



**Universal Mobile Telecommunications System (UMTS);
LTE;
5G;
IP Multimedia Subsystem (IMS);
Multimedia telephony;
Media handling and interaction
(3GPP TS 26.114 version 17.9.0 Release 17)**



Reference

RTS/TSGS-0426114vh90

Keywords

5G,LTE,UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from the
[ETSI Search & Browse Standards application](#).

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format on [ETSI deliver repository](#).

Users should be aware that the present document may be revised or have its status changed, this information is available in the [Milestones listing](#).

If you find errors in the present document, please send your comments to the relevant service listed under [Committee Support Staff](#).

If you find a security vulnerability in the present document, please report it through our [Coordinated Vulnerability Disclosure \(CVD\)](#) program.

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2025.
All rights reserved.

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the [ETSI IPR online database](#).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

Legal Notice

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found at [3GPP to ETSI numbering cross-referencing](#).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

| | |
|---|----|
| Intellectual Property Rights | 2 |
| Legal Notice | 2 |
| Modal verbs terminology..... | 2 |
| Foreword..... | 17 |
| Introduction | 17 |
| 1 Scope | 18 |
| 2 References | 18 |
| 3 Definitions and abbreviations..... | 26 |
| 3.1 Definitions | 26 |
| 3.2 Abbreviations | 28 |
| 4 System description | 30 |
| 4.1 System | 30 |
| 4.2 Client | 31 |
| 4.3 MRFP and MGW | 32 |
| 5 Media codecs..... | 32 |
| 5.1 Media components | 32 |
| 5.2 Codecs for MTSI clients in terminals..... | 33 |
| 5.2.1 Speech..... | 33 |
| 5.2.1.1 General codec requirements | 33 |
| 5.2.1.2 Detailed codec requirements, AMR | 34 |
| 5.2.1.3 Detailed codec requirements, AMR-WB | 34 |
| 5.2.1.4 Detailed codec requirements, EVS..... | 34 |
| 5.2.1.5 Offering multiple audio bandwidths and multiple channels..... | 34 |
| 5.2.1.6 Codec preference order | 34 |
| 5.2.2 Video | 35 |
| 5.2.3 Real-time text..... | 36 |
| 5.2.4 Still Images | 37 |
| 6 Media configuration | 37 |
| 6.1 General | 37 |
| 6.2 Session setup procedures..... | 37 |
| 6.2.1 General..... | 37 |
| 6.2.1a RTP profile negotiation | 38 |
| 6.2.1a.1 General | 38 |
| 6.2.1a.2 Using SDPCapNeg in SDP offer..... | 38 |
| 6.2.1a.3 Answering to an SDP offer using SDPCapNeg | 39 |
| 6.2.2 Speech..... | 39 |
| 6.2.2.1 General | 39 |
| 6.2.2.2 Generating SDP offers | 40 |
| 6.2.2.3 Generating SDP answer | 42 |
| 6.2.3 Video | 51 |
| 6.2.3.1 Introduction..... | 51 |
| 6.2.3.2 Common Session Setup Procedures..... | 52 |
| 6.2.3.3 Coordination of Video Orientation (CVO) | 53 |
| 6.2.3.4 Video Region-of-Interest (ROI)..... | 53 |
| 6.2.3.5 RTP Retransmission..... | 56 |
| 6.2.3.6 Forward Error Correction (FEC)..... | 56 |
| 6.2.4 Text..... | 56 |
| 6.2.5 Bandwidth negotiation | 57 |
| 6.2.5.1 General | 57 |
| 6.2.5.2 Speech | 59 |
| 6.2.5.3 Video..... | 61 |
| 6.2.6 The Synchronization Info attribute "3gpp_sync_info" | 62 |

| | | |
|-----------|--|----|
| 6.2.7 | Negotiated QoS parameters | 63 |
| 6.2.7.1 | General | 63 |
| 6.2.7.2 | Alignment of negotiated QoS parameters and b=AS bandwidth modifier | 63 |
| 6.2.7.3 | Alignment of negotiated QoS parameters and a=bw-info attribute | 64 |
| 6.2.7.4 | The a=3gpp-qos-hint SDP attribute | 64 |
| 6.2.7.4.1 | General | 64 |
| 6.2.7.4.2 | 3gpp-qos-hint ABNF syntax and semantics | 65 |
| 6.2.7.4.3 | Creating an SDP offer | 66 |
| 6.2.7.4.4 | Creating an SDP answer | 66 |
| 6.2.7.4.5 | Offerer receiving an SDP answer | 67 |
| 6.2.8 | Delay Budget Information (DBI) Signaling | 68 |
| 6.2.9 | ANBR Support attribute "anbr" | 69 |
| 6.2.10 | Data channel | 69 |
| 6.2.10.1 | General | 69 |
| 6.2.10.2 | Generating SDP offer | 72 |
| 6.2.10.3 | Generating SDP answer | 73 |
| 6.2.10.4 | Receiving SDP answer | 73 |
| 6.2.11 | Still Images | 73 |
| 6.3 | Session control procedures | 74 |
| 7 | Data transport | 74 |
| 7.1 | General | 74 |
| 7.2 | RTP profiles | 74 |
| 7.3 | RTCP usage | 74 |
| 7.3.1 | General | 74 |
| 7.3.2 | Speech | 75 |
| 7.3.3 | Video | 76 |
| 7.3.4 | Real-time text | 77 |
| 7.3.5 | Void | 77 |
| 7.3.6 | Reduced-Size RTCP | 77 |
| 7.3.7 | Video Region-of-Interest (ROI) Signaling | 78 |
| 7.3.8 | Delay Budget Information (DBI) Signaling | 81 |
| 7.4 | RTP payload formats for MTSI clients | 83 |
| 7.4.1 | General | 83 |
| 7.4.2 | Speech | 83 |
| 7.4.3 | Video | 84 |
| 7.4.4 | Real-time text | 85 |
| 7.4.5 | Coordination of Video Orientation | 85 |
| 7.4.6 | RTP Retransmission | 87 |
| 7.4.7 | Forward Error Correction (FEC) | 87 |
| 7.4.8 | Still Images | 87 |
| 7.5 | Media flow | 87 |
| 7.5.1 | General | 87 |
| 7.5.2 | Media specific | 87 |
| 7.5.2.1 | Speech | 87 |
| 7.5.2.1.1 | General | 87 |
| 7.5.2.1.2 | Default operation | 88 |
| 7.5.2.1.3 | HSPA | 89 |
| 7.5.2.1.4 | EGPRS | 89 |
| 7.5.2.1.5 | GIP | 89 |
| 7.5.2.1.6 | Initial codec mode for AMR and AMR-WB | 89 |
| 7.5.2.1.7 | E-UTRAN and NR | 90 |
| 7.5.2.1.8 | Initial codec mode for EVS | 90 |
| 7.5.2.1.9 | Dual-mono | 90 |
| 7.5.2.2 | Video | 90 |
| 7.5.2.3 | Text | 91 |
| 7.5.3 | Media synchronization | 91 |
| 7.5.3.1 | General | 91 |
| 7.5.3.2 | Text | 91 |
| 7.5.4 | ECN usage in RTP sessions | 91 |
| 7.5.5 | Handling of bit-rate variations | 91 |
| 7.5.5.1 | General | 91 |

| | | |
|-----------|---|-----|
| 7.5.5.2 | Video..... | 92 |
| 7.5.5.3 | Speech..... | 92 |
| 8 | Jitter buffer management in MTSI clients in terminals..... | 92 |
| 8.1 | General..... | 92 |
| 8.2 | Speech..... | 92 |
| 8.2.1 | Terminology..... | 92 |
| 8.2.2 | Functional requirements for jitter-buffer management..... | 94 |
| 8.2.3 | Minimum performance requirements for jitter-buffer management..... | 94 |
| 8.2.3.1 | General..... | 94 |
| 8.2.3.2 | Objective performance requirements..... | 95 |
| 8.2.3.2.1 | General..... | 95 |
| 8.2.3.2.2 | Jitter buffer delay criteria..... | 95 |
| 8.2.3.2.3 | Jitter induced concealment operations..... | 96 |
| 8.2.3.3 | Delay and error profiles..... | 96 |
| 8.2.3.4 | Speech material for JBM minimum performance evaluation..... | 97 |
| 8.3 | Video..... | 97 |
| 8.4 | Text..... | 97 |
| 9 | Packet-loss handling..... | 97 |
| 9.1 | General..... | 97 |
| 9.2 | Speech..... | 98 |
| 9.2.1 | General..... | 98 |
| 9.2.2 | Transmitting redundant frames..... | 98 |
| 9.2.3 | Receiving redundant frames..... | 101 |
| 9.3 | Video..... | 101 |
| 9.3.1 | General..... | 101 |
| 9.3.2 | Receiver behaviour..... | 101 |
| 9.3.3 | Sender behaviour..... | 102 |
| 9.3.4 | Recommendations for packet loss recovery mechanisms usage..... | 102 |
| 9.4 | Text..... | 103 |
| 10 | Adaptation..... | 103 |
| 10.1 | General..... | 103 |
| 10.2 | Speech..... | 104 |
| 10.2.0 | General..... | 104 |
| 10.2.1 | RTCP-APP with codec control requests..... | 105 |
| 10.2.1.1 | General..... | 105 |
| 10.2.1.2 | General Format of RTCP-APP packet with codec control requests..... | 105 |
| 10.2.1.2a | Padding..... | 106 |
| 10.2.1.3 | Redundancy Request..... | 107 |
| 10.2.1.4 | Frame Aggregation Request..... | 107 |
| 10.2.1.5 | Codec Mode Request..... | 108 |
| 10.2.1.6 | Generation of RTP payloads for multiple codec control requests..... | 108 |
| 10.2.1.7 | EVS Primary Rate Request..... | 109 |
| 10.2.1.8 | EVS Bandwidth Request..... | 110 |
| 10.2.1.9 | EVS Channel Aware Request..... | 111 |
| 10.2.1.10 | EVS Primary mode to EVS AMR-WB IO mode Switching Request..... | 111 |
| 10.2.1.11 | EVS AMR-WB IO mode to EVS Primary mode Switching Request..... | 112 |
| 10.2.2 | Example use cases..... | 113 |
| 10.2.3 | SDP negotiation for RTCP-APP..... | 115 |
| 10.3 | Video..... | 116 |
| 10.3.1 | General..... | 116 |
| 10.3.2 | Signaling mechanisms..... | 116 |
| 10.3.3 | Adaptation triggers..... | 117 |
| 10.3.4 | Sender behavior, downswitching..... | 118 |
| 10.3.4.1 | Downswitching divided into phases..... | 118 |
| 10.3.4.2 | Rate reduction phase..... | 118 |
| 10.3.4.3 | Delay recovery phase..... | 119 |
| 10.3.5 | Sender behavior, up-switching..... | 119 |
| 10.3.6 | Receiver behavior, down-switching..... | 120 |
| 10.3.7 | Receiver behavior, up-switch..... | 120 |
| 10.3.8 | ECN triggered adaptation..... | 120 |

| | | |
|------------|---|-----|
| 10.4 | Text | 121 |
| 10.5 | Explicit Congestion Notification | 121 |
| 10.6 | Using the a=bw-info attribute for adaptation | 122 |
| 10.7 | Access network bitrate recommendation | 122 |
| 10.7.1 | General | 122 |
| 10.7.2 | Relation to session signaling bitrate information | 123 |
| 10.7.3 | Use with dynamic bitrate adaptation | 123 |
| 10.7.3.1 | General | 123 |
| 10.7.3.2 | Adaptation of sent media | 124 |
| 10.7.3.3 | Adaptation of received media | 127 |
| 10.7.4 | Message mapping for LTE access | 127 |
| 10.7.5 | Message mapping for NR access | 127 |
| 11 | Front-end handling | 128 |
| 11.1 | General | 128 |
| 12 | Inter-working | 129 |
| 12.1 | General | 129 |
| 12.2 | 3G-324M | 129 |
| 12.2.1 | General | 129 |
| 12.2.2 | Codec usage | 129 |
| 12.2.2.1 | General | 129 |
| 12.2.2.2 | Text | 129 |
| 12.2.3 | Payload format | 129 |
| 12.2.4 | MTSI media gateway trans-packetization | 129 |
| 12.2.4.1 | General | 129 |
| 12.2.4.2 | Speech de-jitter buffer | 129 |
| 12.2.4.3 | Video bitrate equalization | 130 |
| 12.2.4.4 | Data loss detection | 130 |
| 12.2.4.5 | Data integrity indication | 130 |
| 12.2.4.6 | Packet size considerations | 130 |
| 12.2.4.6.0 | General | 130 |
| 12.2.4.6.1 | The Maximum Receive SDU Size attribute "3gpp_MaxRecvSDUSize" | 131 |
| 12.2.4.7 | Setting RTP timestamps | 131 |
| 12.2.4.8 | Protocol termination | 131 |
| 12.2.4.9 | Media synchronization | 131 |
| 12.2.5 | Session control | 132 |
| 12.3 | GERAN/UTRAN CS inter-working | 132 |
| 12.3.0 | 3G-324M | 132 |
| 12.3.1 | Codecs for MTSI media gateways | 132 |
| 12.3.1.1 | Speech interworking between 3GPP PS access and CS GERAN/UTRAN | 132 |
| 12.3.1.1a | Speech inter-working between fixed access and CS GERAN/UTRAN | 133 |
| 12.3.1.2 | Text | 133 |
| 12.3.2 | RTP payload formats for MTSI media gateways | 134 |
| 12.3.2.1 | Speech | 134 |
| 12.3.2.2 | Text | 135 |
| 12.3.3 | Explicit Congestion Notification | 135 |
| 12.3.4 | Codec switching procedures with SRVCC | 135 |
| 12.4 | PSTN | 136 |
| 12.4.1 | 3G-324M | 136 |
| 12.4.2 | Text | 136 |
| 12.5 | GIP inter-working | 136 |
| 12.5.1 | Text | 136 |
| 12.5.2 | Speech | 136 |
| 12.6 | Void | 137 |
| 12.6.1 | Void | 137 |
| 12.6.2 | Void | 137 |
| 12.7 | Inter-working with other IMS and non-IMS IP networks | 137 |
| 12.7.1 | General | 137 |
| 12.7.2 | Speech | 137 |
| 12.7.2.1 | General | 137 |
| 12.7.2.2 | Speech codecs and formats | 137 |

