22 (Geneva, April 2006)

Added new section for H.248.14. Renumbered existing sections.

New:

- 6.7 Clarification of error code usage in wildcarding procedures
- 6.8 ServiceStates clarification for continuity testing
- 6.9 Reference to location of ServiceChangeMgcID definition
- 6.10 Clarification of termination service state upon restart of MG
- 6.11 Alignment of text among events in the Tone Detection Package
- 6.12 Clarification of package definition requirements for enumerations
- 6.13 Clarification on Profile Definition Template
- 6.14 Clarification of use of ABNF encodings of octet strings
- 6.15 Clarification of encoding for packet loss statistic in Annex E.12
- 6.16 Missing ServiceChange parameter from Appendix III
- 6.17 Clarification of ServiceChangeMethod Graceful behavior on ephemeral terminations
- 7.2 Reference Update
- 9.2 Protocol error on command level
- 9.3 New error code 511
- 11.1 Provisioning of the Inactivity Timeout Event

V23 (Ottawa, August 2006)

Added new section for H.248.17. Renumbered existing sections.

Modified:

- 6.15 Clarification of encoding for packet loss statistic in Annex E.12

New:

- 6.18 Clarification of package versions versus protocol versions
- 6.19 Specification of wildcarded response usecases in profile template
- 6.20 Specification of termination and stream type support for packages in profile template
- 6.21 Addition of SDP procedural section to profile template
- 6.22 Correction to profile negotiation in Appendix I
- 6.23 Termination ID in Add Reply with error
- 6.24 TerminationState Descriptor in the profile template
- 6.25 Root and non-Root Terminations in a TerminationIDList
- 6.26 Codec selection
- 6.27 Profile negotiation and control association

- 6.28 Clarification of use of ServiceChangeAddress
- 6.29 Correction of typographical error in Appendix III clause 5.7.4
- 6.30 Clarification of length encoding in ASN.1 syntax
- 6.31 Clarification for context attribute values when omitted from actions
- 6.32 Clarification of the use of ServiceChange Disconnected when re-establishing comms
- 6.33 Clarification of use of unsigned integer
- 9.4 Clarification of the use of error text
- 12.1 Correction of Typographical error in Clause 8.3.1.2/H.248.17

V24 (Geneva, November 2006)

Added new section for H.248.23. Removed existing items pertaining to H.248.2, as they are incorporated in the H.248.2 (01/2005) Amendment 1 (01/2007). Removed existing items pertaining to H.248.30, as they are incorporated in the H.248.30 (01/2007) revision. Renumbered existing sections.

Modified:

- 6.15 Clarification of encoding for packet loss statistic in Annex E.12

New:

- 6.34 Correction to SDP examples in Appendix I
- 6.35 Clarification of SDP requirements for the MG
- 6.36 Correction for Segmentation Error handling
- 6.37 Clarification of the Octets Sent and Octets Received Statistics in the nt and rtp packages
- 14.1 Clarification of default pattern ID
- 14.2 Clarification of solutions to possible race condition in dwa signal

Contact Information

ITU-T Study Group 16 / Question 3 Rapporteur	Christian Groves Australia	Tel: +61 3 9301 6116 E-mail: Christian.Groves@nteczone.com
H.248 Sub-series Implementors' Guide Editor	Kevin Boyle II USA	Tel: +1 919 991 2690 E-mail: kboyle@nortel.com

Table of Contents

1	SCOPE.....	1
2	INTRODUCTION	1
3	DEFECT RESOLUTION PROCEDURE.....	1
4	REFERENCES	2
5	NOMENCLATURE	2
6	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.1 (09/2005).....	3
6.1	CLARIFICATION OF ASN.1 DEFINITION OF TOPOLOGYDIRECTIONEXTENSION	3
6.2	CORRECTION OF ASN.1	4
6.3	CORRECTION OF ANNEX E.14.6 PROCEDURES	4
6.4	CLARIFICATION OF STATISTIC RESET CAPABILITY	5
6.5	CLARIFICATION OF DELAY CALCULATION.....	6
6.6	PROTOCOL VERSION NEGOTIATION	7
6.7	CLARIFICATION OF ERROR CODE USAGE IN WILDCARDING PROCEDURES	7
6.8	SERVICESTATES CLARIFICATION FOR CONTINUITY TESTING	8
6.9	REFERENCE TO LOCATION OF SERVICECHANGEMgcID DEFINITION.....	8
6.10	CLARIFICATION OF TERMINATION SERVICE STATE UPON RESTART OF MG.....	9
6.11	ALIGNMENT OF TEXT AMONG EVENTS IN THE TONE DETECTION PACKAGE	10
6.12	CLARIFICATION OF PACKAGE DEFINITION REQUIREMENTS FOR ENUMERATIONS	11
6.13	CLARIFICATION ON PROFILE DEFINITION TEMPLATE.....	13
6.14	CLARIFICATION OF USE OF ABNF ENCODINGS OF OCTET STRINGS	13
6.15	CLARIFICATION OF ENCODING FOR PACKET LOSS STATISTIC IN ANNEX E.12	14
6.16	MISSING SERVICECHANGE PARAMETER FROM APPENDIX III.....	15
6.17	CLARIFICATION OF SERVICECHANGEMETHOD GRACEFUL BEHAVIOR ON EPHEMERAL TERMINATIONS.....	16
6.18	CLARIFICATION OF PACKAGE VERSIONS VERSUS PROTOCOL VERSIONS.....	16
6.19	SPECIFICATION OF WILDCARDED RESPONSE USECASES IN PROFILE TEMPLATE.....	18
6.20	SPECIFICATION OF TERMINATION AND STREAM TYPE SUPPORT FOR PACKAGES IN PROFILE TEMPLATE.....	19
6.21	ADDITION OF SDP PROCEDURAL SECTION TO PROFILE TEMPLATE	21
6.22	CORRECTION TO PROFILE NEGOTIATION IN APPENDIX I	21
6.23	TERMINATION ID IN ADD REPLY WITH ERROR	22
6.24	TERMINATIONSTATE DESCRIPTOR IN THE PROFILE TEMPLATE.....	23
6.25	ROOT AND NON-ROOT TERMINATIONS IN A TERMINATIONIDLIST	24
6.26	CODEC SELECTION	25
6.27	PROFILE NEGOTIATION AND CONTROL ASSOCIATION	25
6.28	CLARIFICATION OF USE OF SERVICECHANGEADDRESS.....	27
6.29	CORRECTION OF TYPOGRAPHICAL ERROR IN APPENDIX III CLAUSE 5.7.4.....	27
6.30	CLARIFICATION OF LENGTH ENCODING IN ASN.1 SYNTAX	28
6.31	CLARIFICATION FOR CONTEXT ATTRIBUTE VALUES WHEN OMITTED FROM ACTIONS	29
6.32	CLARIFICATION OF THE USE OF SERVICECHANGE DISCONNECTED WHEN RE-ESTABLISHING COMMS	30
6.33	CLARIFICATION OF USE OF UNSIGNED INTEGER.....	30
6.34	CORRECTION TO SDP EXAMPLES IN APPENDIX I	32
6.35	CLARIFICATION OF SDP REQUIREMENTS FOR THE MG	35

6.36	CORRECTION FOR SEGMENTATION ERROR HANDLING	36
6.37	CLARIFICATION OF PACKAGE DEFINITION AND REGISTRATION PROCEDURES	37
6.38	CLARIFICATION OF THE OCTETS SENT AND OCTETS RECEIVED STATISTICS IN THE NT AND RTP PACKAGES.....	39
7	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.5 (2000).....	40
7.1	CLARIFICATION OF “AT-MOST-ONCE” FUNCTIONALITY	40
8	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.8 (2005).....	41
8.1	ERROR TEXT FOR ERROR CODE 449.....	41
8.2	PROTOCOL ERROR ON COMMAND LEVEL.....	41
8.3	NEW ERROR CODE 511	42
8.4	CLARIFICATION OF THE USE OF ERROR TEXT.....	42
9	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.9 (2005).....	43
9.1	CLARIFICATION ON TERMINATING PLAYRECORD SUCCESSFULLY VIA MGC COMMAND	43
9.2	CORRECTION IN TYPE-AHEAD HANDLING.....	45
10	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.14 (2002).....	46
10.1	PROVISIONING OF THE INACTIVITY TIMEOUT EVENT.....	46
11	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.17 (2002) CORR. 1 (2004).....	47
11.1	CORRECTION OF TYPOGRAPHICAL ERROR IN CLAUSE 8.3.1.2/H.248.17	47
12	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.20 (2002).....	47
12.1	MEDIA VALUES.....	47
13	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.22 (2003).....	48
13.1	CORRECTION OF TYPOGRAPHICAL ERRORS	48
14	TECHNICAL AND EDITORIAL CORRECTIONS TO H.248.23 (2005) CORR. 1 (2006).....	49
14.1	CLARIFICATION OF DEFAULT PATTERN ID	49
14.2	CLARIFICATION OF SOLUTIONS TO POSSIBLE RACE CONDITION IN DWA SIGNAL	50
	ANNEX: DEFECT REPORT FORM FOR THE H.248 SUB-SERIES.....	51

Implementors' Guide for the H.248 Sub-series of Recommendations

1 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 through contributions to the ITU-T.

2 Introduction

The H.248 Implementors' Guide is a compilation of reported defects for all versions of the H.248.x sub-series of Recommendations, except H.248.1 Version 1 (03/2002) and H.248.1 Version 2 (05/2002) Corrigendum 1 (03/2004). *For the defects in Version 1, see the H.248.1 Version 1 Implementors' Guide. For the defects in Version 2, see the H.248.1 Version 2 Implementors' Guide.*

In this edition of the Guide, reported defects identified as of 11/2006 are given for:

- H.248.1 version 3 (09/2005)
- H.248.5 (11/2000)
- H.248.8 (09/2005)
- H.248.9 (01/2005)
- H.248.14 (03/2002)
- H.248.17 (11/2002) Corrigendum 1 (03/2004)
- H.248.20 (11/2002)
- H.248.22 (07/2003)
- H.248.23 (01/2005) Corrigendum 1 (05/2006)

The Guide must be read in conjunction with the H.248.x sub-series of Recommendations to serve as an additional source of information for implementors. The changes, clarifications and corrections defined herein are expected to be included in future versions of affected H.248.x Recommendations.

3 Defect Resolution Procedure

Upon discovering technical defects with any components of the H.248.x Sub-series Recommendations, please provide a written description directly to the editors of the affected Recommendations with a copy to the Question 3/16 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.

4 References

This document refers to the following H.248.x sub-series Recommendations:

- ITU-T Recommendation H.248.1 Version 3 (09/2005), *Gateway Control Protocol: Version 3*
- ITU-T Recommendation H.248.5 (11/2000), *Gateway Control Protocol: Transport over ATM*
- ITU-T Recommendation H.248.8 (09/2005), *Gateway Control Protocol: Error code and service change reason description*
- ITU-T Recommendation H.248.9 (01/2005), *Gateway control protocol: Advanced media server packages*
- ITU-T Recommendation H.248.14 (03/2002), *Gateway control protocol: Inactivity timer package*
- ITU-T Recommendation H.248.17 (11/2002), Corrigendum 1 (03/2004), *Gateway control protocol: Line test packages*
- ITU-T Recommendation H.248.20 (11/2002), *Gateway Control Protocol: The use of local and remote descriptors with H.221 and H.223 multiplexing*
- ITU-T Recommendation H.248.22 (07/2003), *Gateway Control Protocol: Shared Risk Group Package*
- ITU-T Recommendation H.248.23 (01/2005), Corrigendum 1 (05/2006), *Gateway Control Protocol: Enhanced alerting packages*

5 Nomenclature

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

Symbol	Description
<u><i>[Begin Correction]</i></u>	Identifies the start of revision marked text based on extractions from the published Recommendations affected by the correction being described.
<u><i>[End Correction]</i></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.
--- <i>SPECIAL INSTRUCTIONS</i> --- {instructions}	Indicates a set of special editing instructions to

be followed.

