

I n t e r n a t i o n a l   T e l e c o m m u n i c a t i o n   U n i o n

# ITU-T

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

# H.248.1

## Amendment 1

(05/2008)

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

---

Gateway control protocol: Version 3  
**Amendment 1: Corrections and clarifications**

Recommendation ITU-T H.248.1 (2005) –  
Amendment 1



ITU-T H-SERIES RECOMMENDATIONS  
AUDIOVISUAL AND MULTIMEDIA SYSTEMS

|                                                                               |                    |
|-------------------------------------------------------------------------------|--------------------|
| CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS                                   | H.100–H.199        |
| INFRASTRUCTURE OF AUDIOVISUAL SERVICES                                        |                    |
| General                                                                       | H.200–H.219        |
| Transmission multiplexing and synchronization                                 | H.220–H.229        |
| Systems aspects                                                               | H.230–H.239        |
| <b>Communication procedures</b>                                               | <b>H.240–H.259</b> |
| Coding of moving video                                                        | H.260–H.279        |
| Related systems aspects                                                       | H.280–H.299        |
| Systems and terminal equipment for audiovisual services                       | H.300–H.349        |
| Directory services architecture for audiovisual and multimedia services       | H.350–H.359        |
| Quality of service architecture for audiovisual and multimedia services       | H.360–H.369        |
| Supplementary services for multimedia                                         | H.450–H.499        |
| MOBILITY AND COLLABORATION PROCEDURES                                         |                    |
| Overview of Mobility and Collaboration, definitions, protocols and procedures | H.500–H.509        |
| Mobility for H-Series multimedia systems and services                         | H.510–H.519        |
| Mobile multimedia collaboration applications and services                     | H.520–H.529        |
| Security for mobile multimedia systems and services                           | H.530–H.539        |
| Security for mobile multimedia collaboration applications and services        | H.540–H.549        |
| Mobility interworking procedures                                              | H.550–H.559        |
| Mobile multimedia collaboration inter-working procedures                      | H.560–H.569        |
| BROADBAND AND TRIPLE-PLAY MULTIMEDIA SERVICES                                 |                    |
| Broadband multimedia services over VDSL                                       | H.610–H.619        |
| Advanced multimedia services and applications                                 | H.620–H.629        |
| IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV                            |                    |
| General aspects                                                               | H.700–H.719        |
| IPTV terminal devices                                                         | H.720–H.729        |

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

# **Recommendation ITU-T H.248.1**

## **Gateway control protocol: Version 3**

### **Amendment 1**

#### **Corrections and clarifications**

##### **Summary**

To achieve greater scalability, this Recommendation decomposes the H.323 Gateway function defined in ITU-T Rec. H.246 into functional subcomponents and specifies the protocols these components use to communicate. This allows implementations of H.323 gateways to be highly scalable and encourages leverage of widely deployed Switched Circuit Network (SCN) capabilities such as SS7 switches. This also enables H.323 gateways to be composed of components from multiple vendors distributed across multiple physical platforms. The purpose of this Recommendation is to add capabilities currently defined for H.323 systems and is intended to provide new ways of performing operations already supported in H.323.

This Recommendation includes several enhancements to ITU-T Rec. H.248.1 Version 2:

- capability to define context properties via packages;
- an IEPS context property;
- a flag to indicate that the MG has OutOfService terminations to report at registration time;
- new message segmentation package and procedures for non-segmenting transports;
- refined package definition requirements and a new package template;
- refined profile definition requirements and a new profile template;
- addition of statistics on a stream level;
- addition of a signal request identifier to differentiate similar signals within a SignalList;
- addition of a base signal parameter to indicate in which direction to play the signal;
- addition of two new Topology types;
- addition of an intersignal delay timer for signals in a SignalList;
- addition of a new ContextIDList construct for command responses;
- addition of a TerminationIDList construct for commands and responses;
- refined ServiceChange procedures;
- addition of a capability for the MGC to regulate the rate at which it receives notifications;
- addition of the ability to add filter conditions to audit requests.

Amendment 1 to Recommendation ITU-T H.248.1 (2005) introduces the corrections and clarifications identified in previous Implementors' Guides to Recommendation ITU-T H.248.1. It also introduces some formatting changes to enhance readability.

##### **Source**

Amendment 1 to Recommendation ITU-T H.248.1 (2005) was approved on 2 May 2008 by ITU-T Study Group 16 (2005-2008) under Recommendation ITU-T A.8 procedure.

## FOREWORD

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

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

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

## NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

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

## INTELLECTUAL PROPERTY RIGHTS

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

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2009

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

# CONTENTS

|                                                                   | <b>Page</b> |
|-------------------------------------------------------------------|-------------|
| 3 Definitions .....                                               | 1           |
| 3.1 Terms defined elsewhere .....                                 | 1           |
| 3.2 Terms defined in this Recommendation.....                     | 1           |
| 4 Abbreviations and acronyms .....                                | 2           |
| 7 Commands .....                                                  | 3           |
| 8 Transactions.....                                               | 10          |
| 9 Transport.....                                                  | 11          |
| 11.3 Negotiation of protocol version.....                         | 11          |
| 11.5 Failure of an MGC.....                                       | 11          |
| 11.7 Compatibility.....                                           | 12          |
| 12 Package definition .....                                       | 12          |
| 12.2 Guidelines to defining parameters to events and signals..... | 15          |
| A.2 ASN.1 syntax specification .....                              | 16          |
| B.2 ABNF specification .....                                      | 17          |
| B.3 Hexadecimal octet coding .....                                | 17          |
| Annex C – Tags for media stream properties .....                  | 18          |
| C.1 General media attributes.....                                 | 18          |
| C.5 Frame relay .....                                             | 18          |
| Annex E – Basic packages .....                                    | 18          |
| E.1 Generic .....                                                 | 18          |
| E.2 Base Root Package .....                                       | 19          |
| E.14 Segmentation Package.....                                    | 21          |
| E.15 Notification Behaviour .....                                 | 22          |
| Appendix III – H.248 Profile Definition template.....             | 27          |



# Recommendation ITU-T H.248.1

## Gateway control protocol: Version 3

### Amendment 1

#### Corrections and clarifications

*Modifications introduced by this amendment are shown in revision marks. Unchanged text is replaced by ellipsis (...). Some parts of unchanged text (clause numbers, etc.) may be kept to indicate the correct insertion points.*

...