| | | |
|------------|---|-----|
| 12.7.2.2.1 | MTSI MGW for interworking between MTSI client in terminal using 3GPP access and other IMS or non-IMS IP networks | 137 |
| 12.7.2.2.2 | MTSI MGW for interworking between MTSI client in terminal using fixed access and other IMS or non-IMS IP networks | 138 |
| 12.7.2.2.3 | Common procedures..... | 138 |
| 12.7.2.3 | Codec preference order for session negotiation | 138 |
| 12.7.2.4 | RTP profiles | 139 |
| 12.7.2.5 | RTP payload formats..... | 139 |
| 12.7.2.6 | Packetization | 139 |
| 12.7.2.7 | RTCP usage and adaptation | 140 |
| 12.7.2.8 | RTP usage | 140 |
| 12.7.2.9 | Session setup and session modification..... | 141 |
| 12.7.2.10 | Audio level alignment..... | 141 |
| 12.7.3 | Explicit Congestion Notification | 141 |
| 12.7.4 | Text..... | 142 |
| 12.7.5 | Inter-working IPv4 and IPv6 networks..... | 142 |
| 12.8 | MGW handling for NO_REQ interworking..... | 142 |
| 13 | Void..... | 143 |
| 13a | Media types, codecs and formats used for MSRP transport..... | 143 |
| 13a.1 | General | 143 |
| 13a.2 | Difference relative to 3GPP TS 26.141 | 143 |
| 13a.2.1 | Video | 143 |
| 14 | Supplementary services..... | 143 |
| 14.1 | General | 143 |
| 14.2 | Media formats and transport..... | 143 |
| 14.3 | Media handling in hold procedures | 144 |
| 15 | Network preference management object..... | 144 |
| 15.1 | General | 144 |
| 15.2 | Nodes Definition | 144 |
| 15.3 | Example Configuration of 3GPP MTSINP MO..... | 156 |
| 16 | Quality of Experience..... | 157 |
| 16.1 | General | 157 |
| 16.2 | Metrics Definition | 158 |
| 16.2.1 | Corruption duration metric | 158 |
| 16.2.2 | Successive loss of RTP packets | 158 |
| 16.2.3 | Frame rate | 159 |
| 16.2.4 | Jitter duration | 159 |
| 16.2.5 | Sync loss duration..... | 159 |
| 16.2.6 | Round-trip time..... | 160 |
| 16.2.7 | Average codec bitrate | 160 |
| 16.2.8 | Codec information | 160 |
| 16.2.9 | Call setup time | 161 |
| 16.3 | Metric Configuration..... | 161 |
| 16.3.1 | QoE metrics reporting management object..... | 161 |
| 16.3.2 | QoE metric reporting configuration..... | 167 |
| 16.3.3 | QoE reporting rule definition..... | 167 |
| 16.4 | Metrics Reporting..... | 168 |
| 16.4.1 | XML schema for QoE report message..... | 169 |
| 16.4.2 | Example XML for QoE report message | 171 |
| 16.5 | QoE Measurement Collection Functionalities..... | 171 |
| 16.5.1 | Configuration and reporting..... | 171 |
| 16.5.2 | XML configuration | 175 |
| 17 | Management of Media Adaptation..... | 176 |
| 17.1 | General | 176 |
| 17.2 | Media adaptation management object..... | 176 |
| 17.3 | Management procedures..... | 204 |
| 17.3.1 | Management of speech adaptation..... | 204 |
| 17.3.2 | Management of video adaptation..... | 206 |

| | | |
|---|---|------------|
| 17.4 | Management of media robustness adaptation..... | 208 |
| 17.4.1 | General..... | 208 |
| 18 | MTSI client in terminal using fixed access | 209 |
| 18.1 | General | 209 |
| 18.2 | Media codecs..... | 209 |
| 18.2.1 | General..... | 209 |
| 18.2.2 | Speech..... | 209 |
| 18.2.2.1 | Speech codecs | 209 |
| 18.2.2.2 | Error concealment procedures..... | 210 |
| 18.2.2.3 | Source controlled rate operation | 210 |
| 18.2.3 | Video | 211 |
| 18.2.4 | Real-time text..... | 211 |
| 18.3 | Media configuration | 211 |
| 18.3.1 | General..... | 211 |
| 18.3.2 | Session setup procedures | 211 |
| 18.3.2.1 | General | 211 |
| 18.3.2.2 | Speech..... | 211 |
| 18.3.2.3 | Video..... | 212 |
| 18.3.2.4 | Text | 212 |
| 18.3.2.5 | Bandwidth negotiation | 212 |
| 18.3.3 | Session control procedures | 212 |
| 18.4 | Data transport | 213 |
| 18.4.1 | General..... | 213 |
| 18.4.2 | Packetization..... | 213 |
| 18.4.3 | RTP payload format..... | 213 |
| 18.5 | Jitter buffer management..... | 214 |
| 18.6 | Packet-loss handling | 214 |
| 18.7 | Adaptation | 214 |
| 18.8 | Front-end handling | 214 |
| 18.9 | Supplementary services | 214 |
| 19 | Additional bandwidth information | 215 |
| 19.1 | General | 215 |
| 19.2 | Bandwidth properties | 215 |
| 19.2.1 | General description | 215 |
| 19.2.2 | Maximum Supported Bandwidth..... | 216 |
| 19.2.3 | Maximum Desired Bandwidth..... | 216 |
| 19.2.4 | Minimum Desired Bandwidth..... | 217 |
| 19.2.5 | Minimum Supported Bandwidth..... | 217 |
| 19.2.6 | IP version..... | 217 |
| 19.2.7 | Maximum Packet Rate..... | 218 |
| 19.2.8 | Minimum Packet Rate | 218 |
| 19.3 | SDP attribute | 218 |
| 19.3.1 | Definition..... | 218 |
| 19.3.2 | SDP grammar..... | 219 |
| 19.3.3 | Declarative use..... | 220 |
| 19.3.4 | Usage in offer/answer | 220 |
| 19.4 | Modifications of the bandwidth information by intermediate network nodes..... | 221 |
| Annex A (informative): Examples of SDP offers and answers..... | | 223 |
| A.1 | SDP offers for speech sessions initiated by MTSI client in terminal..... | 223 |
| A.1.1 | HSPA or unknown access technology..... | 223 |
| A.1.1.1 | Only AMR-NB supported by MTSI client in terminal | 223 |
| A.1.1.2 | AMR and AMR-WB are supported by MTSI client in terminal..... | 224 |
| A.1.1.2.1 | One-phase approach | 224 |
| A.1.1.2.2 | Two-phase approach | 225 |
| A.1.2 | EGPRS | 226 |
| A.1.3 | Generic Access | 227 |
| A.2 | SDP offers for speech sessions initiated by media gateway..... | 227 |
| A.2.1 | General | 227 |

| | | |
|----------|--|-----|
| A.2.2 | MGW between GERAN UE and MTSI | 227 |
| A.2.3 | MGW between legacy UTRAN UE and MTSI..... | 228 |
| A.2.4 | MGW between CS UE and MTSI..... | 228 |
| A.2.5 | MGW between GERAN UE and MTSI when wideband speech is supported | 229 |
| A.3 | SDP answers to SDP speech session offers..... | 229 |
| A.3.1 | General | 229 |
| A.3.1a | SDP answer from an MTSI client in terminal when only narrowband speech was offered | 230 |
| A.3.2 | SDP answer from an MTSI client in terminal | 230 |
| A.3.2a | SDP answer from a non-MTSI UE with AVP..... | 231 |
| A.3.3 | SDP answer from an MTSI client in terminal supporting only AMR | 232 |
| A.3.4 | SDP answer from an MTSI client in terminal using EGPRS access when both AMR and AMR-WB are supported | 233 |
| A.3.4a | SDP answer from an MTSI client in terminal using EGPRS access when only AMR is supported | 234 |
| A.3.4b | SDP answer from an MTSI client in terminal using WLAN..... | 234 |
| A.3.5 | SDP answer from MTSI MGW supporting only one codec mode set for AMR and AMR-WB each | 235 |
| A.3.5a | SDP answer from MTSI MGW supporting only one codec mode set for AMR | 237 |
| A.3.6 | SDP answer from MTSI client in terminal on HSPA for session initiated from MTSI MGW interfacing UE on GERAN..... | 237 |
| A.3.7 | SDP answer from MTSI client in terminal on HSPA for session initiated from MTSI MGW interfacing legacy UE on UTRAN | 238 |
| A.4 | SDP offers and answers for video sessions | 239 |
| A.4.1 | Void..... | 239 |
| A.4.2 | Void..... | 239 |
| A.4.2a | H.264/AVC | 239 |
| A.4.2b | High Granularity CVO example..... | 241 |
| A.4.2c | RTP Retransmission..... | 241 |
| A.4.2d | RTP Forward Error Correction (FEC)..... | 242 |
| A.4.2e | SDP Examples with ROI..... | 243 |
| A.4.3 | Void..... | 249 |
| A.4.4 | Void..... | 249 |
| A.4.4a | H.264/AVC with "imageattr" attribute..... | 249 |
| A.4.4a.1 | H.264/AVC with "imageattr" attribute – different image sizes in SDP offer and answer | 251 |
| A.4.4a.2 | H.264/AVC with "imageattr" attribute – different payload type numbers in offer and answer | 252 |
| A.4.4b | H.264/AVC with "imageattr" attribute with multiple rtpmaps..... | 253 |
| A.4.5 | H.264 with asymmetric video streams | 255 |
| A.4.6 | H.264/AVC with "imageattr" attribute for non-CVO operation..... | 256 |
| A.4.7 | H.264 (AVC) and H.265 (HEVC)..... | 258 |
| A.4.7.1 | MTSI client with 848x480 resolution 5 inch display..... | 258 |
| A.4.7.2 | MTSI client with 1280x720 resolution 5 inch display..... | 259 |
| A.4.7.3 | MTSI client with 848x480 resolution 10 inch display..... | 260 |
| A.4.7.4 | MTSI client with 1280x720 resolution 10 inch display..... | 261 |
| A.4.8 | H.264 (AVC) and H.265 (HEVC) with asymmetric video streams | 262 |
| A.5 | SDP offers for text..... | 263 |
| A.5.1 | T.140 with and without redundancy | 263 |
| A.6 | SDP example with bandwidth information | 265 |
| A.6.1 | General | 265 |
| A.6.2 | SDP examples with bandwidth information declared with bandwidth modifiers..... | 265 |
| A.6.3 | SDP examples giving additional bandwidth information using the a=bw-info attribute..... | 266 |
| A.7 | SDP examples with "3gpp_sync_info" attribute | 269 |
| A.7.1 | Synchronized streams..... | 269 |
| A.7.2 | Nonsynchronized streams..... | 269 |
| A.8 | SDP example with QoS negotiation..... | 270 |
| A.9 | Void..... | 274 |
| A.9a | SDP offer/answer regarding the use of Reduced-Size RTCP..... | 274 |
| A.10 | Examples of SDP offers and answers for inter-working with other IMS or non-IMS IP networks..... | 275 |
| A.10.1 | General | 275 |

| | | |
|---|--|------------|
| A.10.2 | Session initiated by MTSI client in terminal | 275 |
| A.10.2.1 | SDP offers from an MTSI client in terminal..... | 275 |
| A.10.2.2 | SDP offers modified by MTSI MGW when pre-emptively adding inter-working formats | 276 |
| A.10.2.3 | SDP modified by MGW when adding inter-working formats only when the original SDP offer was rejected..... | 277 |
| A.11 | Adding or removing a video component to/from an on-going video call session..... | 279 |
| A.12 | SDP examples when using ECN | 283 |
| A.12.1 | SDP examples when using ECN for speech..... | 283 |
| A.12.1.1 | With RTP/AVP and zero RTCP bandwidth..... | 283 |
| A.12.1.2 | With RTP/AVPF and non-zero RTCP bandwidth | 284 |
| A.12.1.3 | With RTCP ECN feedback messages and RTCP XR ECN summary reports for inter-working with non-MTSI clients | 285 |
| A.12.2 | SDP examples when using ECN for video in RTP..... | 286 |
| A.12.2.1 | Without RTCP AVPF ECN feedback messages and RTCP XR ECN summary reports | 286 |
| A.12.2.2 | With RTCP AVPF ECN feedback messages and RTCP XR ECN summary reports for inter-working with non-MTSI clients | 287 |
| A.13 | SDP examples for MTSI client in terminal using fixed access | 288 |
| A.13.1 | General | 288 |
| A.13.2 | SDP examples for PCM | 288 |
| A.13.3 | SDP example for G.722..... | 289 |
| A.13.4 | SDP example for EVS, AMR-WB, G.722, AMR, PCM and DTMF | 289 |
| A.14 | SDP offers and answers for speech sessions with EVS | 290 |
| A.14.1 | SDP offers initiated by MTSI client in terminal..... | 290 |
| A.14.1.1 | Unknown access technology..... | 290 |
| A.14.1.2 | EGPRS..... | 291 |
| A.14.1.3 | NR/E-UTRAN/HSPA..... | 292 |
| A.14.1.4 | Dual-mono | 292 |
| A.14.2 | SDP offers initiated by media gateway | 293 |
| A.14.2.1 | Network between MTSI using fixed access and MTSI modifying SDP offer to configure EVS AMR-WB IO mode..... | 294 |
| A.14.3 | SDP answers from MTSI client in terminal | 295 |
| A.14.3.1 | SDP answer from MTSI client in terminal when narrowband speech is negotiated..... | 295 |
| A.14.3.2 | SDP answer from MTSI client in terminal when up to wideband speech is negotiated | 295 |
| A.14.3.3 | SDP answer from MTSI client in terminal when only wideband speech is negotiated | 296 |
| A.14.3.4 | SDP answer from MTSI client in terminal when up to super-wideband speech is negotiated | 296 |
| A.14.3.5 | SDP answer from MTSI client in terminal when only super-wideband speech is negotiated | 296 |
| A.14.3.6 | SDP answer from MTSI client in terminal using WLAN..... | 297 |
| A.14.3.7 | SDP answer from MTSI client in terminal supporting dual-mono | 297 |
| A.14.3.8 | SDP answer from MTSI client in terminal supporting dual-mono for send direction | 297 |
| A.14.3.9 | SDP answer from MTSI client in terminal when SC-VBR is negotiated | 298 |
| A.14.4 | SDP answers from MTSI client in terminal using fixed access..... | 298 |
| A.14.4.1 | SDP answer from MTSI client in terminal using fixed access when narrowband speech is negotiated ... | 298 |
| A.14.4.2 | SDP answer from MTSI client in terminal using fixed access when only wideband speech is negotiated..... | 299 |
| A.14.4.3 | SDP answer from MTSI client in terminal using fixed access when only super-wideband speech is negotiated..... | 299 |
| A.15 | SDP offers and answers with ANBR capability signaling | 300 |
| A.16 | SDP offers and answers with QoS hint signalling..... | 301 |
| A.17 | SDP offers and answers with data channel capability signalling | 303 |
| A.18 | SDP offers and answers for ITT4RT..... | 305 |
| Annex B (informative): Examples of adaptation scenarios | | 307 |
| B.1 | Video bitrate adaptation | 307 |
| Annex C (informative): Example adaptation mechanisms for speech..... | | 308 |
| C.1 | Example of feedback and adaptation for speech and video..... | 308 |