6 Technical and Editorial Corrections to H.248.1 (09/2005)

6.1 Clarification of ASN.1 definition of topologyDirectionExtension

Description:	In H.248.1 v3 Annex A.2 ASN.1 Syntax Definition, the new two topology modes ‘onewayexternal’ and ‘onewayboth’ have been defined in a new Information Element. This was due to the lack of extension capability of the existing Information Element topologyDirection. Since the Information Element topologyDirection is mandatory, in those cases where topology is set to one of the new modes, the receiver of the command will receive two different modes, e.g. Oneway and OnewayBoth. It should be clarified that if the new element topologyDirectionExtension is present, it takes precedence to the element topologyDirection. Furthermore, the new type should be OPTIONAL.
Reference:	AVD-2805

[Begin Correction]

A.2 ASN.1 syntax specification

```

TopologyRequest ::= SEQUENCE
{
    terminationFrom          TerminationID,
    terminationTo            TerminationID,
    topologyDirection        ENUMERATED
    {
        bothway(0),
        isolate(1),
        oneway(2)
    },
    ...,
    streamID                 StreamID OPTIONAL,
    topologyDirectionExtension  ENUMERATED OPTIONAL
    {
        onewayexternal(0),
        onewayboth(1),
        ...
    }
    -- if present, topologyDirectionExtension takes precedence over
    -- topologyDirection
}

```

[End Correction]

6.2 Correction of ASN.1

Description:	Per clause 11.3/X.680, ASN.1 Identifiers shall start with a lower case character.
Reference:	AVD-2805

[Begin Correction]

A.2 ASN.1 syntax specification

NotifyBehaviour ::= CHOICE
{
 nNotifyImmediate NULL,
 nNotifyRegulated RegulatedEmbeddedDescriptor,
 nNeverNotify NULL,
 ...
}

[End Correction]

6.3 Correction of Annex E.14.6 Procedures

Description:	ABNF in examples of Annex E.14.6 seems to be not inline concerning short token notation of the "SegmentationCompleteToken", as defined in Annex B.2.
Reference:	AVD-2815

[Begin Correction]

E.14.6 Procedures

Example 1:

Sender: !/3 [12.34.56.78]:2944 P=1/1{C=1{AV=term1{...},
AV=term2{...}}}
Receiver: !/3 [12.34.56.79]:2944 SM=1/1
Sender: !/3 [12.34.56.78]:2944
P=1/2{C=1{AV=term3{...}},C=2{AV=term4{...}}}
Receiver: !/3 [12.34.56.79]:2944 SM=1/2
Sender: !/3 [12.34.56.78]:2944 P=1/3/&#{C=3{AV=term5{...}}}
Receiver: !/3 [12.34.56.79]:2944 SM=1/3/&#
Receiver: !/3 [12.34.56.79]:2944 K=1

Example 2:

```
Sender: !/3 [12.34.56.78]:2944 P=1/1{C=1{AV=term1{...},
AV=term2{...}}}}
Receiver: !/3 [12.34.56.79]:2944 SM=1/1
Sender: !/3 [12.34.56.78]:2944 P=1/4/&#{C=3{AV=term5{...}}}}
Receiver: !/3 [12.34.56.79]:2944 SM=1/4/&#
/* Segmentation Timer Expires */
Receiver: !/3 [12.34.56.79]:2944 ER=459{"2,3"}
```

[End Correction]

6.4 Clarification of statistic reset capability

Description:	It is generally agreed that it is possible to reset the value of a statistic in H.248.1 Version 3. However, there is no text describing how this might be achieved.
Reference:	Discussion at November 2005 Geneva Q3 Rapporteur's Meeting

[Begin Correction]

7.1.15 Statistics Descriptor

...

Statistics are cumulative; reporting statistics does not reset them. The value of a Statistic at a termination level is the result of a meaningful superior function (like for instance sum or average) of the values as if it had been placed on all the streams in the termination. Such a superior function is dependent of the particular statistic type. Unless specified otherwise in the package that defines a particular statistic, the default behaviour is a sum of the values. Statistics are reset when a termination ceases to exist or is returned to the NULL ~~context~~ Context due to a Subtract Command.

For terminations not in the NULL Context, the MGC may send a Modify Command with a Statistics Descriptor to disable one or more statistics followed by a second Modify Command including a Statistics Descriptor to re-enable those statistics. As explained above, this has the effect of resetting the included statistics. By bundling the two commands together into the same action or transaction, the MGC can minimize the time during which statistics are not collected by the MG. An audit of the Statistics Descriptor via the Audit Descriptor in the Modify Command or a separate AuditValue Command must be performed before the included statistics are reactivated to collect their values.

...

[End Correction]

6.5 Clarification of delay calculation

Description:	<p>Is the "propagation delay" statistic <code>rtp/delay</code> intended to be</p> <ul style="list-style-type: none"> • the propagation delay in sending direction (PD_{Tx})? • the propagation delay in receiving direction (PD_{Rx})? • the round-trip propagation delay (RTPD) as indicated in § 6.4.1/RFC 3550 (based on RTCP SR information)? • the round-trip delay (RTD), termed as total round-trip time in § 6.4.1/RFC 3550? • the estimated mean one-way propagation delay (OPD) based on the measured round-trip propagation delay? • the "delay since last SR" (DLSR), i.e., RTCP SR header field DLSR (see also § 6.4.1/RFC 3550)? • others? <p>Our understanding is that H.248 statistic <code>rtp/delay</code> is related to the RTPD metric.</p>
Reference:	AVD-2788

[Begin Correction]

E.12.4.5 Delay

Statistic Name: Delay

StatisticID:delay (0x0008)

Description: Requests the current value of packet round-trip propagation delay (RTPD) expressed in timestamp units. This is the same as average latency. The computation of RTPD may be based upon RTCP sender reports (SR) and receiver reports (RR).

Type: Double

Possible values: any 64 bit integer 0 and up

Level: Either

...

[End Correction]

6.6 Protocol version negotiation

Description:	<p>H.248.1 is silent on what to do when either the MGC or the MG fails to abide by the negotiated protocol version within a control association. Consider the following:</p> <p>The MG offers Version 2, which the MGC accepts. The MG then starts sending all messages as Version 1.</p> <p>This is clearly not what was intended in the version negotiation procedures. H.248.1 should allow the receiver of the “off-version” messaging to reject it as not in line with the negotiated version. The most appropriate error code is 406, “Version not supported”.</p>
Reference:	AVD-2820

[Begin Correction]

11.3 Negotiation of protocol version

...

If the MGC supports the version indicated by the MG, ~~it~~ both the MGC and MG shall conform to that version in all subsequent messages. In this case it is optional for the MGC to return a version in the ServiceChange Reply. Any subsequent messaging that does not conform to the negotiated version shall be rejected with Error Code 406 (“Version Not Supported”).

...

[End Correction]

6.7 Clarification of error code usage in wildcarding procedures

Description:	<p>It appears that the error code required in clause 6.3.2/H.248.1 is a cut and paste error, as error code 435 makes more sense. However, the long-standing existence of 431 in that clause may lead to backwards compatibility problems if the error code is just changed. The text needs to be updated to allow either possibility.</p>
Reference:	COM 16-D 223

[Begin Correction]

6.3.2 ContextID wildcarded (ALL) with TerminationID specific

In the case where the ContextID is wildcarded (i.e. ContextID = ALL) and the TerminationID is fully specified, the effect is identical to a command specifying the non-NULL context that contains the specified termination. Thus a search must be made to find the context and only one instance of the command is executed. No errors are reported for contexts that do not contain the specified

termination. If the termination is not contained in any (non-NULL) context then 435 (“TerminationID is not in specified context”) is returned, though Error Code 431 (“No TerminationID matched a wildcard”) may be returned is returned in order to maintain backward compatibilityis returned. If there are no contexts other than NULL in existence, Error Code 411 (“The transaction refers to an unknown ContextID”) is returned. Use of this form of action rather than one specifying the ContextID is discouraged but may be useful, for example in correcting conflicting state between MG and MGC.

For example: Taking the above gateway configuration. The command:

*Context= *{Command=t1/1{Descriptor/s}}*

Returns:

Context=1{Command=t1/1{Descriptor/s}}

...

[End Correction]

6.8 ServiceStates clarification for continuity testing

Description:	The continuity package does not specify whether or not a termination must be placed in the Test state prior to conducting a continuity test.
Reference:	COM 16-D 224

[Begin Correction]

E.10.5 Procedures

...

When a continuity test is performed on a termination, no echo devices or codecs shall be active on that termination. The termination under test does not need to have its ServiceStates Property set to Test.

...

[End Correction]

6.9 Reference to location of ServiceChangeMgcID definition

Description:	There are dedicated sub-clauses for each parameter of ServiceChange in clause 7.8 with one exception: ServiceChangeMgcID is embedded in the section of ServiceChangeAddress. It is proposed to indicate this by editorial changes.
Reference:	COM 16-D 224

[Begin Correction]

F.5.7 ServiceChangeMgcID

The use of the ServiceChangeMgcID parameter is described in clause 7.2.8/H.248.1, specifically clause 7.2.8.1.3/H.248.1. The MGC may send this parameter in a ServiceChange command directed toward the Root Termination. Upon receipt during a registration attempt, the MG shall attempt registration with the MGC at the specified address. When received in a Handoff command from the MG's primary MGC, the MG shall utilize the procedures outlined in clause 11.5/H.248.1.

...

[End Correction]

6.10 Clarification of termination service state upon restart of MG

Description:	During discussion on the 3GPP and IETF Megaco mailing lists it became apparent that there is a source of confusion on the default states of all terminations after a ServiceChange restart. It is widely agreed that all terminations including physical and ephemeral terminations are default "InService" after the ServiceChange. However H.248.1 doesn't explicitly make this statement.
Reference:	COM 16-D 274

[Begin Correction]

7.2.8 ServiceChange

...

- 3) Restart – indicates that service will be restored on the specified terminations after expiration of the ServiceChangeDelay. The ServiceStates Property should be set to "InService" upon expiry of ServiceChangeDelay. Upon receipt of a ServiceChange Command on Root with ServiceChangeMethod Restart all terminations are assumed to be "InService". This includes physical and ephemeral terminations. Those terminations which are "OutOfService" may be reported by subsequent ServiceChange Commands with ServiceChangeMethod Forced.

...