### 3 Definitions

#### 3.1 Terms defined elsewhere

None.

#### 3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

**3.2.1 access gateway:** A type of gateway that provides a User-Network Interface (UNI) such as ISDN.

**3.2.2 descriptor:** A syntactic element of the protocol that groups related properties. For instance, the properties of a media flow on the MG can be set by the MGC by including the appropriate descriptor in a command.

**3.2.3 media gateway (MG):** The media gateway converts media provided in one type of network to the format required in another type of network. For example, a MG could terminate bearer channels from a switched circuit network (e.g., DS0s) and media streams from a packet network (e.g., RTP streams in an IP network). This gateway may be capable of processing audio, video and T.120 alone or in any combination, and will be capable of full duplex media translations. The MG may also play audio/video messages and perform other IVR functions, or may perform media conferencing.

**3.2.4 media gateway controller (MGC):** Controls the parts of the call state that pertain to connection control for media channels in a MG.

**3.2.5 multipoint control unit (MCU):** An entity that controls the setup and coordination of a multi-user conference that typically includes processing of audio, video and data.

**3.2.6 residential gateway:** A gateway that interworks an analogue line to a packet network. A residential gateway typically contains one or two analogue lines and is located at the customer premises.

**3.2.7 SCN FAS signalling gateway:** This function contains the SCN Signalling Interface that terminates SS7, ISDN or other signalling links where the call control channel and bearer channels are collocated in the same physical span.

**3.2.8 SCN NFAS signalling gateway:** This function contains the SCN Signalling Interface that terminates SS7 or other signalling links where the call control channels are separated from bearer channels.

**3.2.9 stream:** Bidirectional media or control flow received/sent by a media gateway as part of a call or conference.

**3.2.10 trunk:** A communication channel between two switching systems such as a DS0 on a T1 or E1 line.

**3.2.11 trunking gateway:** A gateway between SCN network and packet network that typically terminates a large number of digital circuits.

## 4 **Abbreviations and acronyms**

This Recommendation uses the following abbreviations and acronyms:

...

|             |                                     |
|-------------|-------------------------------------|
| RTP         | Real-time Transport Protocol        |
| <u>RTCP</u> | <u>RTP Control Protocol</u>         |
| <u>RTPD</u> | <u>Round-Trip Propagation Delay</u> |
| SC          | ServiceChange                       |

...

### 6.1.1 **Context attributes and descriptors**

...

- A ContextAttribute Descriptor that enables extra context attributes to be defined by using the packages extension mechanism (see 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.

...

### 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 TerminationIDs should not be contained in a single TerminationIDList.

...

### 6.2.5 **Root Termination**

...

- an AuditValue Command – to examine the values of properties and statistics implemented on Root;
- an ~~AuditCapabilities~~ AuditCapability Command – to determine what properties of Root are implemented;

...

### 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 Error Code 435 ("TerminationID is not in specified context") is returned, though Error Code 431 ("No TerminationID matched a wildcard") may be returned in order to maintain backward compatibility ~~is 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.

...

### 6.3.4 Wildcarded responses

...

If an error occurs during the execution of a wildcarded request that specifies a wildcarded response, special handling is required to provide useful information about the error(s) while still maintaining a modest sized response. When a wildcarded response is requested, all instances (as specified above) of the command shall be executed even if one or more result in errors, but later commands in the transaction will not be executed (unless optional was specified). Multiple command responses shall be returned for the command that encountered the error. The first command response shall be the normal wildcard response containing the UNION of responses for those commands that succeeded. If none of them succeeded, the UNION shall be empty. Additional command responses for each ~~TransactionID~~ TerminationID that failed shall be returned with the appropriate Error Descriptor.

...

## 7 Commands

...

- 5) **AuditValue:** The AuditValue Command returns the current state of properties, events, signals and statistics of terminations.
- 6) ~~**AuditCapabilities**~~ **AuditCapability:** The ~~AuditCapabilities~~ AuditCapability Command returns all the possible values for termination properties, events and signals allowed by the Media Gateway.

...

### 7.1.8 Local and Remote Descriptors

#### 7.1.8.1 Syntax for Local Descriptors and Remote Descriptors

The MGC uses Local and Remote Descriptors to reserve and commit MG resources for media decoding and encoding for the given stream(s) and termination to which they apply. The MG includes these descriptors in its response to indicate what it is actually prepared to support. The MG shall include additional properties and their values in its response if these properties are mandatory yet not present in the requests made by the MGC (e.g., by specifying detailed video encoding parameters where the MGC only specified the payload type).

To avoid ambiguity when requesting the MG to reserve and commit resources, the MGC should supply as much information as needed when using underspecification (i.e., CHOOSE) so that the MG can make an unambiguous selection. For example when using CHOOSE without specifying the required application type (e.g., "media name" in case of SDP encoding), further information may be needed (e.g., attribute lines in case of SDP encoding).

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

#### **7.1.8.1.1 Specific syntax for H.248 text encoding**

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:

...

#### **7.1.8.1.2 Specific syntax for H.248 binary encoding**

When binary encoding the protocol, the descriptor consists of groups of properties (tag-value pairs) as specified in Annex C. Each such group may contain the parameters of a session description.

#### **7.1.8.2 Semantics for Local Descriptor and Remote Descriptor**

Below, the semantics of the Local and Remote Descriptors are specified in detail. The specification consists of two parts. The first part specifies the interpretation of the contents of the descriptor. The second part specifies the actions the MG must take upon receiving the Local and Remote Descriptors. The actions to be taken by the MG depend on the values of the ReserveValue and ReserveGroup Properties of the LocalControl Descriptor.