| | | |
|---|--|------------|
| C.1.1 | Introduction | 308 |
| C.1.2 | Signalling state considerations | 309 |
| C.1.3 | Adaptation state machine implementations | 311 |
| C.1.3.1 | General | 311 |
| C.1.3.2 | Adaptation state machine with four states | 313 |
| C.1.3.3 | Adaptation state machine with four states (simplified version without frame aggregation) | 316 |
| C.1.3.4 | Adaptation state machine with two states | 318 |
| C.1.3.5 | Adaptation when using ECN | 319 |
| C.2 | Example criteria for video bit rate feedback and adaptation | 321 |
| C.2.1 | Introduction | 321 |
| C.2.2 | Media sender side | 321 |
| C.2.3 | Media receiver side | 322 |
| C.2.4 | Video encoder bitrate adaptation, down-switch | 322 |
| C.2.5 | Video encoder bitrate adaptation, receiver-driven up-switch | 323 |
| C.2.6 | Video encoder bitrate adaptation, recovery phase | 324 |
| Annex D (informative): Reference delay computation algorithm | | 326 |
| Annex E (informative): QoS profiles | | 328 |
| E.1 | General | 328 |
| E.2 | Bi-directional speech (AMR12.2, IPv4, RTCP and MBR=GBR bearer) | 328 |
| E.3 | Void | 329 |
| E.4 | Bi-directional real-time text (3 kbps, IPv4 or IPv6, RTCP and MBR=GBR bearer) | 330 |
| E.5 | Bi-directional speech (AMR-WB23.85, IPv4, RTCP and MBR=GBR bearer) | 332 |
| E.6 | Bi-directional video (H.264 AVC level 1.1, 192 kbps, IPv4, RTCP and MBR=GBR bearer) | 332 |
| E.7 | Bi-directional speech (AMR12.2, IPv6, RTCP and MBR=GBR bearer) | 333 |
| E.8 | Bi-directional speech (AMR-WB23.85, IPv6, RTCP and MBR=GBR bearer) | 334 |
| E.9 | Void | 335 |
| E.10 | Bi-directional video (H.264 AVC level 1.1, 192 kbps, IPv6, RTCP and MBR=GBR bearer) | 335 |
| E.11 | Bi-directional speech (AMR, IPv4, RTCP and MBR>GBR bearer) | 336 |
| E.12 | Bi-directional speech (AMR-WB, IPv4, RTCP and MBR>GBR bearer) | 337 |
| E.13 | Void | 338 |
| E.14 | Bi-directional video (H.264 AVC level 1.1, IPv4, RTCP and MBR>GBR bearer) | 338 |
| E.15 | Bi-directional speech (AMR, IPv6, RTCP and MBR>GBR bearer) | 339 |
| E.16 | Bi-directional speech (AMR-WB, IPv6, RTCP and MBR>GBR bearer) | 340 |
| E.17 | Void | 341 |
| E.18 | Bi-directional video (H.264 AVC level 1.1, IPv6, RTCP and MBR>GBR bearer) | 341 |
| E.19 | Bi-directional video (H.264 AVC level 1.2, 384 kbps, IPv4, RTCP and MBR=GBR bearer) | 342 |
| E.20 | Bi-directional video (H.264 AVC level 1.2, 384 kbps, IPv6, RTCP and MBR=GBR bearer) | 343 |
| E.21 | Bi-directional video (H.264 AVC level 1.2, IPv4, RTCP and MBR>GBR bearer) | 345 |
| E.22 | Bi-directional video (H.264 AVC level 1.2, IPv6, RTCP and MBR>GBR bearer) | 348 |
| E.23 | Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 500 kbps, IPv6, RTCP and MBR=GBR bearer) | 349 |
| E.24 | Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 500/40 kbps, IPv6, RTCP and MBR>GBR bearer) | 350 |

| | | |
|------|--|-----|
| E.25 | Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 600 kbps, IPv6, RTCP and MBR=GBR bearer) | 351 |
| E.26 | Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 600/40 kbps, IPv6, RTCP and MBR>GBR bearer) | 352 |
| E.27 | Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 650 kbps, IPv6, RTCP and MBR=GBR bearer) | 353 |
| E.28 | Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 650/40 kbps, IPv6, RTCP and MBR>GBR bearer) | 354 |
| E.29 | Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 750 kbps, IPv6, RTCP and MBR=GBR bearer) | 355 |
| E.30 | Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 750/40 kbps, IPv6, RTCP and MBR>GBR bearer) | 356 |
| E.31 | Bi-directional speech (EVS 13.2, IPv4, RTCP and MBR=GBR bearer) | 357 |
| E.32 | Bi-directional speech (EVS 24.4, IPv4, RTCP and MBR=GBR bearer) | 358 |
| E.33 | Bi-directional speech (EVS 13.2, IPv6, RTCP and MBR=GBR bearer) | 359 |
| E.34 | Bi-directional speech (EVS 24.4, IPv6, RTCP and MBR=GBR bearer) | 360 |

Annex F (Normative): Void 362

Annex G (Normative): DTMF events363

| | | |
|-----|--------------------------------------|-----|
| G.1 | General | 363 |
| G.2 | Encoding of DTMF events | 363 |
| G.3 | Session setup | 363 |
| G.4 | Data transport for DTMF events | 365 |

Annex H (informative): Network Preference Management Object Device Description Framework...367

Annex I (informative): QoE Reporting Management Object Device Description Framework.....368

Annex J (informative): Media Adaptation Management Object Device Description Framework.....369

Annex K (informative): Computation of b=AS for AMR and AMR-WB.....370

| | | |
|-----|---|-----|
| K.1 | General | 370 |
| K.2 | Procedure for computing the bandwidth | 370 |
| K.3 | Computation of RTP payload size..... | 370 |
| K.4 | Detailed computation | 371 |

Annex L (Normative): Facsimile transmission380

| | | |
|-------|---|-----|
| L.1 | General | 380 |
| L.2 | FoIP support in MTSI clients | 380 |
| L.2.1 | FoIP support in MTSI client in terminal | 380 |
| L.2.2 | FoIP support in MTSI MGW | 380 |
| L.2.3 | Recommended configuration..... | 380 |
| L.3 | Session setup | 381 |
| L.3.1 | Session setup for any MTSI client supporting facsimile transmission | 381 |
| L.3.2 | Session setup when the recommended profile is supported | 381 |
| L.4 | Data transport using UDP/IP | 382 |
| L.5 | CS GERAN inter-working | 382 |

| | | |
|--|---|------------|
| L.6 | PSTN inter-working | 383 |
| L.7 | SDP examples | 383 |
| L.7.1 | Facsimile-only session | 383 |
| Annex M (informative): IANA registration information for SDP attributes..... | | 385 |
| M.1 | Introduction | 385 |
| M.2 | 3gpp_sync_info | 385 |
| M.3 | 3gpp_MaxRecvSDUSize..... | 385 |
| M.4 | 3gpp_mtsi_app_adapt..... | 386 |
| M.5 | predefined_ROI..... | 387 |
| M.6 | bw-info | 387 |
| M.7 | ccc_list..... | 388 |
| M.8 | anbr..... | 388 |
| M.9 | PLR_adapt..... | 389 |
| M.10 | MAXimum-e2e-PLR..... | 390 |
| M.11 | 3gpp-qos-hint | 390 |
| Annex N (informative): Computation of b=AS for Video Codec | | 397 |
| N.1 | General | 397 |
| N.2 | Examples | 397 |
| Annex O (informative): IANA registration information for RTP Header Extensions..... | | 398 |
| O.1 | Introduction | 398 |
| O.2 | urn:3gpp:video-orientation..... | 398 |
| O.3 | urn:3gpp:video-orientation:6..... | 398 |
| O.4 | urn:3gpp:roi-sent | 398 |
| O.5 | urn:3gpp:predefined-roi-sent..... | 399 |
| O.6 | urn:3gpp:audio-mixing-gain..... | 399 |
| Annex P (informative): Video packet loss handling operation principles and examples..... | | 400 |
| P.1 | General | 400 |
| P.2 | Video error recovery | 400 |
| P.3 | RTP Retransmission..... | 402 |
| Annex Q (informative): Computation of b=AS for EVS..... | | 404 |
| Q.1 | General | 404 |
| Q.2 | Procedure for computing the bandwidth | 404 |
| Q.3 | Computation of RTP payload size..... | 404 |
| Q.4 | Detailed computation | 405 |
| Annex R (informative): IANA registration information for RTCP Feedback Message Types..... | | 407 |
| R.1 | Video Region-of-Interest (ROI) | 407 |
| R.2 | Delay Budget Information (DBI) | 407 |
| R.3 | Viewport (VP)..... | 408 |

| | |
|---|------------|
| Annex S (normative): Multi-party Multimedia Conference Media Handling | 409 |
| S.1 General | 409 |
| S.2 Video | 409 |
| S.2.1 Conversational video | 409 |
| S.2.2 Non-conversational (screenshare) video | 409 |
| S.3 Audio | 409 |
| S.3.1 General | 409 |
| S.3.2 De-jitter buffer | 410 |
| S.4 SIP | 410 |
| S.4.1 MSMTSI client in terminal | 410 |
| S.4.2 MSMTSI MRF | 411 |
| S.5 Media configuration | 411 |
| S.5.1 General | 411 |
| S.5.2 Main video | 412 |
| S.5.3 Thumbnail video | 412 |
| S.5.4 Screenshare video | 413 |
| S.5.5 Audio | 413 |
| S.5.6 BFCP | 413 |
| S.5.7 Compact Concurrent Codec Negotiation and Capabilities | 413 |
| S.5.7.1 General | 413 |
| S.5.7.2 The Compact CCC SDP Attribute | 413 |
| S.5.7.3 Using the Compact CCC SDP Attribute for CCC Exchange | 414 |
| S.5.7.4 Using the Compact CCC SDP Attribute for Session Initiation | 415 |
| S.5.7.4.1 General | 415 |
| S.5.7.4.2 SDP Offer Rules | 415 |
| S.5.7.4.3 SDP Answer Rules | 415 |
| S.6 Media transport | 415 |
| S.6.1 RTP | 415 |
| S.6.2 RTCP | 416 |
| S.6.3 RTP Stream Selective Forwarding | 416 |
| S.7 BFCP | 417 |
| S.7.1 General | 417 |
| S.7.2 Floor controlled main video | 417 |
| S.7.3 Floor controlled screenshare video | 417 |
| S.7.4 Implicit floor control for audio | 418 |
| S.7.5 Floor control interworking with DTMF-capable MTSI clients | 418 |
| S.8 Rate Adaptation | 418 |
| Annex T (informative): SDP examples for Multi-party Multimedia Conference Media Handling | 419 |
| T.1 General | 419 |
| T.1.1 Introduction | 419 |
| T.1.2 Quality of Service examples | 419 |
| T.2 MSMTSI video offer/answer examples | 420 |
| T.2.1 MSMTSI offer/answer towards an MTSI client | 420 |
| T.2.2 MSMTSI answer from an MSMTSI MRF | 422 |
| T.2.3 MSMTSI answer from an MSMTSI client in terminal | 424 |
| T.2.4 MSMTSI simulcast offer using a single payload type | 425 |
| T.2.5 MSMTSI simulcast offer using two codecs | 425 |
| T.3 MSMTSI audio offer/answer examples | 426 |
| T.3.1 MSMTSI offer with multi-stream audio support | 426 |
| T.3.2 MSMTSI answer with multi-stream audio support | 427 |
| T.3.3 MSMTSI CCCEX SDP offer/answer example | 428 |
| T.3.3a MSMTSI offer/answer examples using the compact CCC SDP attribute | 432 |
| T.3.4 SIP OPTIONS request for multi-stream audio support | 434 |

| | |
|---|------------|
| Annex U (informative): IANA registration information for media types | 436 |
| U.1 Introduction | 436 |
| U.2 application/ccce | 436 |
| Annex V (informative): Delay Adaptation and Example Uses of DBI Signaling..... | 437 |
| V.1 General | 437 |
| V.2 Example Signaling Flows on Delay Adaptation | 437 |
| V.3 SDP Examples on DBI Signaling Capability | 443 |
| Annex W (Normative): Coverage and Handoff Enhancements using Multimedia Error Robustness (CHEM)..... | 445 |
| W.1 General | 445 |
| W.2 Adaptation to Packet Losses without application layer redundancy | 446 |
| W.3 Adaptation to Packet Losses using Application Layer Redundancy | 446 |
| W.4 Negotiation of End-to-End and Uplink/Downlink PLR..... | 447 |
| W.4.1 General | 447 |
| W.4.2 Offering MTSI Client..... | 448 |
| W.4.3 Answering MTSI Client | 449 |
| Annex X (Informative) : Example Maximum Packet Loss Rate (Max. PLR) Values for Setting CHEM Handover Thresholds..... | 451 |
| X.1 Maximum Packet Loss Rate (Max. PLR) for Speech | 451 |
| X.1.1 Max. PLR recommendation without Application Layer Redundancy..... | 451 |
| X.1.2 Max. PLR recommendation with Application Layer Redundancy..... | 451 |
| X.2 SDP Examples of the CHEM Feature (informative)..... | 452 |
| X.2.1 General | 452 |
| X.2.2 Example of Adaptation to Packet Losses without Application Layer Redundancy | 452 |
| X.2.3 Example of Adaptation to Packet Losses with Application Layer Redundancy | 453 |
| X.2.4 Example of Maximum End-to-End Packet Loss Rate | 455 |
| Annex Y (normative): Immersive Teleconferencing and Telepresence for Remote Terminals (ITT4RT) | 457 |
| Y.1 General | 457 |
| Y.2 Architecture and Interfaces | 457 |
| Y.3 Immersive 360-Degree Video Support..... | 459 |
| Y.4 Immersive Voice/Audio Support..... | 461 |
| Y.5 Overlay Support | 461 |
| Y.6 Media configuration | 462 |
| Y.6.1 General | 462 |
| Y.6.2 Main 360-degree video..... | 462 |
| Y.6.2.1 General..... | 462 |
| Y.6.2.2 Projection..... | 464 |
| Y.6.2.3 Field Of View (FOV)..... | 464 |
| Y.6.2.4 Picture Packing | 465 |
| Y.6.2.6 Overlay and 360-degree video | 466 |
| Y.6.2.7 Viewport Size | 467 |
| Y.6.2.8 Viewport-only VDP and Image attributes | 467 |
| Y.6.2.9 Viewport Feedback Trigger | 467 |
| Y.6.3 Still Background..... | 468 |
| Y.6.4 Overlays | 468 |
| Y.6.4.1 General..... | 468 |

| | | |
|--|---|------------|
| Y.6.4.2 | Visual Media..... | 468 |
| Y.6.4.3 | Overlay Configuration | 469 |
| Y.6.4.3.1 | General | 469 |
| Y.6.4.3.2 | Sphere-relative Overlay Configuration | 470 |
| Y.6.4.3.3 | Viewport-relative Overlay Configuration | 471 |
| Y.6.4.3.4 | Overlay info parameter | 472 |
| Y.6.4.3.5 | Additional Overlay Configuration | 473 |
| Y.6.4.4 | Captured Content Replacement | 473 |
| Y.6.5 | Fisheye Video..... | 474 |
| Y.6.5.1 | Identifying the 360-degree fisheye video stream | 474 |
| Y.6.5.3 | Viewport dependent delivery of fisheye video | 476 |
| Y.6.6 | Camera Calibration for Network-based Stitching | 476 |
| Y.6.7 | Support for Stream Pausing/Resuming..... | 478 |
| Y.6.8 | Multiple 360-degree videos | 478 |
| Y.6.8.2 | Excluding other participants' overlays | 479 |
| Y.6.9 | Scene Description-Based Overlays | 479 |
| Y.6.9.1 | General..... | 479 |
| Y.6.9.2 | Offer/Answer Negotiation | 480 |
| Y.6.9.3 | SDP Signaling..... | 480 |
| Y.7 | Media transport | 480 |
| Y.7.1 | RTP | 480 |
| Y.7.2 | RTCP..... | 480 |
| Y.8 | SDP Examples (informative)..... | 482 |
| Y.9 | Recommended audio mixing gains | 483 |
| Y.9.1 | RTP header extension for audio mixing gain | 484 |
| Annex Z (informative): Change history | | 485 |
| History | | 492 |

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
 - 1 presented to TSG for information;
 - 2 presented to TSG for approval;
 - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

Multimedia Telephony Service for IMS (MTSI), here also referred to as Multimedia Telephony, is a standardized IMS telephony service that builds on the IMS capabilities to establish multimedia communications between terminals within and in-between operator networks. The terminals connect to the IMS using either a fixed access network or a 3GPP access network.