[End Correction]

6.11 Alignment of text among events in the Tone Detection Package

Description:	<p>The Tone Detection Package specifies three different events for tone detection: 'Start Tone Detected', 'End Tone Detected' and 'Long Tone Detected'. While the '*' wildcard in the EventsDescriptor parameter 'tl' is allowed for the 'Start Tone Detected' and 'Long Tone Detected' events, H.248 currently doesn't allow it in the 'End Tone Detected' event.</p> <p>There is no reason why the wildcard should not be allowed in the 'End Tone Detected' event. In fact, a very common use of this event, as of the other two, is the detection of DTMF tones. For DTMF it is common to order the MGW to detect any DTMF digit, as it is not known in advance which DTMF digit will be received in the line.</p>
Reference:	COM16 D-303

[Begin Correction]

E.4.2.2 End Tone Detected

Event Name: End tone detected

EventID: etd (0x0002)

Description: Detects the end of a tone.

EventsDescriptor Parameters:

Tone ID List

Parameter Name: Tone ID List

ParameterID: tl (0x0001)

Description: A list of tone IDs to be detected.

Type: Sublist of enumeration

Optional: No

Possible values: The only tone id defined in this package is "wild card" which is "*" in text encoding and 0x0000 in binary. No possible values are specified in this package. Extensions to this package would add possible values for tone ID. If tl is "wild card", any tone id is detected.

Default: None

[End Correction]

6.12 Clarification of package definition requirements for enumerations

Description:	Packages may define properties, statistics and parameters for signals and events of enumeration type. As stated in the guideline for package definition in H.248.1, the possible values for these parameters must be also specified in the package. Once the values are specified, the binary encoding is unambiguous, as with ASN.1 each of the values of an enumeration type is associated to an integer. However, with text encoding, the encoded values may use any character or character string, not only integers. Therefore it is important that the package specifies not only the possible values that a property, statistic or parameter of type enumeration may take, but also the strings to be used to encode each of the values if ABNF, Annex B/H.248.1 is used.
Reference:	COM 16-D 303

[Begin Correction]

12.1.2 Properties

Properties defined by the package, specifying:

...

Type: One of:

Boolean

String: UTF-8 string

Octet String: A number of octets. See Annex A and B.3 for encoding

Integer: 4 byte signed integer

Double: 8 byte signed integer

Character: Unicode UTF-8 encoding of a single letter. Could be more than one octet.

Enumeration: one of a list of possible unique values. Packages MUST define the text and binary encodings for each value in the enumeration.

Sub-list: a list of several values from a list. The type of sub-list shall also be specified. The type shall be chosen from the types specified in this section (with the exception of sub-list). For example, Type: sub-list of enumeration. The encoding of sub-lists is specified in Annexes A and B.2.

...

12.1.5 Statistics

Statistics defined by the package, specifying:

...

Type: One of:

Boolean

String: UTF-8 string

Octet String: A number of octets. See Annex A and Annex B.3 for encoding

Integer: 4 byte signed integer

Double: 8 byte signed integer

Character: Unicode UTF-8 encoding of a single letter. Could be more than one octet.

Enumeration: One of a list of possible unique values. Packages MUST define the text and binary encodings for each value in the enumeration.

Sub-list: A list of several values from a list. The type of sub-list shall also be specified. The type shall be chosen from the types specified in this section (with the exception of sub-list). For example, Type: sub-list of enumeration. The encoding of sub-lists is specified in Annexes A and B.2.

...

12.2 Guidelines to defining parameters to events and signals

...

Type: One of:

Boolean

String: UTF-8 octet string

Octet String: A number of octets. See Annex A and B.3 for encoding

Integer: 4-octet signed integer

Double: 8-octet signed integer

Character: Unicode UTF-8 encoding of a single letter. Could be more than one octet.

Enumeration: one of a list of possible unique values. Packages MUST define the text and binary encodings for each value in the enumeration.

Sub-list: a list of several values from a list (not supported for statistics). The type of sub-list shall also be specified. The type shall be chosen from the types specified in this section (with the exception of sub-list). For example, Type: sub-list of enumeration. The encoding of sub-lists is specified in Annex A and B.2.

...

[End Correction]

6.13 Clarification on Profile Definition Template

Description:	Appendix III/H.248.1 Profile Definition Template provides tables to specify which descriptors are sent in each command. For each command, request and reply and handled separately with different tables, as different descriptors may be supported by the profile in the request and in the reply. However for the Notify command, a single table is provided with the legend “Descriptors used by Notify Request or Notify Reply”. This is misleading, as the reply does not contain descriptors, and the table should refer only to the Notify Request.
Reference:	COM 16-D 303

[Begin Correction]

5.8.7 Notify

Which descriptors can be used in a Notify Command?

Descriptors used by Notify Request or Reply:	<ObservedEvents, Error>
---	-------------------------

[End Correction]

6.14 Clarification of use of ABNF encodings of octet strings

Description:	Properties, statistics and signal and events parameters can be defined as of type Octet String, among other types. This is described in H.248.1 12.1.2, 12.1.5 and 12.2. These chapters refer to Annex B.3 for how the actual encoding of the Octet String shall be done. Annex B.3 does indeed describe a method for the encoding of strings, but fails to make a precise reference to the type Octet String, as the object of the method it is describing. Instead, it talks about “representing a string of octets” or “encoding octet strings”. As ABNF defines still another type called “octetString” to describe SDP lines, and which is different to the type Octet String defined above (is not compatible), there is a risk to misinterpret the applicability of B.3.
Reference:	COM 16-D 303

[Begin Correction]

B.3 Hexadecimal octet coding

Hexadecimal octet coding is a means of representing ~~a string of octets~~package elements of type Octet String as a string of hexadecimal digits, with two digits representing each octet. This octet encoding should be used when encoding ~~octet strings~~values of type Octet String in the text version of the protocol.

For each octet, the 8-bit sequence is encoded as two hexadecimal digits. Bit 0 is the first transmitted; bit 7 is the last.

Bits 7-4 are encoded as the first hexadecimal digit, with Bit 7 as MSB and Bit 4 as LSB. Bits 3-0 are encoded as the second hexadecimal digit, with Bit 3 as MSB and Bit 0 as LSB.

Examples:

Octet bit pattern	Hexadecimal coding
00011011	D8
11100100	27
10000011 10100010 11001000 00001001	C1451390

This encoding is not applicable to the octetString construct defined in section B.2

[End Correction]

6.15 Clarification of encoding for packet loss statistic in Annex E.12

Description:	H.248.1 E.12.4 defines the statistics packet loss rtp/pl to describe the packet loss rate, as a percentage. Although this statistics element is defined as type double, it is meant to hold both the whole part and the fractional part of the percentage. The ASN.1 “double” encoding of this element entails multiplying the percentage by 2^{32} in order to obtain an integer and then use 4 octets to encode the resulting integer. In ABNF is questionable if the same applies, as that would lead to a long string. This seems unnecessary, especially considering that RFC 3550, to which E.12.4 refers to when defining rtp/pl, defines the packet fraction lost with only 8 bits. Therefore it is proposed to clarify that the notation x.y is allowed when encoding rtp/pl with ABNF.
Reference:	COM 16-D 303

EDITOR’S NOTE: *This item was modified by AVD-2916 at the August 2006 Ottawa meeting.*

EDITOR’S NOTE: *This item was modified by COM16 TD-354/WP2 at the November 2006 Geneva meeting.*

[Begin Correction]

E.12.4.3 Packet Loss

...

Possible values: a 32-bit whole number and a 32-bit fraction. The value shall be encoded in ABNF as “x.y” where x is the whole part and y the fractional part of the percentage.

The actual data type is a fixed point number, which is mapped on the H.248 type “double”. The whole number and the fractional part shall be interpreted as 32-bit integer each, thus the

“double” type for rtp/pl shall be encoded as the concatenation of two integers. The fractional part must be therefore first converted into an integer, i.e. multiplied by 2^{32} .

For example, given the percentage 23.625, to express this in the Packet Loss Statistic we perform the following steps:

1. Convert the number to binary: 23.625_{10} equals 10111.101_2 .

2. Multiply the binary number by 2^{32} . Note that multiplying a binary number by a power of two is the same as shifting it to the left the same number of places as the power of two by which it is being multiplied. $10111101000000000000000000000000$ is the binary value of our statistic.

3. Convert the binary value to a decimal double: the statistic is reported as having value 101468602368 .

To return the statistic back to its fractional representation, the steps are reversed. Once the double is converted back to its binary form, the lower 32 bits represent the fraction and the rest are the whole number. From there, conversion back to a floating point number is fairly straightforward.

Binary encoding shall be as described in clause A.2 for type “integer”.

...

[End Correction]

6.16 Missing ServiceChange parameter from Appendix III

Description:	The ServiceChangeProfile parameter is missing out of the Profile Template in Appendix III.
Reference:	COM 16-D 312

[Begin Correction]

Appendix III

5.8.8 ServiceChange

...

Which version of H.248.1 is used by ServiceChangeVersion? The lowest value here should be the minimum version defined in 5.3.

Version used in ServiceChangeVersion:	<1, 2, 3>
--	-----------

Is the ServiceChangeProfile parameter mandatory?

ServiceChangeProfile mandatory:	<Yes/No>
--	----------

Can multiple profiles be supported according to H.248.18?

Profile negotiation as per H.248.18:	<Yes/No>
---	----------

...

[End Correction]

6.17 Clarification of ServiceChangeMethod Graceful behavior on ephemeral terminations

Description:	The wording in clause F.4.1.3 regarding the use of the Graceful ServiceChangeMethod on an ephemeral termination is confusing, as it makes statements that are clearly the behaviour for ephemeral terminations, but does not actually describe what the scenario described really means.
Reference:	COM 16-D 313

[Begin Correction]

F.4.1.3 ServiceChange Method Behavior on Ephemeral Terminations

...

Graceful – When sent by the MG, it indicates that the termination(s) is going OutOfService at the end of the ServiceChangeDelay period. The MGC is responsible for subtracting the termination(s) at the expiry of the ServiceChangeDelay. The MGC shall not send ServiceChangeMethod “Graceful” for ephemeral terminations. Using a ServiceChangeDelay equal to zero indicates that the termination is going OutOfService immediately. ~~shall be destroyed when it is removed from context through subtraction.~~ The MG should set the termination’s ServiceStates Property at the expiry of ServiceChangeDelay ~~or the removal of the termination from an active context (whichever is first),~~ to "Out of Service". The MGC is responsible for subtracting the termination in either case. To cancel a previously sent (and acknowledged) ServiceChange with ServiceChangeMethod of “Graceful”, the entity initiating the Graceful sends a ServiceChange Command with ServiceChangeMethod Restart and the ServiceChangeReason of 918 Cancel Graceful.

...

[End Correction]

6.18 Clarification of package versions versus protocol versions

Description:	SG16 received a liaison from ETSI TISPAN which highlighted possible confusion between package versioning and protocol versioning. Since only certain packages are linked to the protocol version (those with syntactical requirements in the protocol, like segmentation for example), some text clarifying that this linkage is limited to only certain packages would be of benefit.
Reference:	AVD-2945

12 Package definition

The primary mechanism for extension is by means of Packages. Packages define additional properties that may occur on terminations and contexts and events, signals and statistics that may occur on terminations.

Packages and their versions are not generally related to the version of the protocol being used. If packages defined outside H.248.1 have a dependence upon the syntax or features of a particular version of the protocol, they must explicitly state the minimum version of the protocol upon which they are dependent.

Packages defined by IETF will appear in separate RFCs.