##### **7.1.8.2.1 Specification depth of Local Descriptor/Remote Descriptor content**

Either the Local or the Remote Descriptor or both may be:

- unspecified (i.e., absent);
- empty;
- underspecified through use of CHOOSE in a property value;
- fully specified; or
- overspecified through presentation of multiple groups of properties and possibly multiple property values in one or more of these groups.

##### **7.1.8.2.2 Interpretation of Local Descriptor/Remote Descriptor by the MG**

Where the descriptors have been passed from the MGC to the MG, they are interpreted according to the rules given in 7.1.1, with the following additional comments for clarification:

...

##### **7.1.8.2.3 Resource reservation rules for ReserveValue and ReserveGroup properties**

Subject to the above rules, subsequent action depends on the values of the ReserveValue and ReserveGroup Properties in LocalControl.

###### **7.1.8.2.3.1 ReserveValue = "False" AND ReserveGroup = "True"**

If ReserveGroup is "True", the MG reserves the resources required to support as many as possible of the requested property group alternatives that it can currently support.

#### **7.1.8.2.3.2      ReserveValue = "True" AND ReserveGroup = "False"**

If ReserveValue is "True", the MG reserves the resources required to support as many as possible of the requested property value alternatives that it can currently support.

...

#### **7.1.8.2.3.3      ReserveValue = "True" OR ReserveGroup = "True"**

If ReserveGroup is "True" or ReserveValue is "True", then the following rules apply:

...

#### **7.1.8.2.3.4      ReserveValue = "False" AND ReserveGroup = "False"**

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.

...

### **7.1.11 Signals Descriptor**

Signals are MG-generated media such as tones and announcements as well as bearer-related signals such as hookswitch. More complex signals may include a sequence of such simple signals interspersed with and conditioned upon the receipt and analysis of media or bearer-related signals. Examples include echoing of received data as in the Basic Continuity Test-Package (clause E.10). Signals may also request preparation of media content for future signals.

...

Multiple signals and sequential signal lists in the same Signals Descriptor shall be played simultaneously.

Signals have directionality, which is identified by the direction parameter in the base syntax or by package-defined direction parameters. The direction parameter can be used to indicate the direction that the tone is to be sent. External indicates that the tone is sent from the MG to an external point. Internal indicates that the tone is played into the Context to the other terminations. Bothway indicates both internal and external behaviour.

...

### **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 on 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.

...

### 7.1.18 Topology Descriptor

...

- (*T1, T2, ~~isolate~~Isolate*) means that the terminations matching *T2* do not receive media from the terminations matching *T1*, nor vice versa.
- (*T1, T2, ~~oneway~~Oneway*) means that the terminations that match *T2* receive media from the terminations matching *T1*, but not vice versa. In this case, use of the ALL wildcard such that there are terminations that match either *T1* or *T2* *but not both* is allowed.
- (*T1, T2, ~~onewayexternal~~OnewayExternal*) means the terminations that match *T2*, receive media sent externally by terminations matching *T1*, but not vice versa. In this case, use of the ALL wildcard for *T1* is not allowed.
- (*T1, T2, ~~onewayboth~~OnewayBoth*) means the terminations that match *T2*, receive media sent and received externally by terminations matching *T1*, but not vice versa. In this case, use of the ALL wildcard for *T1* and/or *T2* is not allowed.
- (*T1, T2, ~~bothway~~Bothway*) means that the terminations matching *T2* receive media from the terminations matching *T1*, and vice versa. In this case it is allowed to use wildcards such that there are terminations that match both *T1* and *T2*. However, if there is a termination that matches both, no loopback is introduced.

CHOOSE wildcards may be used in *T1* and *T2* as well, under the following restrictions:

- the action (see clause 8) of which the Topology Descriptor is part contains an Add Command in which a CHOOSE wildcard is used;
- if a CHOOSE wildcard occurs in *T1* or *T2*, then a partial name shall not be specified.

The CHOOSE wildcard in a Topology Descriptor matches the TerminationID that the MG assigns in the first Add Command that uses a CHOOSE wildcard in the same action. An existing termination that matches *T1* or *T2* in the context to which a termination is added is connected to the newly added termination as specified by the Topology Descriptor. If a termination is not mentioned within a Topology Descriptor, any topology associated with it remains unchanged. If, however, a new termination is added into a context, its association with the other terminations within the context defaults to ~~bothway~~ Bothway, unless a Topology Descriptor is given to change this (e.g., if *T3* is added to a context with *T1* and *T2* with topology (*T3, T1, ~~oneway~~Oneway*) it will be connected bothway to *T2*).

If the topology is applied to one particular stream (*T1, T2, association, StreamID*), the topology of other streams between the terminations does not change.

A Topology Descriptor shall not include a combination of associations between two terminations (*Ti, Tj*) with and without the optional StreamID field, to avoid undefined behaviour. For example (*T1, T2, ~~bothway~~Bothway*) and (*T1, T2, ~~isolate~~Isolate, S1*) shall not appear in the same descriptor.

Upon receipt of such a Topology Descriptor, a MG shall respond with an error response, including Error Code 421 ("Unknown action or illegal combination of actions").

A oneway connection must be implemented in such a way that the other terminations in the context are not aware of the change in topology.

Figure 7, the table following it and Figure 8 following it show some examples of the effect of including Topology Descriptors in actions. In these examples it is assumed that the Topology Descriptors are applied in sequence. Figures 9 and 10 are stand-alone examples showing the specific effects of the ~~onewayexternal~~ OnewayExternal and ~~onewayboth~~ OnewayBoth topology settings.