The objective of defining a service is to specify the minimum set of capabilities required in the IP Multimedia Subsystem to secure multi-vendor and multi-operator inter-operability for Multimedia Telephony and related Supplementary Services. The objective also includes defining procedures for inter-working between different clients and networks.

The user experience of multimedia telephony is expected to be equivalent to or better than corresponding circuit-switched telephony services. Multimedia telephony also exploits the richer capabilities of IMS. In particular, multiple media components can be used and dynamically added or dropped during a session.

1 Scope

The present document specifies a client for the Multimedia Telephony Service for IMS (MTSI) supporting conversational speech (including DTMF), video and text transported over RTP, and flexible data channel handling, with the scope to deliver a user experience equivalent to or better than that of Circuit Switched (CS) conversational services using the same amount of network resources. It defines media handling (e.g. signalling, transport, jitter buffer management, packet-loss handling, adaptation), as well as interactivity (e.g. adding or dropping media during a call). The focus is to ensure a reliable and interoperable service with a predictable media quality, while allowing for flexibility in the service offerings.

The present document describes two client types:

- An MTSI client in terminal which uses a 3GPP access (NR, LTE, HSPA, or EGPRS) to connect to the IMS. These clients are described in Clauses 5 – 17 and Annexes A – M.
- An MTSI client in terminal which uses a fixed access (corded interface, fixed-wireless interface, e.g. Wi-Fi, Bluetooth or DECT/NG DECT) to connect to the IMS. These clients are described in Clause 18.

MTSI clients using 3GPP access and MTSI clients using fixed access have many common procedures for the media handling. This specification aligns the media handling by using cross references whenever possible. This does not mean that 3GPP terminals must support fixed access, nor does it mean that fixed terminals must support 3GPP access.

The scope includes maintaining backward compatibility in order to ensure seamless inter-working with existing services available in the CS domain, such as CS speech and video telephony, as well as with terminals of earlier 3GPP releases. In addition, inter-working with other IMS and non-IMS IP networks as well as traditional PSTN is covered.

The client may also support the IMS Messaging service and Group 3 facsimile transmission. The scope therefore also includes media handling for non-conversational media using MSRP and UDPTL-based Facsimile over IP (FoIP).

The specification is written in a forward-compatible way in order to allow additions of media components and functionality in releases after Release 7.

NOTE 1: MTSI clients can support more than conversational speech, video and text, which is the scope of the present document. See TS 22.173 [2] for the definition of the Multimedia Telephony Service for IMS.

NOTE 2: TS 26.235 [3] and TS 26.236 [4] do not include the specification of an MTSI client, although they include conversational multimedia applications. Only those parts of TS 26.235 [3] and TS 26.236 [4] that are specifically referenced by the present document apply to Multimedia Telephony Service for IMS.

NOTE 3: The present document was started as a conclusion from the study in TR 26.914 [5] on optimization opportunities in Multimedia Telephony for IMS (TR 22.973 [6]).

NOTE 4: For ECN, the present specification assumes that an interface enables the MTSI client to read and write the ECN field. This interface is outside the scope of this specification.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

- [2] 3GPP TS 22.173: "IP Multimedia Core Network Subsystem (IMS) Multimedia Telephony Service and supplementary services; Stage 1".
- [3] 3GPP TS 26.235: "Packet switched conversational multimedia applications; Default codecs".
- [4] 3GPP TS 26.236: "Packet switched conversational multimedia applications; Transport protocols".
- [5] 3GPP TR 26.914: "Multimedia telephony over IP Multimedia Subsystem (IMS); Optimization opportunities".
- [6] 3GPP TR 22.973: "IMS Multimedia Telephony service; and supplementary services".
- [7] 3GPP TS 24.229: "IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3".
- [8] IETF RFC 4566 (2006): "SDP: Session Description Protocol", M. Handley, V. Jacobson and C. Perkins.
- [9] IETF RFC 3550 (2003): "RTP: A Transport Protocol for Real-Time Applications", H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson.
- [10] IETF RFC 3551 (2003): "RTP Profile for Audio and Video Conferences with Minimal Control", H. Schulzrinne and S. Casner.
- [11] 3GPP TS 26.071: "Mandatory Speech Codec speech processing functions; AMR Speech CODEC; General description".
- [12] 3GPP TS 26.090: "Mandatory Speech Codec speech processing functions; Adaptive Multi-Rate (AMR) speech codec; Transcoding functions".
- [13] 3GPP TS 26.073: "ANSI C code for the Adaptive Multi Rate (AMR) speech codec".
- [14] 3GPP TS 26.104: "ANSI-C code for the floating-point Adaptive Multi Rate (AMR) speech codec".
- [15] 3GPP TS 26.093: "Mandatory speech codec speech processing functions; Adaptive Multi-Rate (AMR) speech codec; Source controlled rate operation".
- [16] 3GPP TS 26.103: "Speech codec list for GSM and UMTS".
- [17] 3GPP TS 26.171: "Speech codec speech processing functions; Adaptive Multi-Rate - Wideband (AMR-WB) speech codec; General description".
- [18] 3GPP TS 26.190: "Speech codec speech processing functions; Adaptive Multi-Rate - Wideband (AMR-WB) speech codec; Transcoding functions".
- [19] 3GPP TS 26.173: "ANSI-C code for the Adaptive Multi Rate - Wideband (AMR-WB) speech codec".
- [20] 3GPP TS 26.204: "Speech codec speech processing functions; Adaptive Multi-Rate - Wideband (AMR-WB) speech codec; ANSI-C code".
- [21] 3GPP TS 26.193: "Speech codec speech processing functions; Adaptive Multi-Rate - Wideband (AMR-WB) speech codec; Source controlled rate operation".
- [22] Void.
- [23] Void.
- [24] Recommendation ITU-T H.264 (04/2017): "Advanced video coding for generic audiovisual services" | ISO/IEC 14496-10:2014: "Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding".
- [25] IETF RFC 6184 (2011): "RTP Payload Format for H.264 Video", Y.-K. Wang, R. Even, T. Kristensen, R. Jesup.
- [26] ITU-T Recommendation T.140 (02/1998): "Protocol for multimedia application text conversation".

- [27] ITU-T Recommendation T.140 (02/2000): "Protocol for multimedia application text conversation - Addendum 1".
- [28] IETF RFC 4867 (2007): "RTP Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs", J. Sjoberg, M. Westerlund, A. Lakaniemi and Q. Xie.
- [29] Void
- [30] Void.
- [31] IETF RFC 4103 (2005): "RTP Payload for Text Conversation", G. Hellstrom and P. Jones.
- [32] Void.
- [33] 3GPP TR 25.993: "Typical examples of Radio Access Bearers (RABs) and Radio Bearers (RBs) supported by Universal Terrestrial Radio Access (UTRA)".
- [34] 3GPP TS 22.105: "Services and service capabilities".
- [35] 3GPP TS 26.131: "Terminal acoustic characteristics for telephony; Requirements".
- [36] 3GPP TS 26.132: "Speech and video telephony terminal acoustic test specification".
- [37] 3GPP TS 28.062: "Inband Tandem Free Operation (TFO) of speech codecs; Service description; Stage 3".
- [38] 3GPP TS 23.153: "Out of band transcoder control; Stage 2".
- [39] IETF RFC 0768 (1980): "User Datagram Protocol", J. Postel.
- [40] IETF RFC 4585 (2006): "Extended RTP Profile for Real-time Transport Control Protocol (RTCP) - Based Feedback (RTP/AVPF)", J. Ott, S. Wenger, N. Sato, C. Burmeister and J. Rey.
- [41] RTP Tools: <http://www.cs.columbia.edu/IRT/software/rtptools/>.
- [42] IETF RFC 3556 (2003): "Session Description Protocol (SDP) Bandwidth Modifiers for RTP Control Protocol (RTCP) Bandwidth", S. Casner.
- [43] IETF RFC 5104 (2008): "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)", S. Wenger, U. Chandra, M. Westerlund and B. Burman.
- [44] Void.
- [45] 3GPP TS 26.111: "Codec for circuit switched multimedia telephony service; Modifications to H.324".
- [46] 3GPP TS 23.172: "Technical realization of Circuit Switched (CS) multimedia service; UDI/RDI fallback and service modification; Stage 2".
- [47] 3GPP TS 23.002: "Network Architecture".
- [48] IETF RFC 3388 (2002): "Grouping of Media Lines in the Session Description Protocol (SDP)", G. Camarillo, G. Eriksson, J. Holler and H. Schulzrinne.
- [49] IETF RFC 4102 (2005): "Registration of the text/red MIME Sub-Type", P. Jones.
- [50] ITU-T H.248 (06/2000): "Packages for text conversation, fax and call discrimination".
- [51] ETSI EG 202 320, v1.2.1 (2005-10): "Human Factors (HF); Duplex Universal Speech and Text (DUST) communications".
- [52] 3GPP TS 26.226: "Cellular text telephone modem; General description".
- [53] IETF RFC 4504 (2006): "SIP Telephony Device Requirements and Configuration", H. Sinnreich, Ed., S. Lass and C. Stredicke.

- [54] ITU-T Recommendation V.151 (05/2006): "Procedures for end-to-end connection of analogue PSTN text telephones over an IP network utilizing text relay".
- [55] ITU-T Recommendation V.152 (09/2010): "Procedures for supporting Voice Band Data over IP networks".
- [56] IETF RFC 3448 (2003): "TCP Friendly Rate Control (TFRC): Protocol Specification", M. Handley, S. Floyd, J. Padhye and J. Widmer.
- [57] 3GPP TS 24.173: "IMS Multimedia Telephony Communication Service and Supplementary Services".
- [58] IETF RFC 3264 (2002): "An Offer/Answer Model with the Session Description Protocol (SDP)", J. Rosenberg and H. Schulzrinne.
- [59] 3GPP TS 26.141: "IP Multimedia System (IMS) Messaging and Presence; Media formats and codecs".
- [60] 3GPP TS 26.234: "Transparent end-to-end Packet-switched Streaming Service; Protocols and codecs".
- [61] IETF RFC 4733 (2006): "RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals", H. Schulzrinne and T. Taylor.
- [62] 3GPP TS 23.014: "Support of Dual Tone Multi-Frequency (DTMF) signalling".
- [63] ETSI ES 201 235-2, v1.2.1: "Specification of Dual Tone Multi-Frequency (DTMF); Transmitters and Receivers; Part 2: Transmitters".
- [64] 3GPP TS 23.107: "Quality of Service (QoS) concept and architecture".
- [65] 3GPP TS 29.163: "Interworking between the IP Multimedia (IM) Core Network (CN) subsystem and Circuit Switched (CS) networks".
- [66] Void.
- [67] OMA-ERELED-DM-V1_2-20070209-A: "Enabler Release Definition for OMA Device Management, Approved Version 1.2".
- [68] Void.
- [69] IETF RFC 5939 (2010): "Session Description Protocol (SDP) Capability Negotiation", F. Andreasen.
- [70] Void
- [71] IETF RFC 1952 (May 1996): "GZIP file format specification version 4.3", P. Deutsch.
- [72] IETF RFC 2326 (1998): "Real Time Streaming Protocol (RTSP)".
- [73] IETF RFC 2616 (June 1999): "Hypertext Transfer Protocol -- HTTP/1.1".
- [74] 3GPP TS 26.346 "Multimedia Broadcast/Multicast Service (MBMS); Protocols and codecs".
- [75] Void
- [76] IETF RFC 6236 (2011): "Negotiation of Generic Image Attributes in the Session Description Protocol (SDP)", I. Johansson and K. Jung.
- [77] ITU-T G.711 (11/1988): "Pulse code modulation (PCM) of voice frequencies".
- [78] ITU-T G.722 (09/2012): "7 kHz audio-coding within 64 kbit/s".
- [79] IETF RFC 4821 (2007): "Packetization Layer Path MTU Discovery".
- [80] 3GPP TS 23.003: "Numbering, addressing and identification".

- [81] IETF RFC 4796 (2007): "The Session Description Protocol (SDP) Content Attribute", J. Hautakorpi and G. Camarillo.
- [82] 3GPP TS 24.247: "Messaging service using the IP Multimedia (IM) Core Network (CN) subsystem".
- [83] IETF RFC 3168 (2001): "The Addition of Explicit Congestion Notification (ECN) to IP", K. Ramakrishnan, S. Floyd and D. Black.
- [84] IETF RFC 6679 (2012): "Explicit Congestion Notification (ECN) for RTP over UDP", M. Westerlund, et. al.
- [85] 3GPP TS 36.300: "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description".
- [86] Void
- [87] IETF RFC 5506 (2009): "Support for Reduced-Size Real-Time Transport Control Protocol (RTCP): Opportunities and Consequences".
- [88] IETF RFC 3611 (2003): "RTP Control Protocol Extended Reports (RTCP XR) ", T. Friedman, R. Caceres and A. Clark.
- [89] 3GPP TS 25.401: "UTRAN overall description".
- [90] 3GPP TS 23.203: "Policy and charging control architecture".
- [91] ITU-T Recommendation T.4 (07/2003): "Standardization of Group 3 facsimile terminals for document transmission".
- [92] ITU-T Recommendation T.30 (09/2005): "Procedures for document facsimile transmission in the general switched telephone network".
- [93] ITU-T Recommendation T.38 (09/2010): "Procedures for real-time Group 3 facsimile communication over IP networks".
- [94] IETF RFC 3362 (2002): "Real-time Facsimile (T.38) - image/t38 MIME Sub-type Registration".
- [95] IETF RFC 5285 (2008): "A General Mechanism for RTP Header Extensions", D. Singer, H. Desineni.
- [96] IETF RFC 5168 (2008): "XML Schema for Media Control", O. Levin, R. Even and P. Hagendorf.
- [97] 3GPP2 C.S0055-A, version 1.0: "Packet Switched Video Telephony Service (PSVT/MCS)".
- [98] ETSI TS 181 005, v3.3.1: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Service and Capability Requirements".
- [99] 3GPP2 C.S0014-E, version 1.0: "Enhanced Variable Rate Codec (EVRC)".
- [100] ITU-T Recommendation G.729 (06/2012): "Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)".
- [101] ITU-T Recommendation G.729.1 (05/2006): "G.729-based embedded variable bit-rate coder: An 8-32 kbit/s scalable wideband coder bitstream interoperable with G.729".
- [102] 3GPP2 C.S0076, version 1.0: "Discontinuous Transmission (DTX) of Speech in cdma2000 Systems".
- [103] IETF RFC 5188 (2008): "RTP Payload Format for the Enhanced Variable Rate Wideband Codec (EVRC-WB) and the Media Subtype Updates for EVRC-B Codec".
- [104] IETF RFC 4749 (2006): "RTP Payload Format for the G.729.1 Audio Codec".
- [105] IETF RFC 5459 (2009): "G.729.1 RTP Payload Format Update: Discontinuous Transmission (DTX) Support".