...

E.1 Generic

Package Name: Generic

PackageID: g (0x0001)

Description: Generic package for commonly encountered items

Version: 2

Extends: None

Version 2 of the generic package requires at least version 3 of the protocol.

...

E.2 Base Root Package

Package Name: Base Root Package

PackageID: root (0x0002)

Description: This package defines Gateway wide properties.

Version: 2

Extends: None

Version 2 of the base root package requires at least version 2 of the protocol.

...

E.14 Segmentation Package

Package Name: Segmentation Package

PackageID: seg (0x000x)

Description: This package defines properties for use when performing H.248-based segmentation on non-segmenting transports.

Version: 1

Extends: root version 2

Version 1 of the segmentation package requires at least version 3 of the protocol.

...

E.15 Notification Behaviour

Package Name: Notification Behaviour Package

PackageID: nb (0x009a)

Description: The package has functionality that enables the MG at the request of the MGC to regulate the sending of Notify cCommands. This package has an interaction with the NotifyBehaviour flag described in §clause 7.1.9. This version of the package describes regulation behaviour based upon a percentage regulation. Other types of regulation behaviour are for further study.

Version: 1

Extends: None

Version 1 of the notification behavior package requires at least version 3 for the protocol.

[End Correction]

6.19 Specification of wildcarded response usecases in profile template

Description:	The profile template has a section for specification of the use of commands marked as optional, but does not cover commands that are marked for wildcarded responses.
Reference:	AVD-2843

[Begin Correction]

Appendix III

...

5.10 Transactions

...

Can commands be marked “Optional”? Describe.

Commands able to be marked “Optional”:	<Add, Modify, Move, Subtract, Auditvalue, Auditcapability, Servicechange, All, None>
---	--

Can commands be marked for wildcarded response?

<u>Wildcarded responses may be requested for:</u>	<u><Add, Modify, Move, Subtract, Auditvalue, Auditcapability, Servicechange, All, None></u>
--	---

And/or, describe which commands and the procedures surrounding the use of wildcarded responses with those commands.

<u>Procedures that make use of wildcarded responses:</u>	<u><Describe the procedures making use of wildcarded responses.></u>
---	--

...

[End Correction]

6.20 Specification of termination and stream type support for packages in profile template

Description:	The profile template allows for specification of which packages are supported, but does not allow for indication of what termination types to which the package would apply.
Reference:	AVD-2843

[Begin Correction]

Appendix III

...

5.14 Packages

Specifies the packages that are supported in this profile.

Mandatory: specifies the packages that shall be supported in this profile.

Mandatory Packages:			
Package Name	PackageID	Version	<u>Termination Types Supported</u>
<name>	<xxxx, (0x00xx)>	<1, 2, 3, ...>	<Describe>

Optional: specifies the packages that may be supported in the profile.

Optional Packages:				
Package Name	PackageID	Version	Support dependent on:	<u>Termination Types Supported</u>
<name>	<xxx, 0x00??>	<1, 2, 3, ...>	<Describe>	<Describe>

Package Usage Information

This table specifies how the packages above will be used. For example:

- *it lists whether the properties/signals/events/statistics are optional or mandatory*
- *if the value of the property/signal/event provisioned the provisioned value should be specified. (e.g. names and number of cycles for an H.248.7 announcement).*

Specifies the values of properties which are specified as provisioned

Package Usage Information:

5.14.x <Package Name>

Properties	Mandatory/Optional	Used in command:	Supported Values:	Provisioned Value:	<u>Termination/Stream Types Supported:</u>
<name and Identity e.g. Packets Sent (rtp/ps, 0x000c/0x0004), ALL or None>	<M/O>	<ADD, MOD, MOVE, AUDITVALUE, AUDITCAP>	<Values / ALL >	<Value / Not Applicable>	<Describe>
Signals	Mandatory/Optional	Used in command:		Duration Provisioned Value:	
<name and Identity >	<M/O>	<ADD, MOD, MOVE, AUDITVALUE, AUDITCAP>		<Value / Not Applicable>	
	Signal Parameters	Mandatory/Optional	Supported Values:	Duration Provisioned Value:	
	<name and Identity>	<M/O>	<Values / ALL>	<Value / Not Applicable>	
Events	Mandatory/Optional	Used in command:			
<name and Identity >	<M/O>	<ADD, MOD, MOVE, NOTIFY, AUDITVALUE, AUDITCAP>			
	Event Parameters	Mandatory/Optional	Supported Values:	Provisioned Value:	
	<name and Identity>	<M/O>	<Values / ALL>	<Value / Not Applicable>	
	ObservedEvent Parameters	Mandatory/Optional	Supported Values:	Provisioned Value:	
	<name and Identity>	<M/O>	<Values / ALL>	<Value / Not Applicable>	
Statistics	Mandatory/Optional	Used in command:	Supported Values:	<u>Termination/Stream Types Supported:</u>	
<name and Identity >	<M/O>	<ADD, MOD, MOVE, SUBTRACT, AUDITVALUE, AUDITCAP>	<Values / ALL >	<Describe>	
Error Codes	Mandatory/ Optional				
<number>	<M/O>				

Additional restrictions may be tabulated as the user desires.

...

[End Correction]

6.21 Addition of SDP procedural section to profile template

Description:	The profile template allows a profile to specify what parts of SDP and/or Annex C need to be supported, but does not allow for detailed explanation of the usage of the SDP/Annex C codepoints. It would be useful to add this to the template.
Reference:	AVD-2843

[Begin Correction]

5.15 Mandatory support of SDP and Annex C information elements

Specifies what SDP attributes and Annex C information elements are to be supported.

Supported Annex C and SDP information elements:		
Information Element	Annex C Support	SDP Support <u>(NOTE)</u>
<name>	<Annex C property>	<Describe>

NOTE: Information elements very often map to various lines and their fields in SDP. The profile should specify which SDP lines and fields are mandatory to support.

Discuss in detail the usage scenarios of the mandatory information elements.

5.16 Optional support of SDP and Annex C information elements

Specifies what SDP attributes and Annex C information elements may be supported.

Optional Annex C and SDP information elements:			
Information Element	Annex C Support	SDP Support <u>(NOTE)</u>	Support Dependent on:
<name>	<Annex C property>	<Describe>	<Describe>

NOTE: Information elements very often map to various lines and their fields in SDP. The profile should specify which SDP lines and fields are optional to support.

Discuss in detail the usage scenarios of the optional information elements.

[End Correction]

6.22 Correction to profile negotiation in Appendix I

Description:	The example in Appendix I shows a profile negotiation. It differs from the profile negotiation text, however, in that it shows the MGC including the same profile parameter in the response to the registration. In this instance, the MGC is agreeing to the profile requested by the MG, and so does not need to return this parameter.
---------------------	---

Reference:	AVD-2843
-------------------	----------

[Begin Correction]

I.1.1 Programming residential GW analog line terminations for idle behaviour

...

2) The MGC sends a reply:

MGC to MG1:

MEGACO/1 [123.123.123.4]:55555

Reply = 9998 {

Context = - {ServiceChange = ROOT {

Services {ServiceChangeAddress=55555,~~Profile=ResGW/1~~} } }

}

...

[End Correction]

6.23 Termination ID in Add Reply with error

Description:	In a typical ADD.request the MGC will use \$ (CHOOSE) for the TerminationID. However if there is an error (e.g. 234 Max number of Terminations in a Context exceeded" the MG may not be able to provide a "real" TerminationID. H.248.1 is silent on what TerminationID should be returned in this case. The contributors believe that following the principle that in cases where the MG is unable to determine a TerminationId from a command request this should be returned in the command reply should be followed.
Reference:	Subject: [Megaco] Max terms in context Date: Thu, 13 Jul 2006 14:50:22 -0700 From: Wainwright, John (Com US) < john.wainwright@siemens.com > To: < megaco@ietf.org >

[Begin Correction]

8 Transactions

...

At the first failing command in a transaction, processing of the remaining commands in that transaction stops. If a command contains a wildcarded TerminationID, the command is attempted with each of the actual TerminationIDs matching the wildcard. A response within the

TransactionReply is included for each matching TerminationID, even if one or more instances generated an error. If the MG cannot return a TerminationID in response to a wildcarded termination then the original wildcard TerminationID should be returned. If any TerminationID matching a wildcard results in an error when executed, any commands following the wildcarded command are not attempted.

...

[End Correction]

6.24 TerminationState Descriptor in the profile template

Description:	The Profile template defines entries for the different H.248 protocol elements in order for a MG to state “compliance” against them. Unfortunately the TerminationState Descriptor was missed from this list. However, TerminationState properties can be found in Appendix III § 5.14. A reference to this section should be made from § 5.7.x in order to enhance readability.
Reference:	<p>Subject: Re: [Megaco] TerminationState descriptor</p> <p>Date: Thu, 20 Apr 2006 08:09:27 +1000</p> <p>From: Christian Groves <Christian.Groves@nteczone.com></p> <p>To: B Venkat S.R Swamy <b.swamy@flextronicssoftware.com></p> <p>cc: Albrecht.Schwarz@alcatel.de , megaco@ietf.org , christer.holmberg@ericsson.com</p> <p>References: <OFDB9D405C.94E37E47-ON65257155.002E73B6-65257155.002E9114@flextronicssoftware.com></p>

[Begin Correction]

Appendix III

...

5.7 Descriptors

...

5.7.10 TerminationState Descriptor

What values of TerminationState are supported?

<u>TerminationState : ServiceStates</u>	<u><InService/OutOfService/Test></u>
<u>TerminationState : EventBufferControl</u>	<u><LockStep/OFF></u>

For other package defined Termination State Properties see § 5.14.

...

[End Correction]

6.25 Root and non-Root Terminations in a TerminationIDList

Description:	<p>The termination ID list has been introduced in H.248.1 where a hierarchical TerminationID structure is not possible. When service change command request is sent then whether the termination ID ROOT can be present with other terminations simultaneously in the termination ID list or not.</p> <p>e.g. MEGACO/3 [y.y.y.y]:y Transaction=2000{ Context=- { ServiceChange=[ROOT,T1]{ Services{ Method=Forced, Reason="907 Transmission failure",20060602T04044100} } }</p> <p>It means that termination T1 and MG as a whole are taken out of service (where T1 also belongs to same MG) and here taking T1 out of service is redundant.</p> <p>Whilst there isn't any current H.248.1 text forbidding the use of Root and non-root terminations in the same list however in some cases mixing the two is superfluous. It is recommended to other some guidance on this.</p>
Reference:	<p>Subject: Re: [Megaco] V3 related clarifications required</p> <p>Date: Tue, 20 Jun 2006 15:21:22 +1000</p> <p>From: Christian Groves <Christian.Groves@nteczone.com></p> <p>To: Sudhanshu Garg <sudhanshu.garg@flextronicssoftware.com></p> <p>cc: megaco@ietf.org</p> <p>References: <OFE7511FA7.9F7105BD-ON65257184.00417B2C-65257184.004419F3@flextronicssoftware.com></p>

[Begin Correction]

6.2.2 TerminationIDs

...

TerminationIDs can also be specified in a list. The use of TerminationIDList is recommended for cases where a hierarchical TerminationID structure is not possible and it is not desired to send individual commands for each TerminationID. Both Root and non-Root TerminationsIDs should not be contained in a single TerminationIDList.