...

| Topology | Description                                                                                                                                                                                                              |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1        | No Topology Descriptors<br>When no Topology Descriptors are included, all terminations have a bothway connection to all other terminations.                                                                              |
| 2        | <i>T1, T2 Isolate</i><br>Removes the connection between <i>T1</i> and <i>T2</i> . <i>T3</i> has a bothway connection with both <i>T1</i> and <i>T2</i> . <i>T1</i> and <i>T2</i> have bothway connections to <i>T3</i> . |
| 3        | <i>T3, T2 <del>oneway</del>Oneway</i><br>An oneway connection from <i>T3</i> to <i>T2</i> (i.e., <i>T2</i> receives media flow from <i>T3</i> ). A bothway connection between <i>T1</i> and <i>T3</i> .                  |
| 4        | <i>T2, T3 <del>oneway</del>Oneway</i><br>An oneway connection between <i>T2</i> to <i>T3</i> . <i>T1</i> and <i>T3</i> remain bothway connected                                                                          |
| 5        | <i>T2, T3 <del>bothway</del>Bothway</i><br><i>T2</i> is bothway connected to <i>T3</i> . This results in the same as Topology 2.                                                                                         |
| 6        | <i>T1, T2 <del>bothway</del>Bothway (T2, T3 <del>bothway</del>Bothway and T1, T3 <del>bothway</del>Bothway may be implied or explicit).</i> All terminations have a bothway connection to all other terminations.        |

...

**Figure 9 – ~~Onewayexternal~~ OnewayExternal contrasted with ~~oneway~~ Oneway topology**

...

**Figure 10 – Operation of ~~onewayboth~~ OnewayBoth topology**

#### 7.1.19 ContextAttribute Descriptor

...

A new setting of the ContextAttribute 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 with the unwanted information stripped out. The inclusion of a ContextAudit or a ContextAttribute Descriptor in an action that contains only an AuditValue or ~~AuditCapabilities~~ AuditCapability Command does not constitute a new setting.

...

#### 7.2.1 Add

...

Add shall not be used on a termination with a ~~serviceState~~ServiceStates of "OutOfService".

...

#### 7.2.4 Move

...

The Move Command does not affect the properties/signals/events/statistics of the termination on which it operates, except those properties explicitly modified by descriptors included in the Move Command. As such, unless the signals descriptor is modified, the playout of signals continues. The Audit Descriptor with a Statistics Descriptor, for example, would return statistics on the termination just prior to the Move. Possible descriptors returned from Move are the same as for Add.

For convenience, if a multiplexing termination is the object of a Move Command, then any bearer terminations listed in its Multiplex Descriptor are also moved as if individual Move Commands listing the terminations were invoked.

Move shall not be used on a termination with a ~~serviceState~~ServiceStates of "OutOfService".

...

#### 7.2.6 ~~AuditCapabilities~~AuditCapability

The ~~AuditCapabilities~~AuditCapability Command returns the possible values of properties, events, signals and statistics associated with terminations. An ~~AuditCapabilities~~AuditCapability may be requested for the contents of a descriptor or for a single property, event, signal or statistic.

...

[,StatisticsDescriptor]

~~AuditCapabilities~~AuditCapability(TerminationIDList,  
AuditDescriptor)

...

If a descriptor audit is requested, the appropriate descriptors, with the possible values for the termination, are returned from ~~AuditCapabilities~~AuditCapability. Descriptors may be repeated where there are multiple possible values.

If a wildcarded response is requested, only one command return is generated, with the contents containing the union of the values of all terminations in the list or matching the wildcard. This convention may reduce the volume of data required to audit a group of terminations.

If a property, signal, event or statistic is audited, the appropriate properties, signals, events and statistics with the capabilities of the termination, are returned from ~~AuditCapabilities~~AuditCapability.

...

The following illustrates other information that can be obtained with the ~~AuditCapabilities~~AuditCapability Command:

...

#### 7.2.8.1.1 ServiceChangeMethod

The ServiceChangeMethod parameter specifies the type of ServiceChange that will or has occurred:

- 1) Graceful: indicates that the specified terminations will be taken out of service after the specified ServiceChangeDelay; established connections are not yet affected, but the Media Gateway Controller should refrain from establishing new connections and should attempt to gracefully tear down existing connections on the termination(s) affected by the ServiceChange Command. The MG should set the termination's ~~serviceState~~ServiceStates Property at the expiry of ServiceChangeDelay or the removal of the termination from an active context (whichever is first), to "~~out-OutOf-s~~Service".
- 2) Forced: indicates that the specified terminations were taken abruptly out of service and any established connections associated with them may be lost. For non-Root terminations, the MGC is responsible for cleaning up the context (if any) with which the failed termination is associated. At a minimum, the termination shall be subtracted from the context. The termination's ~~serviceState~~ServiceStates Property should be "~~out-OutOf-s~~Service". For the Root Termination, the MGC can assume that all connections are lost on the MG and thus can consider that all the terminations have been subtracted.
- 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~~"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.

...

#### 7.2.8.1.3 ServiceChangeAddress and ServiceChangeMgcID

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.

#### 7.2.8.1.4 ServiceChangeDelay

The optional ServiceChangeDelay parameter is expressed in seconds. If the delay is absent or set to zero, the delay value should be considered to be null. In the case of a "Graceful" ServiceChangeMethod, a null delay indicates that the Media Gateway Controller should wait for the natural removal of existing connections and should not establish new connections. For "Graceful" only, a null delay means the MG must not set ~~ServiceState~~ServiceStates "OutOfService" until the termination is in the NULL Context.

...

### 7.2.8.1.8 ExtensionParameter

...

$X\text{-SC} = 1 * (\text{NAME EQUAL paramValue [COMMA]})$

NOTE – When multiple NAME EQUAL paramValue constructs are used, the COMMA shall be used to separate them.

...

### 7.2.8.1.11 ServiceChange Command and Response

...

- 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) ~~consider the registration stage completed and await further commands from the MGC; keep the control association active, so that~~ 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.

...

## 7.2.9 Manipulating and auditing context attributes

...

The following illustrates information that can be obtained with a Context Audit:

| ContextID | TerminationID | AuditValue | <del>AuditCapabilities</del><br><u>AuditCapability</u> |
|-----------|---------------|------------|--------------------------------------------------------|
| ...       |               |            |                                                        |

...

## 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.

...