- [106] IETF RFC 4788 (2007): "Enhancements to RTP Payload Formats for EVRC Family Codecs".
- [107] IETF RFC 4855 (2007): "Media Type Registration of RTP Payload Formats".
- [108] ITU-T Recommendation P.10 (07/2006): "Vocabulary and effects of transmission parameters on customer opinion of transmission quality".
- [109] ETSI TS 103 737, v1.1.2: "Speech and multimedia Transmission Quality (STQ); Transmission requirements for narrowband wireless terminals (handset and headset) from a QoS perspective as perceived by the user".
- [110] ETSI TS 103 738, v1.1.2: "Speech and multimedia Transmission Quality (STQ); Transmission requirements for narrowband wireless terminals (handsfree) from a QoS perspective as perceived by the user".
- [111] ETSI TS 103 739, v1.1.2: "Speech and multimedia Transmission Quality (STQ); Transmission requirements for wideband wireless terminals (handset and headset) from a QoS perspective as perceived by the user".
- [112] ETSI TS 103 740, v1.1.2: "Speech and multimedia Transmission Quality (STQ); Transmission requirements for wideband wireless terminals (handsfree) from a QoS perspective as perceived by the user".
- [113] ETSI TS 202 737, v1.3.2: "Speech and multimedia Transmission Quality (STQ); Transmission requirements for narrowband VoIP terminals (handset and headset) from a QoS perspective as perceived by the user".
- [114] ETSI TS 202 738, v1.3.2: "Speech and multimedia Transmission Quality (STQ); Transmission requirements for narrowband VoIP loudspeaking and handsfree terminals from a QoS perspective as perceived by the user".
- [115] ETSI TS 202 739, v1.3.2: "Speech and multimedia Transmission Quality (STQ); Transmission requirements for wideband VoIP terminals (handset and headset) from a QoS perspective as perceived by the user".
- [116] ETSI TS 202 740, v1.3.2: "Speech and multimedia Transmission Quality (STQ); Transmission requirements for wideband VoIP loudspeaking and handsfree terminals from a QoS perspective as perceived by the user".
- [117] ETSI EN 300 175-8, v2.5.1: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 8: Speech and audio coding and transmission".
- [118] ETSI TS 300 176-2, v2.2.1: "Digital Enhanced Cordless Telecommunications (DECT); Test specification; Part 2: Audio and speech".
- [119] Recommendation ITU-T H.265: "High efficiency video coding" | ISO/IEC 23008-2:2020: "High Efficiency Coding and Media Delivery in Heterogeneous Environments – Part 2: High Efficiency Video Coding". [120] IETF RFC 7798 (2016): "RTP Payload Format for High Efficiency Video Coding (HEVC)", Y.-K. Wang, Y. Sanchez, T. Schierl, S. Wenger, M. M. Hannuksela.
- [121] 3GPP TS 26.441: "Codec for Enhanced Voice Services (EVS); General Overview".
- [122] 3GPP TS 26.442: "Codec for Enhanced Voice Services (EVS); ANSI C code (fixed-point)".
- [123] 3GPP TS 26.443: "Codec for Enhanced Voice Services (EVS); ANSI C code (floating-point)".
- [124] 3GPP TS 26.444: "Codec for Enhanced Voice Services (EVS); Test Sequences".
- [125] 3GPP TS 26.445: "Codec for Enhanced Voice Services (EVS); Detailed Algorithmic Description".
- [126] 3GPP TS 26.446: "Codec for Enhanced Voice Services (EVS); AMR-WB Backward Compatible Functions".
- [127] 3GPP TS 26.447: "Codec for Enhanced Voice Services (EVS); Error Concealment of Lost Packets".

- [128] 3GPP TS 26.448: "Codec for Enhanced Voice Services (EVS); Jitter Buffer Management".
- [129] 3GPP TS 26.449: "Codec for Enhanced Voice Services (EVS); Comfort Noise Generation (CNG) Aspects".
- [130] 3GPP TS 26.450: "Codec for Enhanced Voice Services (EVS); Discontinuous Transmission (DTX)".
- [131] 3GPP TS 26.451: "Codec for Enhanced Voice Services (EVS); Voice Activity Detection (VAD)".
- [132] 3GPP TS 45.003: "Radio Access Network; Channel coding".
- [133] 3GPP TS 23.216: "Single Radio Voice Call Continuity (SRVCC); Stage2".
- [134] 3GPP TS 23.237: "IP Multimedia Subsystem (IMS) Service Continuity; Stage2".
- [135] ITU-T Recommendation H.224 (01/05): "A real time control protocol for simplex applications using the H.221 LSD/HSD/MLP channels".
- [136] ITU-T Recommendation H.224 (2005): Corrigendum 1 (08/07).
- [137] ITU-T Recommendation H.281 (11/94): Transmission of non-telephone signals "A far end camera control protocol for videoconferences using H.224".
- [138] ITU-T Recommendation H.323 (12/2009): "Packet-based multimedia communications systems".
- [139] IETF RFC 4573 (2006): "MIME Type Registration for RTP Payload Format for H.224".
- [140] IETF RFC 4588 (2006): "RTP Retransmission Payload Format", J. Rey, D. Leon, A. Miyazaki, V. Varsa and R. Hakenberg.
- [141] IETF RFC 8627 (2019): "RTP Payload Format for Flexible Forward Error Correction (FEC)".
- [142] TR 26.922: "Video Telephony Robustness Improvements Extensions (VTRI_EXT): Performance Evaluation".
- [143] IETF RFC 5956 (2010): "Forward Error Correction Grouping Semantics in the Session Description Protocol", A. Cengiz.
- [144] 3GPP TR 26.924: "Multimedia telephony over IP Multimedia Subsystem (IMS); Study on improved end-to-end Quality of Service (QoS) handling for Multimedia Telephony Service for IMS (MTSI)".
- [145] Void
- [146] Void.
- [147] 3GPP TS 24.147: "Conferencing Using IP Multimedia Core Network; Stage 3".
- [148] IETF RFC 4575 (2006): "A Session Initiation Protocol (SIP) Event Package for Conference State".
- [149] IETF RFC 4582 (2006): "The Binary Floor Control Protocol (BFCP)".
- [150] IETF RFC 4583 (2006): "Session Description Protocol (SDP) Format for Binary Floor Control (BFCP) Streams".
- [151] Void.
- [152] 3GPP TR 26.980: "Multimedia telephony over IP Multimedia Subsystem (IMS); Media handling aspects of multi-stream multiparty conferencing for Multimedia Telephony Service for IMS (MTSI)".
- [153] IETF RFC 5234 (2008): "Augmented BNF for Syntax Specifications: ABNF", D. Crocker and P. Overell.
- [154] IETF RFC 8853 (2021): "Using Simulcast in Session Description Protocol (SDP) and RTP Sessions"

- [155] IETF RFC 8851 (2021): "RTP Payload Format Restrictions"
- [156] IETF RFC 7728 (2016): "RTP Stream Pause and Resume".
- [157] 3GPP TS 36.321: "Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification".
- [158] 3GPP TS 25.331: "Radio Resource Control (RRC); Protocol specification".
- [159] "Mobile Location Protocol (MLP)", Open Mobile Alliance, OMA-LIF-MLP-V3_1, Approved Version 3.1 – 20 Sep 2011.
- [160] 3GPP TS 36.331: "Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification".
- [161] 3GPP TS 27.007: " Technical Specification Group Core Network and Terminals; AT command set for User Equipment (UE)".
- [162] Void
- [163] 3GPP TS 38.331: "NR; Radio Resource Control (RRC); Protocol Specification".
- [164] 3GPP TS 38.300: "NR; NR and NG-RAN Overall Description; Stage 2".
- [165] 3GPP TS 26.452: "Codec for Enhanced Voice Services (EVS); ANSI C code; Alternative fixed-point using updated basic operators".
- [166] 3GPP TS 38.321: "NR; Medium Access Control (MAC) protocol specification".
- [167] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2".
- [168] 3GPP TR 26.952: "Codec for Enhanced Voice Services (EVS); Performance characterization".
- [169] 3GPP TR 26.959: "Study on enhanced Voice over LTE (VoLTE) performance".
- [170] 3GPP TS 36.323: "Evolved Universal Terrestrial Radio Access (E-UTRA); Packet Data Convergence Protocol (PDCP) specification".
- [171] 3GPP TS 37.324: "Evolved Universal Terrestrial Radio Access (E-UTRA) and NR; Service Data Adaptation Protocol (SDAP) specification".
- [172] IETF RFC 8864 (2021): "Negotiation Data Channels Using the Session Description Protocol (SDP)"
- [173] IETF RFC 4960 (2007): "Stream Control Transmission Protocol"
- [174] IETF RFC 8261 (2017): "Datagram Transport Layer Security (DTLS) Encapsulation of SCTP Packets"
- [175] IETF RFC 8831 (2021): "WebRTC Data Channels".
- [176] 3GPP TS 23.501: "System Architecture for the 5G System; Stage 2".
- [177] IETF RFC 5688 (2010): "A Session Initiation Protocol (SIP) Media Feature Tag for MIME Application Subtypes".
- [178] 3GPP TS 28.405; "Management of Quality of Experience (QoE) measurement collection; Control and configuration"
- [179] ISO/IEC 23090-2:2019: " Information technology -- Coded representation of immersive media -- Part 2: Omnidirectional media format".
- [180] 3GPP TS 26.118: "3GPP Virtual reality profiles for streaming applications".
- [181] ITU-T Recommendation G.1028 (06/2019): "End-to-end quality of service for voice over 4G mobile networks".

- [182] 3GPP TS 24.526: "User Equipment (UE) policies for 5G System (5GS); Stage 3".
- [183] ISO/IEC 23090-14: Information technology — Coded representation of immersive media — Part 14: Scene Description for MPEG Media.
- [184] IETF RFC 8839 (2021): "Session Description Protocol (SDP) Offer/Answer Procedures for Interactive Connectivity Establishment (ICE)".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1] and the following apply:

NOTE: A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

example: text used to clarify abstract rules by applying them literally.

360-degree video: A real-world visual scene captured by a set of cameras or a camera device with multiple lenses and sensors covering the sphere in all directions around the centre point of the camera set or camera device. The term 360-degree video may be used to include also limited 360-degree video.

Limited 360-degree video: A 360-degree video in which the visual scene does not cover the entire sphere around the center point of the camera set or camera device but only a part of it. A limited 360-degree video may be limited i) in the horizontal field to less than 360 degrees, or ii) in the vertical field to less than 180 degrees or iii) in both the vertical and horizontal fields.

AMR, AMR-NB: Both names refer to the AMR codec (TS 26.071 [11]) and are used interchangeably in this specification.

Bitstream: A bitstream that conforms to a video or audio encoding format.

bitstream: A sequence of bits that forms the representation of one or more coded video or audio sequences.

CHEM: The Coverage and Handoff Enhancements using Multimedia error robustness feature.

Codec mode: Used for the AMR and AMR-WB codecs to identify one specific bitrate. For example AMR includes 8 codec modes (excluding SID), each of different bitrate.

Constrained terminal: UE that is (i) operating in radio access capability category series "M" capable of supporting conversational services, and/or (ii) a wearable device which is constrained in size, weight or power consumption (e.g. connected watches), excluding smartphones and feature phones.

DCMTSI client: A data channel capable MTSI client supporting data channel media as defined in clause 6.2.10.

DCMTSI client in terminal: A DCMTSI client that is implemented in a terminal or UE. The term "DCMTSI client in terminal" is used in this document when entities such as MRFP, MRFC or media gateways are excluded.

Dual-mono: A variant of 2-channel stereo encoding where two instances of a mono codec are used to encode a 2-channel stereo signal.

Evolved UTRAN: Evolved UTRAN is an evolution of the 3G UMTS radio-access network towards a high-data-rate, low-latency and packet-optimized radio-access network.

EVS codec: The EVS codec includes two operational modes: EVS Primary operational mode ('EVS Primary mode') and EVS AMR-WB Inter-Operable ('EVS AMR-WB IO mode'). When using EVS AMR-WB IO mode the speech frames are bitstream interoperable with the AMR-WB codec [18]. Frames generated by an EVS AMR-WB IO mode encoder can be decoded by an AMR-WB decoder, without the need for transcoding. Likewise, frames generated by an AMR-WB encoder can be decoded by an EVS AMR-WB IO mode decoder, without the need for transcoding.

EVS Primary mode: Includes 11 bit-rates for fixed-rate or multi-rate operation; 1 average bit-rate for variable bit-rate operation; and 1 bit-rate for SID (TS 26.441 [121]). The EVS Primary can encode narrowband, wideband, super-wideband and fullband signals. None of these bit-rates are interoperable with the AMR-WB codec.

EVS AMR-WB IO mode: Includes 9 codec modes and SID. All are bitstream interoperable with the AMR-WB codec (TS 26.171 [17]).

Field of View: The extent of visible area expressed with vertical and horizontal angles, in degrees in the 3GPP 3DOF reference system as defined in TS 26.118 [180].

Fisheye Video: Video captured by a wide-angle camera lens that usually captures an approximately hemispherical field of view and projects it as a circular image.

Frame Loss Rate (FLR): The percentage of speech frames not delivered to the decoder. FLR includes speech frames that are not received in time to be used for decoding.

ITT4RT client: MTSI client supporting the Immersive Teleconferencing and Telepresence for Remote Terminals (ITT4RT) feature, as defined in Annex Y.

ITT4RT-Tx client: ITT4RT client only capable of sending immersive video.

ITT4RT-Rx client: ITT4RT client only capable of receiving immersive video

ITT4RT MRF: An ITT4RT client implemented by functionality included in the MRFC and the MRFP.