[End Correction]

6.26 Codec Selection

Description:	H.248.1 describes procedures for determines capabilities to be supported in the Local and Remote descriptors. These rules can be used to determine the codec/s that is/are to be supported on a stream. The rules allow for symmetric codec selection OR asymmetric codec selection. This may not be clear from the text. It is proposed to add a note to this effect.
Reference:	AVD-2890

[Begin Correction]

7.1.8 Local and Remote Descriptors

...

If ReserveGroup is “False” and ReserveValue is “False”, then the MG should apply the following rules to resolve Local and Remote to a single alternative each:

- The MG chooses the first alternative in Local for which it is able to support at least one alternative in Remote.
- If the MG is unable to support at least one Local and one Remote alternative, it returns Error Code 510 (“Insufficient resources”).
- The MG returns its selected alternative in each of Local and Remote.

NOTE: The above rules allow the MG to prioritize the selection of the same codec in both the Local and Remote Descriptors; however, it also permits the MG to choose different codecs in each descriptor.

A new setting of a Local or Remote Descriptor completely replaces the previous setting of that descriptor in the MG. Thus, to retain information from the previous setting, the MGC must include that information in the new setting. If the MGC wishes to delete some information from the existing descriptor, it merely resends the descriptor (in a Modify Command) with the unwanted information stripped out.

[End Correction]

6.27 Profile negotiation and control association

Description:	<p>There seems to be an apparent conflict in terminology describing profile handling in the case of a ServiceChange Command and Response between H.248.1 section 7.2.8 and Annex F.</p> <p>H.248.1 Section 7.2.8.1.11 ServiceChange Command and Response describes behavior in terms of continuing or keeping a control association, whereas Annex F discusses “Registration.”</p> <p>In order to keep consistency between 7.2.8 and Annex F is proposed to replace the text describing control associations with the text below.</p>
---------------------	---

Reference:	From: Albrecht.Schwarz@alcatel.de To: xiaoshaoping 38157 cc: megaco@ietf.org Sent: Wednesday, June 28, 2006 1:43 PM Subject: Re: [Megaco] Profile negotiation and Control association
-------------------	---

[Begin Correction]

7.2.8.1.11 ServiceChange Command and Response

...

The return from ServiceChange is empty except when the Root TerminationID is used. In that case, it includes the following parameters as required:

- ServiceChangeAddress, if the responding MGC wishes to specify a new destination for messages from the MG for the remainder of the association;
- ServiceChangeMgcID, if the responding MGC does not wish to sustain an association with the MG;
- ServiceChangeProfile, if the responder wishes to negotiate the profile to be used for the association. The profile (name and version) is only returned in the reply in the case that the MGC cannot support the specified profiles in the ServiceChangeRequest. The returned reply shall indicate the profile and version supported or "NoProfile" if no profile is supported. Upon reception of a profile in the reply the MG may continue the relationship with the current MGC or contact secondary MGCs and establish a relationship with them. If the MGC returned a profile in the reply other than the one that was provided in the request the MG shall:
 - a. ~~continue the control association by issuing~~ a new ServiceChange Command with an agreed profile to confirm to the MGC that the MG has agreed with the profile, or
 - b. ~~keep the control association active, so that~~ consider the registration stage completed and await further commands from the MGC; the MGC will use the profile that it sent in the ServiceChange Reply, or
 - c. initiate a control association with a different MGC using its original profile.

NOTE: In order to ensure both the MGC and MG are using the same profile, option a) is encouraged.

...

[End Correction]

6.28 Clarification of use of ServiceChangeAddress

Description:	The ServiceChange parameter "ServiceChangeAddress" was discussed in the April 2006 meeting in the context of H.Sup7 (see D-219 & D-217). Potential uses cases were identified and described in § 5.5.1.2/H.Sup7.
Reference:	AVD-2917

[Begin Correction]

7.2.8.1.3 ServiceChangeAddress

The optional ServiceChangeAddress parameter specifies the address (e.g. IP port number for IP networks) to be used for subsequent communications. It can be specified in the input parameter descriptor or the returned result descriptor. ServiceChangeAddress and ServiceChangeMgcID parameters must not both be present in the ServiceChange Descriptor or the ServiceChangeResult Descriptor. The ServiceChangeAddress provides an address to be used within the context of the association currently being negotiated, while the ServiceChangeMgcID provides an alternate address where the MG should seek to establish another association. ~~Note that the use of ServiceChangeAddress is not encouraged.~~ MGCs and MGs must be able to cope with the ServiceChangeAddress being either a full address or just a port number in the case of TCP-IP-based transports, such as UDP, TCP or SCTP.

...

F.5.4 ServiceChangeAddress

The use of the ServiceChangeAddress parameter is described in clause 7.2.8/H.248.1. ~~The use of the ServiceChangeAddress is discouraged.~~ If the parameter is present, it may only be used with ServiceChange commands on the Root Termination, and any new transactions shall be sent to the new address and/or port number specified. Replies shall be sent back to the address from which the corresponding request came.

[End Correction]

6.29 Correction of typographical error in Appendix III Clause 5.7.4

Description:	The “Halted by new Signals Descriptor” NotifyCompletion method is mistyped.
Reference:	AVD-2921

[Begin Correction]

Appendix III

...

5.7.4 Signals Descriptor

...

Is “notifyCompletion” supported? What types are supported? Is the RequestID used with “NotifyCompletion”?

NotifyCompletion supported:	<Yes/No>	
<i>If yes</i>	SignalID	Type of completion supported
	<Signal name and Identity e.g. Playtone (tonegen/pt, 0x0003/0x0001) or ALL>	<ALL, TO, EV, ED <u>SD</u> , NC, PI>
RequestID Parameter supported:	<Yes/No>	

...

[End Correction]

6.30 Clarification of length encoding in ASN.1 syntax

Description:	<p>X.690 defines a set of Distinguished Encoding Rules (DER) and a set of Canonical Encoding Rules (CER) both of which provide constraints on the Basic Encoding Rules (BER). The key difference between them is that DER uses the definite length form of encoding while CER uses the indefinite length form.</p> <p>Theoretically use of both length encoding schemes of X.690 is possible for H.248 with binary encoding. But, practically, X.690, § 8.1.3.2 (c) is NOT relevant for H.248, because "all (to be encoded protocol data is) immediately available", when considering instances of H.248-served user implementations.</p> <p>Thus, the "definite form of length" should be sufficient. Either way, some statement of support should be made.</p>
Reference:	AVD-2922

[Begin Correction]

5.9 Generic command syntax and encoding

Specifies what encodings are supported by the profile.

Supported Encodings:	<Text and Binary, Binary, Text>
<u>If binary encoding, is indefinite length encoding supported:</u>	<Yes, No>

[End Correction]

6.31 Clarification for context attribute values when omitted from actions

Description:	Clause 6.2.4/H.248.1 clarifies the handling of properties and descriptors for terminations, and clause 7.1.19/H.248.1 clarifies the handling of properties specified through the ContextAttribute Descriptor. Such a clarification would be also beneficial for attributes on context-level.
Reference:	AVD-2890

[Begin Correction]

6.1.1 Context attributes and descriptors

The attributes of contexts are:

- ContextID.
- The Topology Descriptor (who hears/sees whom).
The topology of a context describes the flow of media between the terminations within a context. In contrast, the Mode Property of a termination (“SendOnly”/”RecvOnly”/...) describes the flow of the media at the egress/ingress of the media gateway.
- The priority is used for a context in order to provide the MG with information about a certain precedence handling for a context. The MGC can also use the priority to control autonomously the traffic precedence in the MG in a smooth way in certain situations (e.g. restart), when a lot of contexts must be handled simultaneously. Priority 0 is the lowest priority and a priority of 15 is the highest priority.
- An indicator for an emergency call is also provided to allow a preference handling in the MG.
- An indicator for an IEPS call is provided to allow the features and techniques of E.106 to be achieved.
- A ContextAttribute Descriptor that enables extra context attributes to be defined by using the packages extension mechanism (see clause 7.1.19).

In general, if a context attribute is completely omitted from a H.248 action, the attribute of the corresponding context retains its prior value.

[End Correction]

6.32 Clarification of the use of ServiceChange Disconnected when re-establishing comms

Description:	Discussions on the mailing list have highlighted that the use of ServiceChange Disconnected in the instance of lost communications with the MGC are not entirely clear. Some clarification would help make things clearer.
Reference:	AVD-2925

[Begin Correction]

F.3.6 MG Lost Communication

When the MG has detected a loss and subsequent re-establishment of communication with the MGC (NOTE 1), the MG sends a ServiceChange Command (NOTE 2) with a ServiceChangeMethod of “Disconnected” to the MGC in the current control association. If the MGC fails to respond, the MG then sends a ServiceChange Command with a ServiceChangeMethod of “Failover” and ServiceChangeReason 909 (“MGC Impending Failure”) to each MGC in its list in turn until it has successfully established a new control association, or it has exhausted its list of MGCs. If the MGC does respond, the control association continues as if it were not interrupted.

NOTE 1: The two main causes for lost communications between the MGC and MG are 1) failures or short-term interruptions of the H.248 transport connection, or 2) the primary MGC going “OutOfService”. The MG will not necessarily be able to discriminate between the two, therefore the ServiceChange procedures are the same in both cases.

NOTE 2: The MG may send one or more ServiceChange Commands. The transmission of subsequent ServiceChange Commands may be timer-controlled. Multiple re-establishment attempts may help in situations with short-term failures, either of the transport connection or of the MGC, thereby avoiding the invocation of failover procedures when they are not warranted.

...

[End Correction]

6.33 Clarification of use of unsigned integer

Description:	It was discovered when discussing statistics and their encoding that unsigned integer was not specified in clause 12, but is defined in both the ASN.1 and ABNF encodings. Since the protocol is capable of encoding this, it should be available for use in packages.
Reference:	Discussions at August 2006 Ottawa meeting

[Begin Correction]

12.1.2 Properties

Properties defined by the package, specifying:

Property Name: only descriptive

PropertyID: is an identifier.

Description: is a description of the function of the property

Type: One of:

Boolean

String: UTF-8 string

Octet String: A number of octets. See Annex A and B.3 for encoding

Integer: 4 byte signed integer

Unsigned Integer: 4-octet unsigned integer

Double: 8 byte signed integer

Character: Unicode UTF-8 encoding of a single letter. Could be more than one octet.

Enumeration: one of a list of possible unique values.

Sub-list: a list of several values from a list. The type of sub-list shall also be specified. The type shall be chosen from the types specified in this section (with the exception of sub-list). For example, Type: sub-list of enumeration. The encoding of sub-lists is specified in Annexes A and B.2.

...

12.1.5 Statistics

Statistics defined by the package, specifying:

Statistic name: only descriptive

StatisticID: is an identifier

StatisticID is used in a Statistics Descriptor

Description: a description of the statistic

Type: One of:

Boolean

String: UTF-8 string

Octet String: A number of octets. See Annex A and Annex B.3 for encoding

Integer: 4 byte signed integer

Unsigned Integer: 4-octet unsigned integer

Double: 8 byte signed integer

Character: Unicode UTF-8 encoding of a single letter. Could be more than one octet.

Enumeration: One of a list of possible unique values.