## 9 Transport

The transport mechanism for the protocol should allow the reliable transport of transactions between a MGC and MG. The transport shall remain independent of what particular commands are being sent, and shall be applicable to all application states. There are several transports defined for the protocol, which are defined in annexes to this Recommendation and other H.248 Subseries Recommendations (e.g., H.248.4 and H.248.5). Additional transports may be defined as additional Recommendations in the H.248 subseries Recommendations. For transport of the protocol over IP, MGCs shall implement both TCP and UDP/~~ALF~~,ALF; a MG shall implement TCP or UDP/ALF or both.

...

### 11.3 Negotiation of protocol version

...

If the MGC supports the version indicated by the MG, both the MGC and MG ~~it~~ 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").

Protocol version negotiation may also occur at "Handoff" and "Failover" ServiceChanges.

For rules regarding protocol extension see clause 11.7.

~~When extending the protocol with new versions, the following rules should be followed:~~

- ~~1) Existing protocol elements, i.e., procedures, parameters, descriptor, property, values, should not be changed unless a protocol error needs to be corrected or it becomes necessary to change the operation of the service that is being supported by the protocol.~~
- ~~2) The semantics of a command, a parameter, a descriptor, a property, or a value should not be changed.~~
- ~~3) Established rules for formatting and encoding messages and parameters should not be modified.~~
- ~~4) When information elements are found to be obsolete they can be marked as not used. However, the identifier for that information element will be marked as reserved. In that way it cannot be used in future versions.~~

...

### 11.5 Failure of an MGC

...

In partial failure, or for manual maintenance reasons, an MGC may wish to direct its controlled MGs to use a different MGC. To do so, it sends a ServiceChange ~~method~~ command to the MG with a "Handoff" method, and its designated replacement in ServiceChangeMgcID. If "Handoff" is supported, the MG shall send a ServiceChange message with a "Handoff" method and a "MGC directed change" reason to the designated MGC. If it fails to get a reply from the designated MGC, the MG shall behave as if its MGC failed, and start contacting secondary MGCs as specified in the previous paragraph. If the MG is unable to establish a control relationship with any MGC, it shall wait a random amount of time as described in 9.2 and then start contacting its primary, and if necessary, its secondary MGCs again.

...

## **11.7 Compatibility**

### **11.7.1 Compatibility rules – forward and backward compatibility**

H.248, through its versioning and package extension capabilities, allows forward compatibility. As per clause 12.3 of ITU-T Rec. Q.1400, forward compatibility mechanisms are defined as a scheme to enable a version of a protocol to communicate effectively with and interwork with future versions of the protocol. That is, a version of a protocol should not restrict future protocols from providing extra capabilities.

When extending the H.248 protocol, backwards compatibility shall be maintained. As per clause 12.3 of ITU-T Rec. Q.1400, backward compatibility rules are defined as a scheme to ensure that nodes implementing future versions of the protocol will be able to send protocol messages of the previous version which will be understood and fully processed by the node supporting the previous version. That is, future versions of a protocol must allow earlier versions to operate with it and not reduce the earlier version's service level, i.e., a version 1 message with the version removed shall be able to be sent with a version 3 protocol with exactly the same meaning.

The H.248-specific backward compatibility rules that should be followed when extending the protocol (including packages and profiles) with new versions, are as follows:

- 1) Existing protocol elements, i.e., procedures, parameters, descriptor, property, values, should not be changed unless a protocol error needs to be corrected or it becomes necessary to change the operation of the service that is being supported by the protocol.
- 2) The semantics of a command, a parameter, a descriptor, a property, or a value should not be changed.
- 3) Established rules for formatting and encoding messages and parameters should not be modified.
- 4) When information elements are found to be obsolete, they can be marked as not used. However, the identifier for that information element will be marked as reserved. In that way, it cannot be used in future versions.

### **11.7.2 Compatibility mechanism**

Protocol, package and profile versioning are integral parts of the H.248 architecture. The versioning of protocol elements together with the concept of the H.248 control association permits effective compatibility mechanisms.

#### **11.7.2.1 Aspect of H.248 Control Association**

Negotiated protocol capabilities are used for the lifetime of the H.248 control association. An initial ServiceChange registration message is encoded as protocol version 1 (See clause 11.3); however, the final protocol version used is negotiated and may be the subject of determination/negotiation of one or more profiles.

#### **11.7.2.2 Capability Change or Service Change on existing H.248 Control Associations**

A capability upgrade, leading to version negotiation procedures, may lead to a re-newing or re-establishment of the control association. This compatibility mechanism allows, principally, the replacement of version N protocol, package(s) and/or profile(s) by a lower or higher version N-K or N+M.

## **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.

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

The package definition:

- 1) should be aA 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.

...

### 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-; cCould 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 in 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 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. 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 in B.2.