ITT4RT client in terminal: An ITT4RT client that is implemented in a terminal or UE. The term "ITT4RT client in terminal" is used in this document when entities such as ITT4RT MRF is excluded.

Mode-set: Used for the AMR and AMR-WB codecs to identify the codec modes that can be used in a session. A mode-set can include one or more codec modes.

MSMTSI client: A multi-stream capable MTSI client supporting multiple streams as defined in Annex S. An MTSI client may support multiple streams, even of the same media type, without being an MSMTSI client. Such an MTSI client may, for example, add a second video to an ongoing video telephony session as shown in Annex A.11. In that case, the MTSI client is an MSMTSI client only if it is fully compliant with Annex S.

MSMTSI MRF: An MSMTSI client implemented by functionality included in the MRFC and the MRFP.

MSMTSI client in terminal: An MSMTSI client that is implemented in a terminal or UE. The term "MSMTSI client in terminal" is used in this document when entities such as MRFP, MRFC or media gateways are excluded.

MTSI client: A function in a terminal or in a network entity (e.g. a MRFP) that supports MTSI.

MTSI client in terminal: An MTSI client that is implemented in a terminal or UE. The term "MTSI client in terminal" is used in this document when entities such as MRFP, MRFC or media gateways are excluded.

MTSI media gateway (or MTSI MGW): A media gateway that provides interworking between an MTSI client and a non MTSI client, e.g. a CS UE. The term MTSI media gateway is used in a broad sense, as it is outside the scope of the current specification to make the distinction whether certain functionality should be implemented in the MGW or in the MGCF.

Omnidirectional media: Media such as image or video and its associated audio that enable rendering according to the user's viewing orientation, if consumed with a head-mounted device, or according to user's desired viewport, otherwise, as if the user was in the spot where and when the media was captured.

Operational mode: Used for the EVS codec to distinguish between EVS Primary mode and EVS AMR-WB IO mode.

Overlay: A piece of visual media, rendered over omnidirectional video or image, or a viewport.

Pose: Position and rotation information associated to a viewport.

Projected picture: Picture that has a representation format specified by an omnidirectional video projection format.

Projection: Inverse of the process by which the samples of a projected picture are mapped to a set of positions identified by a set of azimuth and elevation coordinates on a unit sphere.

Simulcast: Simultaneously sending different encoded representations (simulcast formats) of a single media source (e.g. originating from a single microphone or camera) in different simulcast streams.

Simulcast format: The encoded format used by a single simulcast stream, typically represented by an SDP format and all SDP attributes that apply to that particular SDP format, indicated in RTP by the RTP header payload type field.

Simulcast stream: The RTP stream carrying a single simulcast format in a simulcast.

Viewport: Region of omnidirectional image or video suitable for display and viewing by the user.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply:

NOTE: An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

| | |
|-----------|---|
| 3DOF | 3 Degrees of freedom |
| 5GC | 5G Core Network |
| AC | Alternating Current |
| AL-SDU | Application Layer - Service Data Unit |
| AMR | Adaptive Multi-Rate |
| AMR-NB | Adaptive Multi-Rate - NarrowBand |
| AMR-WB | Adaptive Multi-Rate - WideBand |
| AMR-WB IO | Adaptive Multi-Rate - WideBand Inter-operable Mode, included in the EVS codec |
| ANBR | Access Network Bitrate Recommendation |
| ANBRQ | Access Network Bitrate Recommendation Query |
| APP | APPLICATION-defined RTCP packet |
| ARQ | Automatic repeat ReQuest |
| AS | Application Server |
| ATCF | Access Transfer Control Function |
| ATGW | Access Transfer GateWay |
| AVC | Advanced Video Coding |
| BFCP | Binary Floor Control Protocol |
| CCM | Codec Control Messages |
| CDF | Cumulative Distribution Function |
| cDRX | Connected Mode DRX |
| CHEM | Coverage and Handoff Enhancements using Multimedia error robustness feature |
| CMP | Cube-Map Projection |
| CMR | Codec Mode Request |
| cps | characters per second |
| CS | Circuit Switched |
| CSCF | Call Session Control Function |
| CTM | Cellular Text telephone Modem |
| CVO | Coordination of Video Orientation |
| DBI | Delay Budget Information |
| DRB | Data Radio Bearer |
| DRX | Discontinuous Reception |
| DTLS | Datagram Transport Layer Security |
| DTMF | Dual Tone Multi-Frequency |
| DTX | Discontinuous Transmission |
| ECN | Explicit Congestion Notification |
| ECN-CE | ECN Congestion Experienced |
| ECT | ECN Capable Transport |
| eNodeB | E-UTRAN Node B |
| ERP | EquiRectangular Projection |
| E-UTRAN | Evolved UTRAN |
| EVS | Enhanced Voice Services |
| FECC | Far End Camera Control |
| FIR | Full Intra Request |
| FLR | Frame Loss Rate |
| FoIP | Facsimile over IP |

| | |
|-----------|--|
| FOV | Field Of View |
| GIP | Generic IP access |
| GOB | Group Of Blocks |
| H-ARQ | Hybrid - ARQ |
| HEVC | High Efficiency Video Coding |
| HMD | Head Mounted Display HSPA High Speed Packet Access |
| ICM | Initial Codec Mode |
| IDR | Instantaneous Decoding Refresh |
| IFP | Internet Facsimile Protocol |
| IFT | Internet Facsimile Transfer |
| IMS | IP Multimedia Subsystem |
| IP | Internet Protocol |
| IPv4 | Internet Protocol version 4 |
| IRAP | Intra Random Access Point |
| ITT4RT | Immersive Teleconferencing and Telepresence for Remote Terminals |
| ITU-T | International Telecommunications Union - Telecommunications |
| JBM | Jitter Buffer Management |
| MGCF | Media Gateway Control Function |
| MGW | Media GateWay |
| MIME | Multipurpose Internet Mail Extensions |
| MO | Management Object |
| MPEG | Moving Picture Experts Group |
| MRFC | Media Resource Function Controller |
| MRFP | Media Resource Function Processor |
| MSMTSI | Multi-Stream Multimedia Telephony Service for IMS |
| MSRP | Message Session Relay Protocol |
| MTSI | Multimedia Telephony Service for IMS |
| MTU | Maximum Transfer Unit |
| NACK | Negative ACKnowledgment |
| NNI | Network-to-Network Interface |
| NTP | Network Time Protocol |
| OMAF | Omnidirectional MediA Format |
| PCM | Pulse Code Modulation |
| PDCP | Packet Data Convergence Protocol |
| PDP | Packet Data Protocol |
| PLI | Picture Loss Indication |
| PLR | Packet Loss Ratio |
| POI | Point Of Interconnect |
| PSTN | Public Switched Telephone Network |
| PTZF | Pan, Tilt, Zoom and Focus |
| QCI | QoS Class Identifier |
| QMC | QoE Measurement Collection |
| QoE | Quality of Experience |
| QoS | Quality of Service |
| QP | Quantization Parameter |
| RoHC | Robust HeaderCompression |
| ROI | Region of Interest |
| RR | Receiver Report |
| RTCP | RTP Control Protocol |
| RTP | Real-time Transport Protocol |
| RWP | Region-Wise Packing |
| SB-ADPCM | Sub-Band Adaptive Differential PCM |
| SC-VBR | Source Controlled VBR |
| SCTP | Stream Control Transmission Protocol |
| SDAP | Service Data Adaptation Protocol |
| SDP | Session Description Protocol |
| SDPCapNeg | SDP Capability Negotiation |
| SEI | Supplemental Enhancement Information |
| SID | Silence Descriptor |
| SIP | Session Initiation Protocol |
| SR | Sender Report |
| SRVCC | Single Radio Voice Call Continuity |

| | |
|--------|---|
| TFO | Tandem-Free Operation |
| TISPAN | Telecoms and Internet converged Services and Protocols for Advanced Network |
| TMMBN | Temporary Maximum Media Bit-rate Notification |
| TMMBR | Temporary Maximum Media Bit-rate Request |
| TrFO | Transcoder-Free Operation |
| UDP | User Datagram Protocol |
| UDPTL | Facsimile UDP Transport Layer (protocol) |
| UE | User Equipment |
| VDP | Viewport Dependent Processing |
| VoIP | Voice over IP |
| VOP | Video Object Plane |
| VR | Virtual Reality |
| WebRTC | Web Real-Time Communication |

4 System description

4.1 System

A Multimedia Telephony Service for IMS call uses the Call Session Control Function (CSCF) mechanisms to route control-plane signalling between the UEs involved in the call (see figure 4.1). In the control plane, Application Servers (AS) should be present and may provide supplementary services such as call hold/resume, call forwarding and multi-party calls, etc.

The scope of the present document is to specify the media path. In the example in figure 4.1, it is routed directly between the PS Domains outside the IMS.

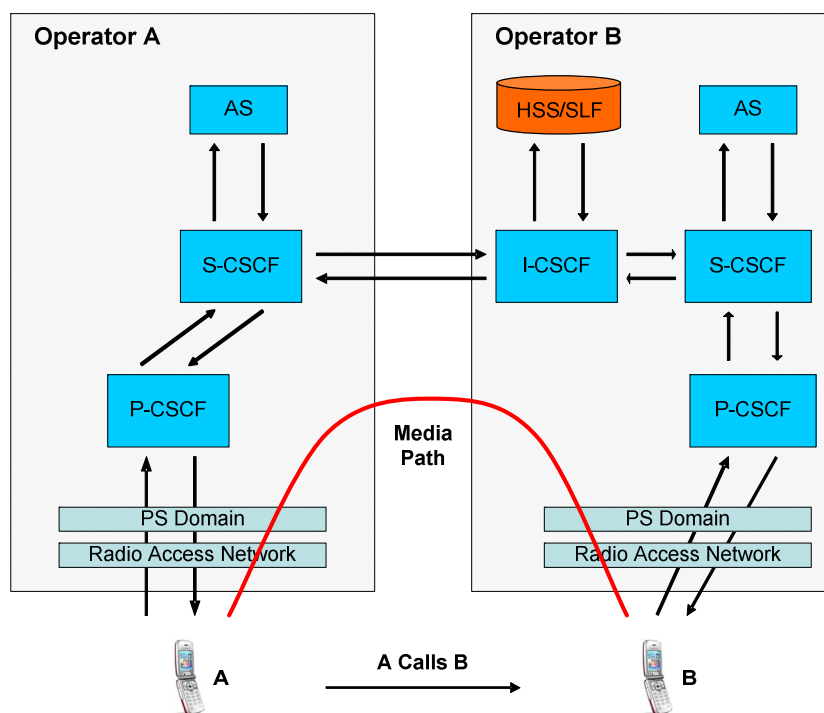
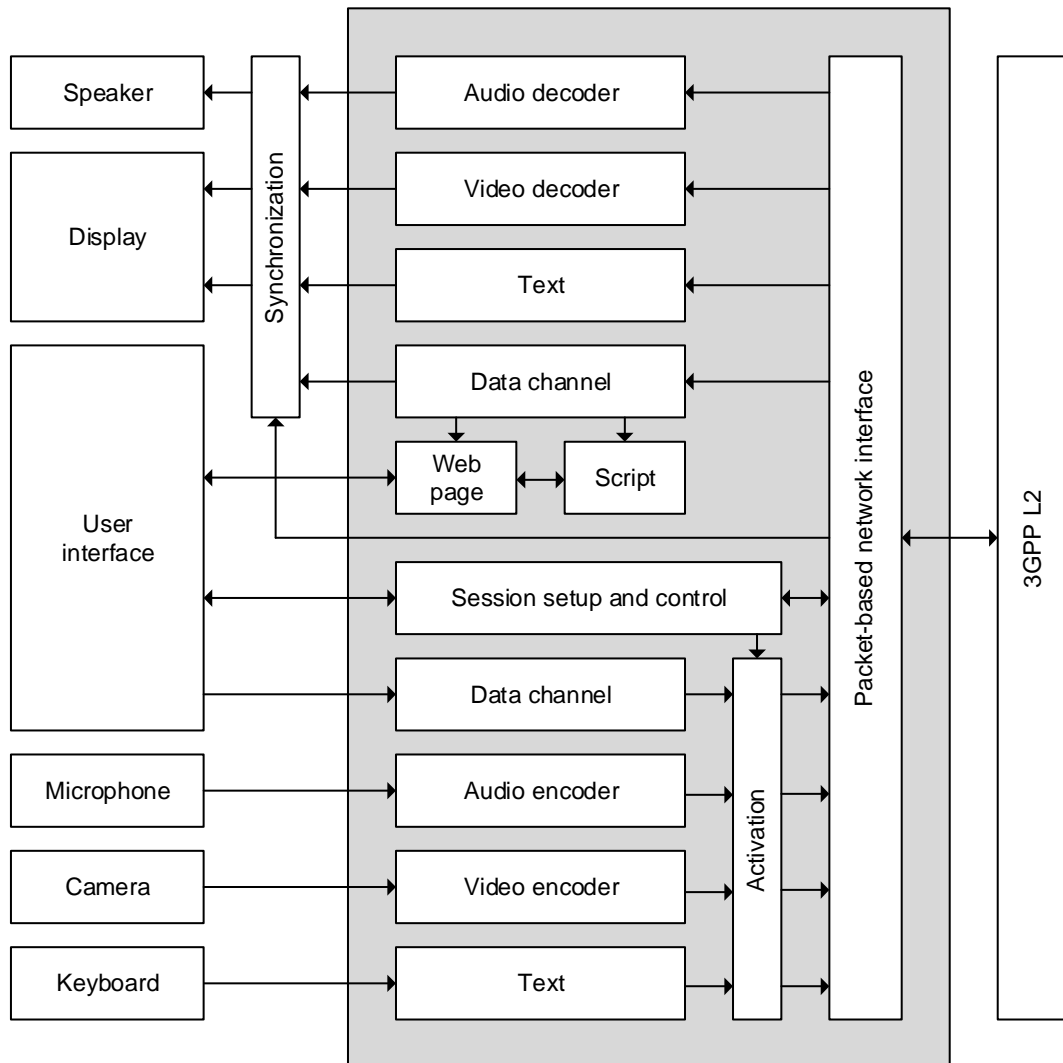


Figure 4.1: High-level architecture figure showing two MTSI clients in terminals using 3GPP access involved in an MTSI call set-up. The terminals connect to the IMS network over a 3GPP radio access network.

The call setup for an MTSI client in terminal using fixed access is the same as shown in Figure 4.1 for 3GPP terminals except that a fixed access is used instead of the 3GPP access and that the PS Domain is not necessarily used.

4.2 Client

The functional components of a terminal including an MTSI client in terminal using 3GPP access are shown in figure 4.2. An MTSI client in terminal using fixed access can have the same functional components except that it does not have any 3GPP Layer 2 protocol.



NOTE: The grey box marks the scope of the present document.