Sub-list: A list of several values from a list. The type of sub-list shall also be specified. The type shall be chosen from the types specified in this section (with the exception of sub-list). For example, Type: sub-list of enumeration. The encoding of sub-lists is specified in Annexes A and B.2.

...

12.2 Guidelines to defining parameters to events and signals

Parameter Name: only descriptive

ParameterID: is an identifier. The textual ParameterID of parameters to events and signals shall not start with "EPA" and "SPA", respectively. The textual ParameterID shall also not be "ST", "Stream", "SY", "SignalType", "DR", "Duration", "NC", "NotifyCompletion", "KA", "KeepActive", "EB", "Embed", "DM", "DigitMap", "SPADI", "SPADirection", "SPARQ" or "SPARquestID".

Description: a description of the function of the parameter.

Type: One of:

Boolean

String: UTF-8 octet string

Octet String: A number of octets. See Annex A and B.3 for encoding

Integer: 4-octet signed integer

Unsigned Integer: 4-octet unsigned integer

Double: 8-octet signed integer

Character: Unicode UTF-8 encoding of a single letter. Could be more than one octet.

Enumeration: one of a list of possible unique values.

Sub-list: a list of several values from a list (not supported for statistics). The type of sub-list shall also be specified. The type shall be chosen from the types specified in this section (with the exception of sub-list). For example, Type: sub-list of enumeration. The encoding of sub-lists is specified in Annex A and B.2.

...

[End Correction]

6.34 Correction to SDP examples in Appendix I

Description:	The SDP used in the example in Appendix I is not compliant to RFC 2327 or RFC 4566 in that the t= line is appearing before the c= line, even though the c= line applies to the entire description.
Reference:	COM16 C-69

[Begin Correction]

I.1.2 Collecting originator digits and initiating termination

...

13) MG1 acknowledges the new termination and fills in the Local IP address and UDP port. It also makes a choice for the codec based on the MGC preferences in Local. MG1 sets the RTP port to 2222.

```

MEGACO/3 [124.124.124.222]:55555
Reply = 10003 {
    Context = 2000 {
        Add = A4444,
        Add=A4445{
            Media {
                Stream = 1 {
                    Local {
v=0
o=- 2890844526 2890842807 IN IP4 124.124.124.222
s=-
t=0 0
c=IN IP4 124.124.124.222
t= 0 0
m=audio 2222 RTP/AVP 4
a=ptime:30
a=recvonly
    } ; RTP profile for G.723.1 is 4
    }
    }
}

```

...

15) This is acknowledged. The stream port number is different from the control port number. In this case it is 1111 (in SDP).

```

MG2 to MGC:
MEGACO/3 [125.125.125.111]:55555
Reply = 50003 {
    Context = 5000 {
        Add = A5555,
        Add = A5556{
            Media {
                Stream = 1 {
                    Local {
v=0
o=- 7736844526 7736842807 IN IP4 125.125.125.111
s=-
t=0 0
c=IN IP4 125.125.125.111
t= 0 0
m=audio 1111 RTP/AVP 4
    } ; RTP profile for G.723.1 is 4
    }
    }
}

```


16) The above IPAddr and UDPport need to be given to MG1 now.

MGC to MG1:

```
MEGACO/3 [123.123.123.4]:55555
Transaction = 10005 {
  Context = 2000 {
    Modify = A4444 {
      Signals {cg/rt}
    },
    Modify = A4445 {
      Media {
        Stream = 1 {
          Remote {
v=0
o=- 7736844526 7736842807 IN IP4 125.125.125.111
s=-
t=0-0
c=IN IP4 125.125.125.111
t= 0 0
m=audio 1111 RTP/AVP 4
      }
    } ; RTP profile for G.723.1 is 4
  }
}
}
```

...

20) The MG2 replies.

```
MEGACO/3 [125.125.125.111]:55555
Reply = 50007 {
  Context = 5000 {
    AuditValue = A5556 {
      Media {
        TerminationState { ServiceStates = InService,
          Buffer = OFF },
        Stream = 1 {
          LocalControl { Mode = SendRecv,
            nt/jit=40 },
          Local {
v=0
o=- 7736844526 7736842807 IN IP4 125.125.125.111
s=-
t=0-0
c=IN IP4 125.125.125.111
t= 0 0
m=audio 1111 RTP/AVP 4
a=ptime:30
      },
    }
  }
}
```

```

Remote {
v=0
o=- 2890844526 2890842807 IN IP4 124.124.124.222
s=-
t=0 0
c=IN IP4 124.124.124.222
t= 0 0
m=audio 2222 RTP/AVP 4
a=ptime:30
    } } },
Events,
Signals,
DigitMap,
Packages {nt-1, rtp-1},
Statistics { rtp/ps=1200, ; packets sent
              nt/os=62300, ; octets sent
              rtp/pr=700, ; packets received
              nt/or=45100, ; octets received
              rtp/pl=0.2, ; % packet loss
              rtp/jit=20,
              rtp/delay=40 } ; avg latency
    }
}

```

...

[End Correction]

6.35 Clarification of SDP requirements for the MG

Description:	There have been many questions about whether the MG is required to be compliant to RFC 2327 with regards to SDP it sends. This is due to the exceptions outlined in clause 7.1.8 that apply only to the MGC. The MG requirement needs to be explicit.
Reference:	Discussions at November 2006 Geneva meeting based on COM 16 TD 271/GEN

[Begin Correction]

7.1.8 Local and Remote Descriptors

...

Local refers to the media received by the MG and Remote refers to the media sent by the MG.

When text encoding the protocol, the descriptors consist of session descriptions as defined in SDP (RFC 2327). In session descriptions sent from the MG to the MGC, the SDP must comply with RFC 2327. In session descriptions sent from the MGC to the MG, the following exceptions to the syntax of RFC 2327 are allowed:

...

[End Correction]

6.36 Correction for Segmentation Error handling

Description:	The H.248.1 version 3 segmentation procedures specify that error 459 should be sent if all the segmented responses are not received. However, the sample call flow does not show which transaction the error message is addressing to. Is it correct? Should there be the transaction id in the error message returned? In order to correlate the error with the applicable transaction the error should also be returned. Unfortunately the H.248.1 syntax does not allow this. Thus it is proposed that this error is returned as error text in the error descriptor.
Reference:	Subject: [Megaco] Segmentation error Date: Thu, 25 May 2006 17:53:32 -0400 From: Ron Ho < ron.ho.megaco@gmail.com > To: megaco < megaco@ietf.org >

[Begin Correction]

E.14.5 Error Codes

Error Code #: 459

Name: Segments not received

Definition: This error indicates that the recipient of a segmented TransactionReply timed out waiting for all the segments to be delivered.

Error Text in the Error Descriptor: The TransactionID followed by the missing segment numbers are included.

Comment: –

E.14.6 Procedures

...

Example 2:

```
Sender:    !/3 [12.34.56.78]:2944 P=1/1{C=1{AV=term1{...},
AV=term2{...}}}}
Receiver:  !/3 [12.34.56.79]:2944 SM=1/1
Sender:    !/3 [12.34.56.78]:2944 P=1/4/#{C=3{AV=term5{...}}}}
Receiver:  !/3 [12.34.56.79]:2944 SM=1/4/#
/* Segmentation Timer Expires */
Receiver:  !/3 [12.34.56.79]:2944 ER=459{"1,2,3"}
```

...

[End Correction]

6.37 Clarification of package definition and registration procedures

Description:	<p>Over the last several years a number of organisations have published H.248 packages and have then sought that these packages be adopted by ITU Study Group 16. During review Q.3/16 have identified a number of short comings in the use of H.248 in these packages. As the packages have already been published it is problematic to rectify these issues. In addition the IETF MECAGO working group is now disbanded so there is no formal mechanism for review also.</p> <p>As such given this situation it is proposed to update the package definition and registration rules in order that Q.3/16 is given a chance to review the package before a package identity is received and before the package can be published. This is only applicable to public packages.</p>
Reference:	COM 16-C 63

[Begin Correction]

12 Package definition

The primary mechanism for extension is by means of Packages. Packages define additional properties that may occur on terminations and contexts and events, signals and statistics that may occur on terminations.

Packages defined by IETF will appear in separate RFCs.

Packages defined by ITU-T may appear in the relevant Recommendations (e.g., as in the H.248 subseries of Recommendations).

- 1) A public document or a standard forum document, which can be referenced as the document that describes the package following the guideline above, should be specified.
- 2) The document shall specify the version of the Package that it describes.
- 3) The document should be available on a public web server and should have a stable URL. The site should provide a mechanism to provide comments and appropriate responses should be returned.

Package definitions should be sent to the ITU-T question responsible (Study Group 16 Question 3 at the time of approval of this Recommendation) for H.248 for review at the earliest possible stage (at least before approval) in order to ensure that the package is consistent with H.248 methodologies. IANA will not mark the package as being “Final” until the package is deemed suitable by the ITU-T.

...

14 IANA considerations

14.1 Packages

The following considerations shall be met to register a package with IANA:

- 1) A unique string name, unique serial number and version number is registered for each package. The string name is used ~~with~~ as the PackageID for text encoding. The serial number ~~shall be~~ is used ~~with~~ as the PackageID for binary encoding. Serial Numbers 0x8000 to 0xFFFF are reserved for private use. Serial number 0 is reserved.

2) A contact name, email and postal addresses for that contact shall be specified. The contact information shall be updated by the defining organization as necessary.

3) A reference to a document that describes the package, which should be public:

The document shall specify the version of the package that it describes.

If the document is public, it should be located on a public web server and should have a stable URL. The site should provide a mechanism to provide comments and appropriate responses should be returned.

If the document is not public, it must be made available for review at the time of the application.

4) Packages registered by other than recognized standards bodies shall have a minimum package name length of 8 characters.

5) All other package names are first come-first served if all other conditions are met.

IANA shall register the following information in the registry:

1. Binary ID (or serial number)

2. Text ID

3. Package version

4. Extension information – Binary ID and package version

5. Status – one of IP (“In Progress”), UR (“Under Review”) or Final

6. Reference and Contact information

IANA shall register packages upon request when the five criteria for registering a package are met. The documenting text does not have to be publicly available at the time of the registration request, as long as the text is made available for review at the time of application. IANA shall register new packages with a status of “IP”. Registration should occur as early as practicable in the design process, to reserve a binary ID. Binary IDs shall be published in the document defining the package.

Public packages under development shall be sent to the ITU-T question responsible for the H.248 Subseries for review. This should occur as soon as practicable after the rough draft of the definition is completed and at least before the package is approved in order to ensure the package is consistent with H.248 methodologies and package design principles. If the package is deemed suitable, the ITU-T question shall issue a statement indicating its approval, copied to IANA.

Once the package is complete, IANA shall be notified of the completion of the package by the group developing the package. Upon receipt of this notification, IANA shall update the registration details. If IANA has received the approval indication from ITU-T, the status of the package shall be set to “Final”. If IANA has not received this notification, the status shall be set to “UR” and IANA shall inform a designated ITU-T representative of the requested status update. This representative should be the Rapporteur of the responsible question or a designated delegate. In either case, IANA shall notify the group developing the package of the outcome of the status update.

If the ITU-T responds to IANA’s update notification with an approval indication, IANA shall update the status to “Final”. If the ITU-T responds that the package has not been reviewed, or was deemed unsuitable, IANA shall alter the status back to “IP”. In either case, IANA shall notify the developing group of the update.