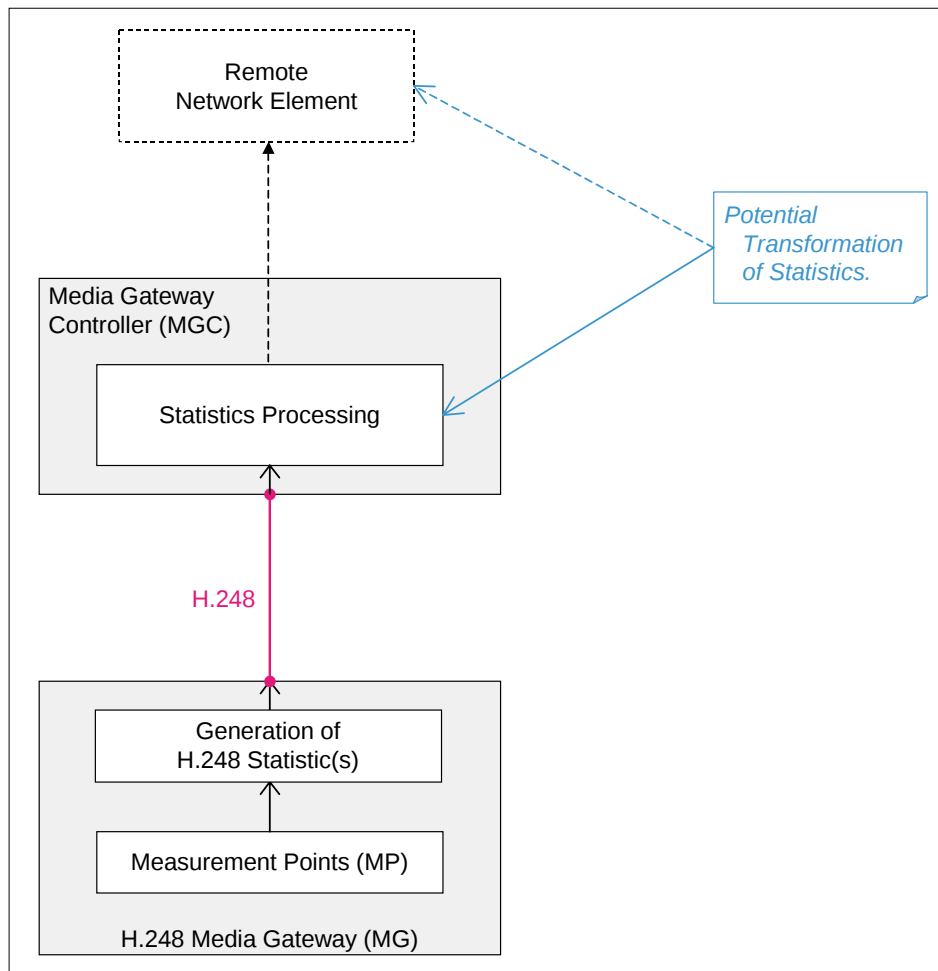
**Possible values:**

A package must indicate the unit of measure, e.g., milliseconds, packets, either here or along with the type above, as well as indicating any restriction on the range.

**Level:** Specify if the statistic can be kept at the termination level, Stream level or either.

**12.1.5.1 Aspects of statistics transformation**

H.248 statistics are generated by Media Gateways. Such statistics are either based on MG local measurements, that is, the MG is the measurement point, or the statistics are based on remote measurements from the MG perspective. The reported statistics by the MG could be transformed in some specific cases, e.g., by the served user of the statistics (see Figure 12).



**Figure 12 – Potential location of statistics transformation**

Transformation of statistics is applicable to certain categories of statistics (e.g., statistics related to traffic-volume and traffic-rate metrics), and then only in very specific scenarios. Typically, transformation rules are only applied when a set of conditions is fulfilled. These conditions are generally proscribed by the bearer technology, protocol stack and other service aspects related to the H.248 stream or termination.

By defining statistics explicitly related to the layer or condition that is desired to be collected, post-collection transformation of statistics can be avoided.

...

## 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

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. 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.

...

## 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
    {
        onewayexternal(0),
        onewayboth(1),
        ...
    } OPTIONAL
}
-- if present, topologyDirectionExtension takes precedence over
-- topologyDirection
```

```
AmmRequest ::= SEQUENCE
{
    terminationID         TerminationIDList,
    descriptors            SEQUENCE OF AmmDescriptor,
    -- At most one descriptor of each type (see AmmDescriptor)
    -- allowed in the sequence.
    ...
}
```

...

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

...

## B.2 ABNF specification

...

```
; at-most-once, and DigitMapToken and PackagesToken are not allowed
; in AuditCapabilitiesAuditCapability command
auditItem = auditReturnItem / SignalsToken / EventBufferToken /
           EventsToken / indAudterminationAudit
```

...

```
indAudterminationStateDescriptor
    = TerminationStateToken LBRKT
      indAudterminationStateParm RBRKT