Figure 4.2: Functional components of a terminal including an MTSI client in terminal using 3GPP access

The scope of the present document is to specify media handling and interaction, which includes media control, media codecs, as well as transport of media and control data. General control-related elements of an MTSI client, such as SIP signalling (TS 24.229 [7]), fall outside this scope, albeit parts of the session setup handling and session control for conversational media are defined here:

- usage of SDP (RFC 4566 [8]) and SDP capability negotiation (RFC 5939 [69]) in SIP invitations for capability negotiation and media stream setup.
- set-up and control of the individual media streams between clients. It also includes interactivity, such as adding and dropping of media components.

Transport of media consists of the encapsulation of the coded media in a transport protocol as well as handling of coded media received from the network. This is shown in figure 4.2 as the "packet based network interface" and is displayed, for conversational media, in more detail in the user-plane protocol stack in figure 4.3. The basic MTSI client defined here specifies media codecs for speech, video and text (see clause 5). Data channels do not require use of any codec but

allows for real-time interaction in parallel to the conversational media (see clause 6.2.10). The User interface in Figure 4.2 interacts with a web page and a related script received through a downlink data channel to handle the data channel I/O and data formatting. All conversational media components are transported over RTP with each respective payload format mapped onto the RTP (RFC 3550 [9]) streams. The data channels are using SCTP (RFC 4960 [173]) over DTLS (RFC 8261 [174]), used as specified for WebRTC data channels (RFC 8831 [175]).

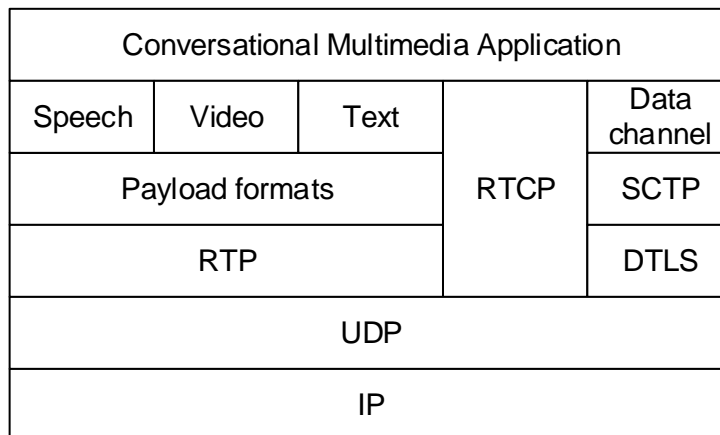


Figure 4.3: User plane protocol stack for a basic MTSI client

An MTSI client may also support non-conversational media, for example IMS messaging. The functional entities and the protocols used for IMS messaging are described in TS 24.247 [82].

The 3GPP Layer 2 protocol to be interfaced with MTSI client is PDCP [170] for EPC. For 5GC, another user-plane protocol, SDAP [171], is used on top of PDCP as shown in clause 4.4.1 of [164]. It is assumed that the SDAP would be configured without header for both directions in the typical MTSI cases, effectively interfacing with PDCP, as SDAP header would be needed only when more than one QoS flows are multiplexed in a DRB or reflective mapping is enabled.

4.3 MRFP and MGW

A Media Resource Function Processor (MRFP), see TS.23.002 [47], may be inserted in the media path for certain supplementary services (e.g. conference) and/or to provide transcoding and may therefore act as a MTSI client together with other network functions, such as a MRFC.

A Media Gateway (MGW), see TS 23.002 [47], may be used to provide inter-working between different networks and services. For example, a MTSI MGW may provide inter-working between MTSI and 3G-324M services. The MTSI MGW may have more limited functionality than other MTSI clients, e.g. when it comes to the supported bitrates of media. The inter-working aspects are described in more detail in clause 12.

5 Media codecs

5.1 Media components

The Multimedia Telephony Service for IMS supports simultaneous transfer of multiple media components with real-time characteristics. Media components denote the actual components that the end-user experiences.

The following media components are considered as core components. Multiple media components (including media components of the same media type) may be present in a session. At least one of the first three of these components is present in all conversational multimedia telephony sessions.

- **Speech:** The sound that is picked up by a microphone and transferred from terminal A to terminal B and played out in an earphone/loudspeaker. Speech includes detection, transport and generation of DTMF events.
- **Video:** The moving image that is, for example, captured by a camera of terminal A, transmitted to terminal B and, for example, rendered on the display of terminal B.

- **Text:** The characters typed on a keyboard or drawn on a screen on terminal A and rendered in real time on the display of terminal B. The flow is time-sampled so that no specific action is needed from the user to request transmission.
- **Data:** Any other data for real-time interaction, closely related to the multimedia telephony session that may be generated or consumed by either one of terminal A or terminal B, possibly via terminal external connections and/or physical connectors, optionally processed by application-specific logic at one or both terminals, and optionally presented on and controlled by the user interface at one or both terminals.

The first three of the above core media components are transported in real time from one MTSI client to the other using RTP (IETF RFC 3550 [9]). The "data" media component for real-time interaction is transported using SCTP (IETF RFC 4960 [173]) over DTLS (IETF RFC 8261 [174]), as described by WebRTC data channels [175]. All media components can be added or dropped during an ongoing session as required either by the end-user or by controlling nodes in the network, assuming that when adding components, the capabilities of the MTSI client support the additional component.

NOTE: The terms voice and speech are synonyms. The present document uses the term speech. The media type is called "audio" in SDP and therefore also the term "audio" is used as synonym.

MTSI specifications also support other media types than the core components described above, for example facsimile (fax) transmission.

Facsimile transmission is described in Annex L.

5.2 Codecs for MTSI clients in terminals

5.2.1 Speech

5.2.1.1 General codec requirements

MTSI clients in terminals offering speech communication shall support narrowband, wideband and super-wideband communication. The only exception to this requirement is for the MTSI client in constrained terminal offering speech communication, in which case the MTSI client in constrained terminal shall support narrowband and wideband, and should support super-wideband communication.

In addition, MTSI clients in terminals offering speech communication shall support:

- .AMR speech codec (TS 26.071 [11], TS 26.090 [12], TS 26.073 [13] and TS 26.104 [14]) including all 8 modes and source controlled rate operation TS 26.093 [15]. The MTSI client in terminal shall be capable of operating with any subset of these 8 codec modes. More detailed codec requirements for the AMR codec are defined in clause 5.2.1.2.

MTSI clients in terminals offering wideband speech communication at 16 kHz sampling frequency shall support:

- AMR-WB codec (TS 26.171 [17], TS 26.190 [18], TS 26.173 [19] and TS 26.204 [20]) including all 9 modes and source controlled rate operation TS 26.193 [21]. The MTSI client in terminal shall be capable of operating with any subset of these 9 codec modes. More detailed codec requirements for the AMR-WB codec are defined in clause 5.2.1.3. When the EVS codec is supported, the EVS AMR-WB IO mode may serve as an alternative implementation of AMR-WB as defined in clause 5.2.1.4.

MTSI clients in terminals offering super-wideband or fullband speech communication shall support:

- EVS codec (TS 26.441 [121], TS 26.444 [124], TS 26.445 [125], TS 26.447 [127], TS 26.451 [131], TS 26.442 [122], TS 26.452 [165] and TS 26.443 [123]) as described below including functions for backwards compatibility with AMR-WB (TS 26.446 [126]) and discontinuous transmission (TS 26.449 [129] and TS 26.450 [130]). More detailed codec requirements for the EVS codec are defined in clause 5.2.1.4.

Encoding of DTMF is described in Annex G.

5.2.1.2 Detailed codec requirements, AMR

When transmitting, the MTSI client in terminal shall be capable of aligning codec mode changes to every frame border, and shall also be capable of restricting codec mode changes to be aligned to every other frame border, e.g. like UMTS_AMR_2 (TS 26.103 [16]). The MTSI client in terminal shall also be capable of restricting codec mode changes to neighbouring codec modes within the negotiated codec mode set. When receiving, the MTSI client in terminal shall allow codec mode changes at any frame border and to any codec mode within the negotiated codec mode set.

The codec modes and the other codec parameters (mode-change-capability, mode-change-period, mode-change-neighbor, etc), applicable for each session, are negotiated as described in clauses 6.2.2.2 and 6.2.2.3.

5.2.1.3 Detailed codec requirements, AMR-WB

When transmitting, the MTSI client in terminal shall be capable of aligning codec mode changes to every frame border, and shall also be capable of restricting codec mode changes to be aligned to every other frame border, e.g. like UMTS_AMR_WB (TS 26.103 [16]). The MTSI client in terminal shall also be capable of restricting codec mode changes to neighbouring codec modes within the negotiated codec mode set. When receiving, the MTSI client in terminal shall allow codec mode changes at any frame border and to any codec mode within the negotiated codec mode set.

The codec modes and the other codec parameters (mode-change-capability, mode-change-period, mode-change-neighbor, etc), applicable for each session, are negotiated as described in clauses 6.2.2.2 and 6.2.2.3.

5.2.1.4 Detailed codec requirements, EVS

When the EVS codec is supported, the MTSI client in terminal may support dual-mono encoding and decoding.

When the EVS codec is supported, EVS AMR-WB IO may serve as an alternative implementation of the AMR-WB codec, [125]. In this case, the requirements and recommendations defined in this specification for the AMR-WB codec also apply to EVS AMR-WB IO.

NOTE: The DTX operation of EVS Primary and AMR-WB IO can be configured in sending direction with either a fixed SID update interval (from 3 to 100 frames) or an adaptive SID update interval - more details can be found in clauses 4.4.3 and 5.6.1.1 of TS 26.445 [125]. Implementers of MTSI clients are advised to take into account this SID flexibility of EVS.

5.2.1.5 Offering multiple audio bandwidths and multiple channels

MTSI clients in terminals offering wideband speech communication shall also offer narrowband speech communications.

When offering super-wideband speech, both wideband speech and narrowband speech shall also be offered. When offering fullband speech, super-wideband speech, wideband speech and narrowband speech shall also be offered.

MTSI clients in terminals offering dual-mono, shall also offer mono.

5.2.1.6 Codec preference order

When offering both wideband speech and narrowband speech communication, payload types offering wideband shall be listed before payload types offering only narrowband speech in the 'm=' line of the SDP offer (RFC 4566 [8]).

When offering super-wideband speech, wideband and narrowband speech communication, payload types offering super-wideband shall be listed before payload types offering lower bandwidths than super-wideband speech in the 'm=' line of the SDP offer (RFC 4566 [8]).

For an MTSI client in terminal supporting EVS the following rules apply when creating the list of payload types on the m= line:

- When the EVS codec is offered for NB by an MTSI client in terminal supporting NB only, it shall be listed before other NB codecs.
- When the EVS codec is offered for up to WB, it shall be listed before other WB codecs.

When dual-mono is offered then this may be preferable over mono depending on the call scenario.

5.2.2 Video

MTSI clients in terminals offering video communication shall support:

- H.264 (AVC) [24] Constrained Baseline Profile (CBP) Level 1.2;
- H.265 (HEVC) [119] Main Profile, Main Tier, Level 3.1. The only exception to this requirement is for the MTSI client in constrained terminal offering video communication, in which case the MTSI client in constrained terminal should support H.265 (HEVC) Main Profile, Main Tier, Level 3.1.

In addition they should support:- H.264 (AVC) [24] Constrained High Profile (CHP) Level 3.1.

For backwards compatibility to previous releases, if H.264 (AVC) [24] Constrained High Profile Level 3.1 is supported, then H.264 (AVC) [24] Constrained Baseline Profile (CBP) Level 3.1 should also be offered.

H.264 (AVC) shall be used without requirements on output timing conformance (annex C of [24]). Each sequence parameter set of H.264 (AVC) shall contain the vui_parameters syntax structure including the num_reorder_frames syntax element set equal to 0.

H.265 (HEVC) Main Profile shall be used with general_progressive_source_flag equal to 1, general_interlaced_source_flag equal to 0, general_non_packed_constraint_flag equal to 1, general_frame_only_constraint_flag equal to 1, and sps_max_num_reorder_pics[i] equal to 0 for all i in the range of 0 to sps_max_sub_layers_minus1, inclusive, without requirements on output timing conformance (annex C of [119]).

For both H.264 (AVC) and H.265 (HEVC), the decoder needs to know the Sequence Parameter Set (SPS) and the Picture Parameter Set (PPS) to be able to decode the received video packets. A compliant H.265 (HEVC) bitstream must include a Video Parameter Set (VPS), although the VPS may be ignored by the decoder in the context of the present specification. When H.264 (AVC) or H.265 (HEVC) is used it is recommended to transmit the parameter sets within the SDP description of a stream, using the relevant MIME/SDP parameters as defined in RFC6184 [25] for H.264 (AVC) and in [120] for H.265 (HEVC), respectively. Each media source (SSRC) shall transmit the currently used parameter sets at least once in the beginning of the RTP stream before being referenced by the encoded video data to ensure that the parameter sets are available when needed by the receiver. If the video encoding is changed during an ongoing session such that the previously used parameter set(s) are no longer sufficient then the new parameter sets shall be transmitted at least once in the RTP stream prior to being referenced by the encoded video data to ensure that the parameter sets are available when needed by the receiver. When a specific version of a parameter set is sent in the RTP stream for the first time, it should be repeated at least 3 times in separate RTP packets with a single copy per RTP packet and with an interval not exceeding 0.5 seconds to reduce the impact of packet loss. A single copy of the currently active parameter sets shall also be part of the data sent in the RTP stream as a response to FIR. Moreover, it is recommended to avoid using a sequence or picture parameter set identifier value during the same session to signal two or more parameter sets of the same type having different values, such that if a parameter set identifier for a certain type is used more than once in either SDP description or RTP stream, or both, the identifier always indicates the same set of parameter values of that type.

The video decoder in a multimedia MTSI client in terminal shall either start decoding immediately when it receives data, even if the stream does not start with an IDR/IRAP access unit (IDR access unit for H.264, IRAP access unit for H.265) or alternatively no later than it receives the next IDR/IRAP access unit or the next recovery point SEI message, whichever is earlier in decoding order. The decoding process for a stream not starting with an IDR/IRAP access unit shall be the same as for a valid video bit stream. However, the MTSI client in terminal shall be aware that such a stream may contain references to pictures not available in the decoded picture buffer. The display behaviour of the MTSI client in terminal is out of scope of the present document.

An MTSI client in terminal offering H.264 (AVC) CBP support at a level higher than Level 1.2 shall support negotiation to use a lower Level as described in [25] and [58].

An MTSI client in terminal offering H.264 (AVC) CHP support at a level higher than Level 3.1 shall support negotiation to use a lower Level as described in [25] and [58].

An MTSI client in terminal offering video support shall include in the SDP offer H.264 CBP at Level 1.2 or higher.

An MTSI client in terminal offering video support for H.265 (HEVC) [119] Main Profile, Main Tier, Level 3.1, should normally set it to be preferred.

An MTSI client in terminal offering H.265 (HEVC) shall support negotiation to use a lower Level than the one in the offer, as described in [120] and [58].

If a codec is supported at a certain level, then all (hierarchically) lower levels shall be supported as well.

NOTE 1: An example of a lower level than Level 1.2 is Level 1 for H.264 (AVC) Constrained Baseline Profile.