A package shall not be set to status “Final” without the express approval of the ITU-T.

[End Correction]

6.38 Clarification of the Octets Sent and Octets Received Statistics in the nt and rtp packages

Description:	There continues to be some confusion as to what “transport overhead” means in calculating the Octets Sent and Octets Received Statistics in the nt and rtp packages. There needs to be explicit text in both packages defining exactly how the statistics are calculated.
Reference:	COM16 C-88

[Begin Correction]

E.11.5.1.5 Calculating the Octets Sent and Octets Received Statistics

When calculating the os and or statistics, the transport overhead is to be excluded. This means that the network transport overhead is not included, but overhead for the transport protocol and application layer is included in the count. For example, in an IP network using UDP for the transport protocol, to calculate the os statistic, the overhead associated with layers 1-3 (IP and below) is excluded, but overhead for layers 4-7 (UDP and above) is included in the count. Figure E.1 illustrates the concept of an IP stack and various possibilities for the definition of transport overhead. For the nt package, option c is the correct definition.

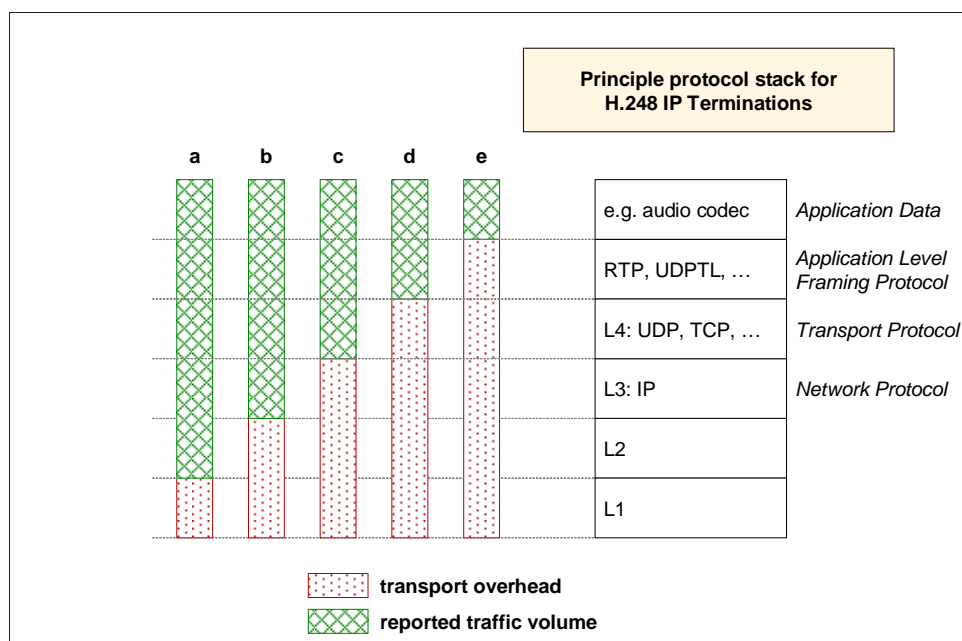


Figure E.1/H.248.1 – Possible transport overhead calculations for IP-based networks

...

E.12.5 Procedures

E.12.5.1 Working with RTP

When RTCP is associated with an RTP stream, RTCP shall remain unaffected by the H.248.1 Mode Property in the LocalControl Descriptor.

When RTCP is associated with an RTP stream and the MG receives an Empty Remote Descriptor for that stream, the MG shall stop the RTCP stream along with the corresponding RTP stream.

E.12.5.2 Calculation of the Octets Sent and Octets Received Statistics

Because the os and or statistics are inherited from the nt package, they must retain the same calculation for both packages. Therefore, these statistics are calculated as defined in the nt package, even when addressed in the rtp package. This means that the octet counts include all data from layer 4 and above, including UDP, RTP and the application data.

[End Correction]

7 Technical and Editorial Corrections to H.248.5 (2000)

7.1 Clarification of “At-Most-Once” Functionality

Description:	<p>At the January 2004 Geneva SG16 meeting D376 introduced a clarification to H.248.4 on the issue of providing the at most once functionality. It described the issue as:</p> <p><i>“In section 3 “Providing the at most once functionality” the procedure recommends that the procedures of H.248 Annex D.1.1 be followed apart from the use of LONG TIMER and TransactionResponseAck.</i></p> <p><i>When referencing a potential confusion exists in that the procedure to compare and remove duplicate transaction identities uses the LONG TIMER. Readers may assume that as LONG TIMER is not used then the procedure of comparing and identifying duplicate transaction identities is also not supported. This is an incorrect assumption. It is proposed to clarify that procedures to identify duplicate transaction ID are needed.”</i></p> <p>H.248.5 has the same text with regards to providing at most once functionality. Thus it is proposed that a clarification is added that the MTP backward sequence number is added to H.248.5.</p>
Reference:	AVD-2663

[Begin Correction]

2 References

– Recommendation ITU-T Q.703 (07/1996), *Specifications of Signalling System No. 7 – Message transfer part.*

...

4.1 Providing At-Most-Once functionality

Messages, being carried over MTP3b, may be subject to losses. In the absence of a timely response, commands are repeated. Most commands are not idempotent. The state of the MG would become unpredictable if, for example, Add commands were executed several times. The transmission procedures shall thus provide an "At-Most-Once" functionality.

The procedures in D.1.1/H.248.1 shall be followed with two exceptions:

- The LONG-TIMER shall not be used to remove a Transaction Identity from the list of responses. The MTP Backward Sequence Number (as defined in Q.703 § 5.2) or a response to the requested command shall be used.;
- The TransactionResponseAck parameter shall not be used.

...

[End Correction]

8 Technical and Editorial Corrections to H.248.8 (2005)

8.1 Error text for error code 449

Description:	The new version of H.248.8 has introduced the new error code 449 “Unsupported or Unknown Parameter or Property Value”. It states that in the error text, the receiver of the command shall specify the unsupported or unknown property value. However, the sender of the command, when it receives the error reply, doesn’t necessarily know to which parameter it corresponds.
Reference:	AVD-2805

[Begin Correction]

4.2.27 Error Code #: 449

Name: Unsupported or Unknown Parameter or Property Value

Definition: The value of a Property or Parameter within a descriptor is not recognized and the command including the property/item is not carried out.

Error Text in the error Descriptor: The parameter or property name followed by the unsupported or unknown value is included in the error text in the error descriptor.

Comment: –

[End Correction]

8.2 Protocol error on command level

Description:	H.248.8 defines Error Code 401 “Protocol Error” on transaction level, indicating that the transaction request has been disregarded due to a violation of Megaco procedures. There are situations when protocol error could occur at command level and therefore it is proposed to allow this error code at command level.
Reference:	COM 16-D 303

[Begin Correction]

4.2.2 Error Code #: 401

Name: Protocol Error

Definition: The transaction or command request(s) has been disregarded due to a violation of Megaco protocol procedures having been detected.

Error Text in the Error Descriptor: –

Comment: Use more specific error codes (e.g. 505) if possible.

[End Correction]

8.3 New error code 511

Description:	H.248.8 defines Error Code 510 “Insufficient resources” to indicate that the MG can not execute the command due to lack of resources. The MGC may also experience temporary congestion of software resources. Therefore a new error code is proposed to indicate that the MG has temporary lack of resources to execute the command.
Reference:	D-303

[Begin Correction]

4.2.47 Error Code #: 511

Name: Temporarily Busy

Definition: The command(s) was rejected due to a temporary busy condition in the MGC.

Error Text in the Error Descriptor: None

Comment: Upon receiving this error code, the command may be resent as the busy condition may have abated.

...

{Editor’s Note: Subsequent sections of H.248.8 will be renumbered when H.248.8 is next amended.}

[End Correction]

8.4 Clarification of the use of error text

Description:	H.248.8 specifies the use of ErrorText for some error codes. For many error codes, however, the use of ErrorText is not specified in H.248.8. In those cases, either ‘-’ or “NA” is indicated. It’s Ericsson’s opinion that H.248.8 does not preclude the use of Error Text
---------------------	--

	in any error codes, even if a particular use is not indicated in the recommendation. It is of course understood that the use of error text in those cases not specified in H.248.8 would mainly serve tracing and debugging purposes, i.e. no corrective action based on the content of the error text is expected from the other end, if the error text is not fully specified.
Reference:	AVD-2838

[Begin Correction]

4.1 Assigning Error Codes

The following considerations SHALL be met to register an error code with IANA:

- 1) An error number and a one-line (80 character maximum) string isare registered for each error.
- 2) A complete description of the conditions under which the error is detected shall be included in a publicly available document. The description shall be sufficiently clear to differentiate the error from all other existing error codes.
- 3) The document should be available on a public web server and should have a stable URL.
- 4) Error numbers registered by recognized standards bodies shall have 3- or 4-character error numbers.
- 5) Error numbers registered by all other organizations or individuals shall have 4-character error numbers.
- 6) An error number shall not be redefined, nor modified except by the organization or individual that originally defined it, or their successors or assigns.

Parameters included in the error text shall be coded according to the principles as specified in Annexes A or B/H.248.1. If more than one parameter is included they shall be included in the order defined in this Recommendation and separated by a comma.

Error text may be included in the Error Descriptor, even for those error codes for which no specific error text is defined in this recommendation. If error text is included it should contain the information defined in the error specification.

NOTE – The actions carried out after an error is discovered, which results in the sending of an error code described in 4.2, are specified in clause 8/H.248.1.

[End Correction]

9 Technical and Editorial Corrections to H.248.9 (2005)

9.1 Clarification on terminating PlayRecord successfully via MGC command

Description:	<p>This is the basic scenario: The media server is used to record a message for a subscriber. The user finishes recording and hangs up. As a result, a SIP BYE is sent to the MGC controlling the media server.</p> <p>The media server at this point has detected silence and has started a post speech timer. This timer is necessary to handle pauses in natural speech.</p>
---------------------	---

	<p>The MGC receives the BYE. It wants to terminate the running PlayRecord Signal in such a way that the recording is successful and the results of the recording (the URL where the recording is stored, the recording length, etc) are returned to the MGC. Having the MGC to wait for the post speech timer to trip is not acceptable for a number of reasons, including but not limited to resource utilization, etc.</p> <p>NotifyCompletion was examined as a possible mechanism. NotifyCompletion can be used to generate a SignalComplete Event when the PlayRecord Signal is terminated due to the receipt of an empty Signals Descriptor. The problem with NotifyCompletion is that the recording information is not returned to the MGC with the SignalComplete Event. What is needed is a mechanism that lets the MGC terminate the PlayRecord Signal in such a way that the signal returns success (if possible) along with the recording parameters.</p> <p>There is no current mechanism described in H.248 to do this. However, a convention seems to have sprung up in the vendor community to handle this problem. The MGC sends an empty Signals Descriptor along with a new Events Descriptor containing the PlayRecord Success and Audio Operation Failure Events. The MG interprets this as a request to terminate the playrec Signal immediately, successfully if possible (i.e. if audio has been recorded), and to return the precsucc or audfail Event with the event parameters that describe the recording.</p> <p>The H.248.9 spec should document this industry convention, so as to promote the widest possible interoperation.</p>
Reference:	AVD-2820

[Begin Correction]

10.5 Procedures

...