/at-most-once-per-item
indAudterminationStateParm
    = pkgdName / propertyParm / ServiceStatesToken
      [(EQUAL/INEQUAL) serviceStatesValue ] / BufferToken
; When values are included a Select operation is implied.
; AND/OR logic is specified at context level.
```

...

## B.3 Hexadecimal octet coding

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

NOTE – The above encoding results in the reversal of bits from the original octet.

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 clause B.2

...

## Annex C

### Tags for media stream properties

...

When a type is smaller than one octet, the value shall be stored in the low-order bits of an octet string of size one octet. 0-padding should be used for the remaining high-order bits.

#### C.1 General media attributes

| PropertyID         | Property tag | Type             | Value                                |
|--------------------|--------------|------------------|--------------------------------------|
| Media              | 1001         | Enumeration      | Audio(0), Video(1), Data(2)          |
| Transmission-mode  | 1002         | Enumeration      | Send(0), Receive(1), Send&Receive(2) |
| Number-of Channels | 1003         | Unsigned integer | 0-255                                |
| Sampling-rate      | 1004         | Unsigned integer | 0-2 <sup>32</sup>                    |
| ...                |              |                  |                                      |

...

#### C.5 Frame relay

| PropertyID           | Property tag | Type             | Value                                         |
|----------------------|--------------|------------------|-----------------------------------------------|
| DLCI                 | 5001         | Unsigned integer | Data link connection ID                       |
| CID                  | 5002         | Unsigned integer | sub-channel ID                                |
| SID/Noiselevel       | 5003         | Unsigned integer | silence insertion descriptor                  |
| Primary-Payload type | 5004         | Unsigned integer | Primary Payload Type<br>Covers FAX and codecs |

...

## Annex E

### Basic packages

This annex contains definitions of some packages for use with this Recommendation.

#### E.1 Generic

...

**Extends:** None

Version 2 of the Generic Package requires at least version 3 of the protocol.

...

## E.2 Base Root Package

...

**Extends:** None

Version 2 of the Base Root Package requires at least version 2 of the protocol.

### E.2.1 Properties

#### E.2.1.1 Maximum Number of Contexts

**Property Name:** MaxNrOfContexts

**PropertyID:** maxNumberOfContexts (0x0001)

**Description:** The value of this property gives the maximum number of contexts that can exist at any time. The NULL Context is not included in this number.

The maximum number of contexts may be calculated by a MG in different ways, e.g., dependent on the number of connection models to be supported, dependent on applied stream, termination and/or context admission control functions, or specific, MG-internal QoS support mechanisms (e.g., best-effort versus over-provisioning models), or other reasons.

The value of the property must be therefore interpreted as a best-case figure. Requests for additional contexts over and above this number may receive an error response.

...

#### E.4.2.2 End Tone Detected

...

**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 property tl is set to "wild card", any tone id is detected.

...

## E.10.5 Procedures

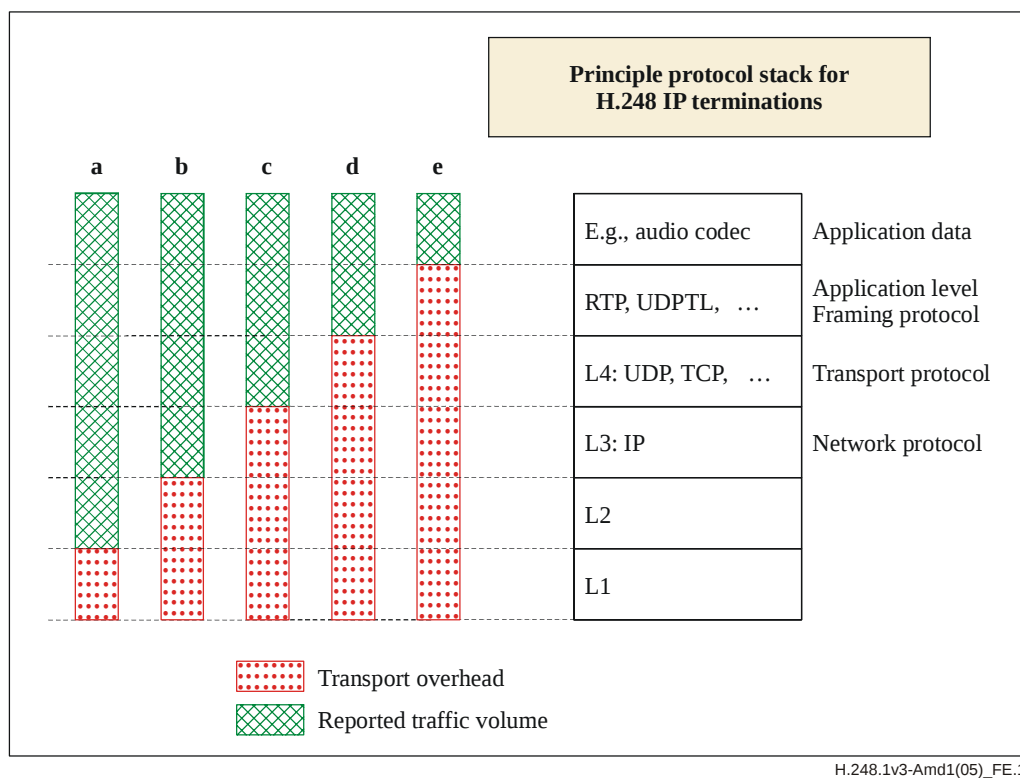
...

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.

...

#### 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 – Possible transport overhead calculations for IP-based networks**

...

#### E.12.4.3 Packet Loss

...

**Possible values:** a 32-bit whole number and a 32-bit fraction.

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 a 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".

**Level:** Either

...

#### 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 and receiver reports.

**Type:** Double

**Possible values:** any 64-bit integer 0 and up

**Level:** Either

#### 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.

...

#### E.14 Segmentation Package

**Package Name:** Segmentation Package

**PackageID:** seg (0x00A3)

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

**Version:** 1

**Extends:** root version 2

Version 2 of the Segmentation Package requires at least version 3 of the protocol.

...

#### 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

...

When segmenting a transaction reply, the sender shall ensure that the message contains complete commands and/or actions. Each segment is indicated by additional segment information at the transaction level. Each segment shall use the same ~~transactionID~~ TransactionID, and shall not repeat commands or completed actions in subsequent segments. Each segment shall be numbered serially starting at 1 and continuing through the last segment, which is denoted by inclusion of the SegmentationCompleteToken. In all cases, the segmented messages shall be syntactically valid constructs. The receiver replies to each segment in turn, utilizing the segmentation response. Because the segments are complete messages in themselves, the receiving entity does not need to wait for further segments before processing any particular segment.

...

### 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=16/1{C=1{AV=term1{...}, AV=term2{...}}}
Receiver:    !/3 [12.34.56.79]:2944 SM=16/1
Sender:      !/3 [12.34.56.78]:2944 P=16/4/##/&{C=3{AV=term5{...}}}
Receiver:    !/3 [12.34.56.79]:2944 SM=16/4/##/&
/* Segmentation Timer Expires */
Receiver:    !/3 [12.34.56.79]:2944 ER=459{"6,2,3"}
```

## 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 Commands. This package has an interaction with the NotifyBehaviour flag described in 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 21 of the Notification Behaviour Package requires at least version 3 of the protocol.

...

## 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.

...

#### **F.4.1.2 ServiceChange Method Behaviour on Physical Terminations**

...

- 3) Graceful – When sent by the MG, it indicates that the termination(s) is going OutOfService after the ServiceChangeDelay. When sent by the MGC, the MG shall take the termination(s) OutOfService at the end of the ServiceChangeDelay period. The ServiceStates Property shall be set to "OutOfService" upon expiry of the ServiceChangeDelay or when the termination(s) is removed from an active context (whichever is first) and the MGC is responsible for cleaning up any contexts or resources associated with the termination(s). Using a ServiceChangeDelay equal to zero or an absent ServiceChangeDelay indicates that the termination shall go OutOfService when it is removed from context through subtraction. Terminations already in the NULL Context shall go out of service immediately. The MGC shall not use the indicated termination(s) for connection until the Graceful is cancelled or the termination is brought back into service by a subsequent ServiceChange Command. 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 918 ("Cancel Graceful").

...

#### **F.4.1.3 ServiceChange Method Behaviour on Ephemeral Terminations**

...

- 3) 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".

...

#### F.5.4 ServiceChangeAddress

The use of the ServiceChangeAddress parameter is described in clause 7.2.8. ~~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.

#### F.5.5 ServiceChangeProfile

...

#### F.5.7 ServiceChangeMgcID

The use of the ServiceChangeMgcID parameter is described in clause 7.2.8, specifically clause 7.2.8.1.3. 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 ServiceChange Handoff Command from the MG's primary MGC, the MG shall utilize the procedures outlined in 11.5.

The MGC should not use the ServiceChangeMgcID Parameter in a ServiceChange Reply when responding to a ServiceChange Forced or Graceful Command with reason 908.

#### F.5.8 TimeStamp

The use of the optional TimeStamp parameter is described in clause 7.2.8. The TimeStamp parameter has no effect on the execution of ServiceChange Commands, but may be of use to the receiver of the ServiceChange Command for other purposes, such as billing or timing coordination.

...

#### 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} } }
}
```

...

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
  }
}
}

MG1 to MGC:
MEGACO/3 [124.124.124.222]:55555
Reply = 10005 {
    Context = 2000 {Modify = A4444, Modify = A4445}
}

```

...

## 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

```

```

    }

```

```

}

```

```

}

```

...

## Appendix III

### H.248 Profile Definition template

...

*Is "notifyCompletion" supported? What types are supported? Is the RequestID used with "NotifyCompletion"?*

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

...

#### **6.7.10 TerminationState Descriptor**

*What values of TerminationState are supported?*

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

*For other package-defined Termination State Properties see clause 6.14.*

...

#### **6.8.6 ~~AuditCapabilities~~AuditCapability**

...

#### **6.8.7 Notify**

*Which descriptors can be used in a Notify Command?*

|                                                                |                         |
|----------------------------------------------------------------|-------------------------|
| <b>Descriptors used by Notify Request <del>or Reply</del>:</b> | <ObservedEvents, Error> |
|----------------------------------------------------------------|-------------------------|

...

#### **6.8.8 ServiceChange**

...

*Which version of ITU-T Rec. H.248.1 is used by ServiceChangeVersion? The lowest value here should be the minimum version defined in 6.3.*

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

Is the ServiceChangeProfile parameter mandatory?

|                                               |                       |
|-----------------------------------------------|-----------------------|
| <b><u>ServiceChangeProfile mandatory:</u></b> | <u>&lt;Yes/No&gt;</u> |
|-----------------------------------------------|-----------------------|

...

## 6.9 Generic command syntax and encoding

*Specifies what encodings are supported by the profile.*

|                                                                            |                                              |
|----------------------------------------------------------------------------|----------------------------------------------|
| <b>Supported encodings:</b>                                                | <u>&lt;Text and Binary, Binary, Text&gt;</u> |
| <b><u>If binary encoding, is indefinite length encoding supported:</u></b> | <u>&lt;Yes, No&gt;</u>                       |

## 6.10 Transactions

...

*Can commands be marked "Optional"? Describe.*

|                                               |                                                                                                   |
|-----------------------------------------------|---------------------------------------------------------------------------------------------------|
| <b>Commands able to be marked "Optional":</b> | <u>&lt;Add, Modify, Move, Subtract, Auditvalue, Auditcapability, Servicechange, All, None&gt;</u> |
|-----------------------------------------------|---------------------------------------------------------------------------------------------------|

Can commands be marked for wildcarded response?

|                                                          |                                                                                                   |
|----------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| <b><u>Wildcarded responses may be requested for:</u></b> | <u>&lt;Add, Modify, Move, Subtract, Auditvalue, Auditcapability, Servicechange, All, None&gt;</u> |
|----------------------------------------------------------|---------------------------------------------------------------------------------------------------|

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

|                                                                 |                                                                            |
|-----------------------------------------------------------------|----------------------------------------------------------------------------|
| <b><u>Procedures that make use of wildcarded responses:</u></b> | <u>&lt;Describe the procedures making use of wildcarded responses.&gt;</u> |
|-----------------------------------------------------------------|----------------------------------------------------------------------------|

...

## 6.14 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, ...> | <u>&lt;Describe&gt;</u>            |

*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>            | <u>&lt;Describe&gt;</u>            |

...

#### 6.14.x <Package Name>

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

Additional restrictions may be tabulated as the user desires.

#### 6.15 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>                |
| <p><i><u>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.</u></i></p> <p><i><u>Discuss in detail the usage scenarios of the mandatory information elements.</u></i></p> |                    |                           |

## 6.16 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>            |
| <p><i><u>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.</u></i></p> <p><i><u>Discuss in detail the usage scenarios of the mandatory information elements.</u></i></p> |                    |                           |                       |

...



## SERIES OF ITU-T RECOMMENDATIONS

|                 |                                                                                             |
|-----------------|---------------------------------------------------------------------------------------------|
| Series A        | Organization of the work of ITU-T                                                           |
| Series D        | General tariff principles                                                                   |
| Series E        | Overall network operation, telephone service, service operation and human factors           |
| Series F        | Non-telephone telecommunication services                                                    |
| Series G        | Transmission systems and media, digital systems and networks                                |
| <b>Series H</b> | <b>Audiovisual and multimedia systems</b>                                                   |
| Series I        | Integrated services digital network                                                         |
| Series J        | Cable networks and transmission of television, sound programme and other multimedia signals |
| Series K        | Protection against interference                                                             |
| Series L        | Construction, installation and protection of cables and other elements of outside plant     |
| Series M        | Telecommunication management, including TMN and network maintenance                         |
| Series N        | Maintenance: international sound programme and television transmission circuits             |
| Series O        | Specifications of measuring equipment                                                       |
| Series P        | Telephone transmission quality, telephone installations, local line networks                |
| Series Q        | Switching and signalling                                                                    |
| Series R        | Telegraph transmission                                                                      |
| Series S        | Telegraph services terminal equipment                                                       |
| Series T        | Terminals for telematic services                                                            |
| Series U        | Telegraph switching                                                                         |
| Series V        | Data communication over the telephone network                                               |
| Series X        | Data networks, open system communications and security                                      |
| Series Y        | Global information infrastructure, Internet protocol aspects and next-generation networks   |
| Series Z        | Languages and general software aspects for telecommunication systems                        |