Failure of the MakePersistent signal must be reported as an appropriate error code in the response to the transaction invoking it. That is, the response must not be returned to the MGC until the outcome of the MakePersistent operation is known.

When the MGC sends a Modify Command containing a Signals Descriptor that no longer contains the playrec Signal along with a new Events Descriptor containing the audfail and precsucc Events, the MG should interpret this as a request to immediately halt recording and return precsucc if the recording succeeded or audfail if the recording failed.

...

[End Correction]

9.2 Correction in type-ahead handling

Description:	<p>H.248.9 says that type-ahead is supported, and can be controlled with the clear digit buffer command. It goes on to say that the digit buffer should be cleared at the start of each play-collect, and then cleared on reattempts within the signal, if the clear digit buffer parameter is set to true.</p> <p>This is an error: the digit buffer should be cleared on reattempts within the signal, regardless of the setting of the clear digit buffer parameter.</p> <p>Type-ahead should work in the following manner:</p> <ul style="list-style-type: none"> • For the first play-collect or play-record in a call, the digit buffer should be cleared to make sure there are no digits left over from the previous call. • For re-attempts within a single play-collect or play-record, the buffer should always be cleared. By definition, the previous attempt failed, so the subsequent attempt should start with a clean slate. • For play-collects or play-records subsequent to the first play-collect or play-record for the call, the digit buffer should only be cleared if the clear buffer parameter is set to true. <p>This resolves issues with holding over digits from a previous failed attempt, thereby corrupting a subsequent attempt.</p>
Reference:	AVD-2820

[Begin Correction]

9.5.1 PlayCollect digit processing model

- ...
- 2) Collection loop. Increment number of attempts. Initialize digit map processing. If ~~<ClearDigitBuffer> is TRUE, clear the digit collection buffer.~~ Clear the digit collection buffer if:
- this is the first PlayCollect Signal set on this termination in this context,
 - the number of attempts within this PlayCollect Signal is greater than one, or
 - <ClearDigitBuffer> is TRUE.
- ...
- 11) Failure to match digit map. Check number of attempts. If it is equal to <MaxAttempts>, play <FailureAnnouncement> if one has been specified, exit and generate an Audio Operation Failure event with return code 619 "Max Attempts Exceeded". Otherwise set current announcement to <Reprompt>, ~~retain any digit accumulation buffer contents beyond the digits already processed (i.e., discarding the digit which "broke the pattern")~~ discard all digit accumulation buffer contents, and return to step 2).

[End Correction]

10 Technical and Editorial Corrections to H.248.14 (2002)

10.1 Provisioning of the Inactivity Timeout Event

Description:	<p>The development of H.248.14 was controversial in that people did not see the necessity for it for high traffic gateways. However it was agreed that the MGC shouldn't be burdened with having to do large amount of extra signalling or processing in order to get this inactivity timer functionality. AVD-2119 shows the original rationale behind the proposal.</p> <p>Given the history of H.248.14 the contributors believe the following statements can be made:</p> <ol style="list-style-type: none">1. In the original discussions in mid-2001 people favoured a mechanism that did not cause extra burden on the MGC.2. H.248.14 was seen to be used for residential gateways where the signalling load was insufficient for a keep-alive mechanism.3. There could be tens of thousands of residential gateways per MGC.4. If the MGC had to set the "timeout" event on each gateway at restart this would cause a significant signalling load on the MGC. <p>Therefore in keeping with one of the original aims of H.248.14 and given its nature of use with large numbers of MGs it is proposed to recommend that the inactivity timeout event should be provisioned. Furthermore this recommendation is extended to other events which may be provisioned at start up. Another benefit of the provisioning is that the MG may detect if the MGC fails immediately after restart and can take corrective action.</p>
Reference:	COM 16-D 274

[Begin Correction]

5.5 Procedures

...

If the MGC has failed, the event will not receive a reply. If no reply is received, the MG will consider the MGC to have failed and will follow the procedures of 11.5/H.248.1.

NOTE – To minimize signalling load at MGC restarts in networks where there are large numbers of MGs that implement the Inactivity Timer Package, the Inactivity Timeout Event may be provisioned in both the MGC and MG. This also enhances the ability of the MG to detect MGC failures immediately after the MGC restarts.

...

[End Correction]

11 Technical and Editorial Corrections to H.248.17 (2002) Corr. 1 (2004)

11.1 Correction of Typographical error in Clause 8.3.1.2/H.248.17

Description:	In clause 8.3.1.2/H.248.17 there is a reference to the "bcg/bdtq" (A.8/Q.1950) signal. This should be "bcg/bdt".
Reference:	AVD-2843

[Begin Correction]

8.3.1.2 Auto Answering Trunk (AAT)

Auto Answering Trunk with forced disconnection

To perform the line test described in JJ-90-10 Appendix A.4.1 over an MGC/MG interface, the MGC shall request the MG, via the signal "bgc/brt"(A.8/Q.1950, "Basic Call Progress Tones Generator with Directionality") in an AMM command, to initiate a ringing tone. After 6 seconds the MGC shall then request the MG, via the signal "bcg/bdtq" (A.8/Q.1950), to initiate a dial tone. After reception of Modify acknowledgement, the MGC having timed the Dial Tone for a period of 10 seconds will request MG to stop Dial Tone by sending a MOD command.

...

[End Correction]

12 Technical and Editorial Corrections to H.248.20 (2002)

12.1 Media values

Description:	<p>H.248.20 defines the following for the use of the Media field in the SDP m= line:</p> <p>5.1.2 "m=line" Line for H.221 and H.223 MUX termination</p> <p><i>The syntax of the media field:</i></p> <p><i>media-field = "m=" media SP port ["/" integer] SP proto 1*(SP fmt) CRLF</i></p> <p><i>The possible media values for media are "audio", "video", "data" and "control", depending on the media type within the specific H.248.1 Stream. "Control" is used if a Stream is defined for the demultiplexed H.245 messages.</i></p> <p>Currently the IETF are working on updating the SDP RFC (see: http://www.ietf.org/internet-drafts/draft-ietf-mmusic-sdp-new-23.txt). In this text the IETF have removed the value "control" from the allowed list of media types. Thus it will soon be invalid to use value "control". It is therefore proposed to allow the use of "application" for a de-multiplexed</p>
---------------------	---

	<p>H.245 message.</p> <p>The draft defines the users of the media type “application” as:</p> <p><i>“Voice over IP, video teleconferencing, streaming media, instant messaging, etc. See also section 3 of RFC XXXX.”</i></p> <p>From this definition it is seen that type “application” would be appropriate for use for H.248.20.</p>
Reference:	AVD-2663

[Begin Correction]

5.1.2 "m=line" Line for H.221 and H.223 MUX termination

The syntax of the media field:

media-field = "m=" media SP port ["/" integer] SP proto 1*(SP fmt) CRLF

The possible media values for media are "audio", "video", "data" and "application~~control~~", depending on the media type within the specific H.248.1 Stream. "application~~Control~~" is used if a Stream is defined for the demultiplexed H.245 messages.

Note: Some older applications may use the value “control”. To aid interoperability MGs should be able to recognize “control”.

...

[End Correction]

13 Technical and Editorial Corrections to H.248.22 (2003)

13.1 Correction of typographical errors

Description:	<p>H.248.22 contains typographical errors in that the property “shrisk/srgi” is referenced however the correct property reference is “shrisk/srgir”.</p> <hr/> <p>H.248.1 defines the “Include shared risk group” property with the values “on/off”. Eg.</p> <p>5.1.1 Property Name: Include shared risk group</p> <p>PropertyID: incl, 0x0001</p> <p>Description:</p> <p style="padding-left: 40px;">The value of this property indicates if the shared risk group specified is requested to be used or to not be used (see 5.5.1.1 for further details).</p> <p>Type: Sublist of type Boolean</p> <p>Possible Values:</p> <p style="padding-left: 40px;">"on" (TRUE) Use resources from the specified SRGI only [Default]</p> <p style="padding-left: 40px;">"off" (FALSE) Use resources from any but the specified SRGI</p>
---------------------	--

	However the procedures use yes/no instead of on/off in one place. The procedures should be corrected to align with 5.1.1.
Reference:	AVD-2467 and COM 16 D-44

[Begin Correction]

5.5.1.1 Usage of the “Include shared risk group” property

The "Include shared risk group" property shall be used to indicate to the MG if resources from the specified shared risk group identity (*shrisk/srgir*) must be used for the termination (*shrisk/incl* = ~~yes~~*on*) or if resources from the specified risk group identity must not be used for the termination (*shrisk/incl* = ~~no~~*off*). If the MGC is not concerned with which shared risk groups are used then it should not include the *shrisk/incl* nor *shrisk/srgir* properties. The *shrisk/incl* and *shrisk/srgir* properties are valid only for the command that they are contained in. They cannot be read/audited after the execution of the command. Wildcarding values with CHOOSE (\$) or ALL (*) shall not be used with *shrisk/incl* and/or *shrisk/srgir*. For example: in the case of a semi-permanent connection and a protective secondary link, by specifying (*shrisk/incl* = *on*, *shrisk/srgir* = *I*) for the primary link and (*shrisk/incl* = *off*, *shrisk/srgir* = *I*) for the secondary, the MGC is assured that the primary and secondary links are not sharing the same groups of resources.

...

[End Correction]

14 Technical and Editorial Corrections to H.248.23 (2005) Corr. 1 (2006)

14.1 Clarification of default pattern ID

Description:	The Corrigendum deleted a sentence that was in direct conflict with the defined default value for the pattern parameter of the dwa signal. Some confusion as to how the pattern is applied when not included in the signal has ensued. As with any other signal, the default is applied when the parameter is absent. A notation indicating that data to be sent without alerting should use the data signal would help ease the confusion.
Reference:	COM 16-C 110

[Begin Correction]

6.3.1.1.2 Pattern

...

Description: The pattern is an abstract indication of the distinctive alerting pattern that will be applied to the line. If data is to be applied without alerting, the data signal should be used.

...

[End Correction]

14.2 Clarification of solutions to possible race condition in dwa signal

Description:	A possible, but rare, race condition for the dwa signal has been raised in a number of contributions to SG16. There is a solution that utilizes the base protocol constructs that needs documentation in the package.
Reference:	Discussions arising from COM16 C-70

[Begin Correction]

6.5 Procedures

...

For ADSI, there are constraints similar to the off-hook data with alerting around ACK digits and softkey/digit responses. Any responses that the MGC wishes to be made aware of should be requested in the Events descriptor. Digitmaps may be used for this application.

There is a possible race condition where an MGC might apply the dwa signal intending power ringing and data to be applied, but before the signal reaches the MG, the termination undergoes a transition to offhook. The result is that the signal would apply a call waiting tone in the interim while the MG reports the offhook and the MGC responds with updated Events and Signals Descriptors. It may be desirable to prevent any possibility of this race condition occurring. To achieve this, implementations may embed the andisp/dwa signal on the onhook event with its strict parameter set to "state". This will enforce that the dwa signal is only applied if the termination is still onhook when the command arrives at the MG.

Binary encoding SHALL carry the binary data. Text encoding SHALL carry the data as a hex string encoded as big-endian hex data.

...

[End Correction]

Annex: Defect Report Form for the H.248 Sub-series

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

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