NOTE 2: All levels are minimum requirements. Higher levels may be supported and used for negotiation.

NOTE 3: MTSI clients in terminals may use full-frame freeze and full-frame freeze release SEI messages of H.264 (AVC) to control the display process. For H.265 (HEVC), MTSI clients may set the value of `pic_output_flag` in the slice segment headers to either 0 or 1 to control the display process.

NOTE 4: An H.264 (AVC) encoder should code redundant slices only if it knows that the far-end decoder makes use of this feature (which is signalled with the `redundant-pic-cap` MIME/SDP parameter as specified in RFC 6184 [25]). H.264 (AVC) encoders should also pay attention to the potential implications on end-to-end delay. The redundant slice header is not supported in H.265 (HEVC).

NOTE 5: If a codec is supported at a certain level, it implies that on the receiving side, the decoder is required to support the decoding of bitstreams up to the maximum capability of this level. On the sending side, the support of a particular level does not imply that the encoder will produce a bitstream up to the maximum capability of the level. This method can be used to set up an asymmetric video stream. For H.264 (AVC), another method is to use the SDP parameters `'level-asymmetry-allowed'` and `'max-recv-level'` that are defined in the H.264 payload format specification, [25]. For H.265 (HEVC) it is possible to use the SDP parameter `'max-recv-level-id'` defined in the H.265 payload format specification, [120], to indicate a higher level in the receiving direction than in the sending direction. See also clause 6.2.3.2, Annex A.4.5 for SDP examples with asymmetric video using H.264 (AVC) and Annex A.4.8 for SDP examples with asymmetric video using both H.264 (AVC) and H.265 (HEVC). Other methods for asymmetric video transmission are also possible.

NOTE 6: If video is used in a session, an MTSI client in terminal should offer at least one video stream with a picture aspect ratio in the range from 0.7 to 1.4. For all offered video streams, the width and height of the picture should be integer multiples of 16 pixels. For example, 224x176, 272x224, and 320x240 are image sizes that satisfy these conditions.

NOTE 7: For H.264 (AVC) and H.265 (HEVC), respectively, multiple sequence and picture parameter sets can be defined, as long as they have unique parameter set identifiers, but only one sequence and picture parameter set can be active between two consecutive IDRs and IRAPs, respectively.

NOTE 8: For H.264 (AVC), Constrained High Profile (CHP) Level 3.1 is not required to be supported as it is less bit rate efficient than H.265 (HEVC) Main Profile, Main Tier, Level 3.1. However, it is recommended for interoperability.

5.2.3 Real-time text

MTSI clients in terminals offering real time text conversation shall support:

- ITU-T Recommendation T.140 [26] and [27].

T.140 specifies coding and presentation features of real-time text usage. Text characters are coded according to the UTF-8 transform of ISO 10646-1 (Unicode).

A minimal subset of the Unicode character set, corresponding to the Latin-1 part shall be supported, while the languages in the regions where the MTSI client in terminal is intended to be used should be supported.

Presentation control functions from ISO 6429 are allowed in the T.140 media stream. A mechanism for extending control functions is included in ITU-T Recommendation T.140 [26] and [27]. Any received non-implemented control code must not influence presentation.

A MTSI client in terminal shall store the conversation in a presentation buffer during a call for possible scrolling, saving, display re-arranging, erasure, etc. At least 800 characters shall be kept in the presentation buffer during a call.

Note that erasure (backspace) of characters is included in the T.140 editing control functions. It shall be possible to erase all characters in the presentation buffer. The display of the characters in the buffer shall also be impacted by the erasure.

5.2.4 Still Images

MTSI clients supporting still images shall support HEVC encoded images conforming to the HEVC bitstream requirements of 5.2.2.

Still images encoded using the HEVC shall have `general_progressive_source_flag` equal to 1, `general_interlaced_source_flag` equal to 0, `general_non_packed_constraint_flag` equal to 1, `general_frame_only_constraint_flag` equal to 1.

For HEVC encoded images/image sequence, the display properties are carried as SEI and VUI within the bitstream, and the RTP timestamps determine the presentation time of the images.

6 Media configuration

6.1 General

MTSI uses SIP, SDP and SDPCapNeg for media negotiation and configuration. General SIP signalling and session setup for IMS are defined in TS 24.229 [7], whereas this clause specifies SDP and SDPCapNeg usage and media handling specifically for MTSI, including offer/answer considerations in the capability negotiation. The MTSI client in the terminal may use the OMA-DM solution specified in Clause 15 for enhancing SDP negotiation and resource reservation process.

The support for ECN [83] in E-UTRAN is specified in [85]. The support for ECN in UTRA/HSPA is specified in [89]. The support of ECN in NR is specified in [164]. MTSI may use Explicit Congestion Notification (ECN) to perform rate adaptation for speech and video. When the MTSI client in terminal supports, and the session allows, adapting the media encoding at multiple bit rates and the radio access bearer technology is known to the MTSI client to be E-UTRAN or UTRA/HSPA, the MTSI client may negotiate the use of ECN [83] to perform ECN triggered media bit-rate adaptation. An MTSI MGW supporting ECN supports ECN in the same way as the MTSI client in terminal as described in clauses 12.3.3 and 12.7.3.

The support of ECN is optional for both MTSI client in terminal and MTSI MGW.

It is assumed that the network properly handles ECN-marked packets as described in [84] end-to-end between the MTSI clients in terminals.

An MTSI MGW can be used for inter-working with:

- a client that does not use ECN;
- a client that supports ECN in different way than what is specified for MTSI clients;
- a CS network;
- a network which does not handle ECN-marked packets properly.

In such cases, the ECN protocol, as specified for MTSI clients, is terminated in the MTSI MGW.

6.2 Session setup procedures

6.2.1 General

The session setup for RTP transported media shall determine for each media: IP address(es), RTP profile, UDP port number(s); codec(s); RTP Payload Type number(s), RTP Payload Format(s). The session setup may also determine: ECN usage and any additional session parameters.

The session setup for UDP transported media without RTP shall determine: IP address(es), UDP port number(s) and additional session parameters.

The session setup for data channel (SCTP over DTLS over UDP transported) media shall determine for each media: IP address(es), UDP port number(s), SCTP port number(s), DTLS server/client role(s), DTLS ID(s), DTLS certificate fingerprint(s), and ICE-related information for data channel media as described in clause 6.2.10. The session setup may also determine use of ICE Lite for data channel media and may determine additional session parameters.

The session setup for RTP and data channel transported media shall, when the port number is not set to zero, determine the maximum bandwidth that is allowed in the session, see also clause 6.2.5. The maximum bandwidth for the receiving direction is specified with the "b=AS" bandwidth modifier. Additional requirements and/or recommendations on the bandwidth negotiation are found in clause 6.2.2.1 for speech, in clause 6.2.3.2 for video, and in clause 6.2.10 for data channel.

An MTSI client shall offer at least one RTP profile for each RTP media stream. Multiple RTP profiles may be offered using SDPCapNeg as described in Clause 6.2.1a. For voice and real-time text, the first SDP offer shall include at least the AVP profile. For video, the first SDP offer for a media type shall include at least the AVPF profile. Subsequent SDP offers may include only other RTP profiles if it is known from a preceding offer that this RTP profile is supported by the answerer. The MTSI client shall be capable of receiving an SDP offer containing both AVP and AVPF offers in order to support interworking.

The configuration of ECN for media transported with RTP is described in clause 6.2.2 for speech and in clause 6.2.3.2 for video. The negotiation of ECN at session setup is described in [84]. The adaptation response to congestion events is described in clause 10.

6.2.1a RTP profile negotiation

6.2.1a.1 General

MTSI clients should support SDPCapNeg to be able to negotiate RTP profiles for all media types where AVPF is supported. MTSI clients supporting SDPCapNeg shall support the complete SDPCapNeg framework.

SDPCapNeg is described in [69]. This clause only describes the SDPCapNeg attributes that are directly applicable for the RTP profile negotiation, i.e. the *tcap*, *pcfg* and *acfg* attributes. TS 24.229 [7] may outline further requirements needed for supporting SDPCapNeg in SDP messages.

NOTE: This clause describes only how to use the SDPCapNeg framework for RTP profile negotiation using the *tcap*, *pcfg* and *acfg* attributes. Implementers may therefore (incorrectly) assume that it is sufficient to implement only those specific parts of the framework that are needed for RTP profile negotiation. Doing so would however not be future proof since future versions may use other parts of the framework and there are currently no mechanisms for declaring that only a subset of the framework is supported. Hence, MTSI clients are required to support the complete framework.

6.2.1a.2 Using SDPCapNeg in SDP offer

For voice and real-time text, SDPCapNeg shall be used when offering AVPF the first time for a new media type in the session since the support for AVPF in the answering client is not known at this stage. For video, an MTSI client shall either offer AVPF and AVP together using SDPCapNeg, or the MTSI client shall offer only AVPF without using SDPCapNeg. If an MTSI client has offered only AVPF for video, and then receives as response either an SDP answer where the video media component has been rejected, or an SIP 488 or 606 failure response with an SDP body indicating that only AVP is supported for video media, the MTSI client should send a new SDP offer with AVP as transport for video. Subsequent SDP offers, in a re-INVITE or UPDATE, may offer AVPF without SDPCapNeg if it is known from an earlier re-INVITE or UPDATE that the answering client supports this RTP profile. If the offer includes only AVP then SDPCapNeg does not need to be used, which can occur for: text; speech if RTCP is not used; and in re-INVITES or UPDATES where the RTP profile has already been negotiated for the session in a preceding INVITE or UPDATE.

When offering AVP and AVPF using SDPCapNeg, the MTSI client shall offer AVP on the media (m=) line and shall offer AVPF using SDPCapNeg mechanisms. The SDPCapNeg mechanisms are used as follows:

- The support for AVPF is indicated in an attribute (a=) line using the transport capability attribute 'tcap'. AVPF shall be preferred over AVP.

- At least one configuration using AVPF shall be listed using the attribute for potential configurations 'pcfg'.

6.2.1a.3 Answering to an SDP offer using SDPCapNeg

An invited MTSI client should accept using AVPF whenever supported. If AVPF is to be used in the session then the MTSI client:

- Shall select one configuration out of the potential configurations defined in the SDP offer for using AVPF.
- Indicate in the media (m=) line of the SDP answer that the profile to use is AVPF.
- Indicate the selected configuration for using AVPF in the attribute for actual configurations 'acfg'.

If AVP is to be used then the MTSI shall not indicate any SDPCapNeg attributes for using AVPF in the SDP answer.

6.2.2 Speech

6.2.2.1 General

For AMR or AMR-WB encoded media, the session setup shall determine the applicable bandwidth(s) as defined in clause 6.2.5, what RTP profile to use; if all codec modes can be used or if the operation needs to be restricted to a subset; if the bandwidth-efficient payload format can be used or if the octet-aligned payload format must be used; if codec mode changes shall be restricted to be aligned to only every other frame border or if codec mode changes can occur at any frame border; if codec mode changes must be restricted to only neighbouring modes within the negotiated codec mode set or if codec mode changes can be performed to any mode within the codec mode set; the number of speech frames that should be encapsulated in each RTP packet and the maximum number of speech frames that may be encapsulated in each RTP packet. For EVS encoded media, the session setup shall determine the RTP profile to use in the session.

If the session setup negotiation concludes that multiple configuration variants are possible in the session then the default operation should be used as far as the agreed parameters allow, see clause 7.5.2.1. It should be noted that the default configurations are slightly different for different access types.

An MTSI client offering a speech media session for narrow-band speech and/or wide-band speech should generate an SDP offer according to the examples in Annexes A.1 to A.3. An MTSI client offering EVS should generate an SDP offer according to the examples in Annex A.14.

An MTSI client in terminal supporting EVS should support the RTCP-APP signalling for speech adaptation defined clause 10.2.1, and shall support the RTCP-APP signalling when the MTSI client in terminal supports adaptation for call cases where the RTP-based CMR cannot be used.

NOTE 1: Examples of call cases where the RTP-based CMR cannot be used are: when the RTP-based CMR is disabled; or for uni-directional media (sendonly or recvonly).

Some of the request messages are generic for all speech codecs while other request messages are codec-specific. Request messages that can be used in a session are negotiated in SDP, see clause 10.2.3.

An MTSI client shall at least offer AVP for speech media streams. An MTSI client should also offer AVPF for speech media streams. An MTSI client shall offer AVPF for speech media streams when offering to use RTCP-APP signalling. RTP profile negotiation shall be done as described in clause 6.2.1a. When AVPF is offered then the RTCP bandwidth shall be greater than zero.

If an MTSI client in terminal offers to use ECN for speech in RTP streams then the MTSI client in terminal shall offer ECN Capable Transport as defined below. If an MTSI client in terminal accepts an offer for ECN for speech then the MTSI client in terminal shall declare ECN Capable Transport in the SDP answer as defined below. The SDP negotiation of ECN Capable Transport is described in [84].

ECN shall not be used when the codec negotiation concludes that only fixed-rate operation is used.

An MTSI client may support multiple codecs where ECN-triggered adaptation is supported only for some of the codecs. An SDP offer for ECN may therefore include multiple codecs where ECN-triggered adaptation is supported only for some of the codecs. An MTSI client receiving an SDP offer including multiple codecs and an offer for ECN should first select which codec to accept and then accept or reject the offer for ECN depending on whether ECN-triggered

adaptation is supported for that codec or not. An MTSI client receiving an SDP answer accepting ECN for a codec where ECN-triggered adaptation is not supported should re-negotiate the session to disable ECN.

NOTE 2: ECN-triggered adaptation is currently undefined for EVS. This does not prevent ECN-triggered adaptation from being negotiated and used for AMR or AMR-WB.

The use of ECN for a speech stream in RTP is negotiated with the 'ecn-capable-rtp' SDP attribute, [84]. ECN is enabled when both clients agree to use ECN as configured below. An MTSI client in terminal using ECN shall therefore also include the following parameters and parameter values for the ECN attribute:

- 'leap', to indicate that the leap-of-faith initiation method shall be used;
- 'ect=0', to indicate that ECT(0) shall be set for every packet.

An MTSI client offering ECN for speech may indicate support of the RTCP AVPF ECN feedback messages [84] using "rtcp-fb" attributes with the "nack" feedback parameter and the "ecn" feedback parameter value. An MTSI client offering ECN for speech may indicate support for RTCP XR ECN summary reports [84] using the "rtcp-xr" SDP attribute [88] and the "ecn-sum" parameter.

An MTSI client receiving an offer for ECN for speech without an indication of support of RTCP AVPF ECN feedback messages [84] within an "rtcp-fb" attribute should accept the offer if it supports ECN.

An MTSI client receiving an offer for ECN for speech with an indication of support of the RTCP AVPF ECN feedback message [84] should also accept the offer and may indicate support of the RTCP AVPF ECN feedback messages [84] in the answer.

An MTSI client accepting ECN for speech in an answer may indicate support for RTCP XR ECN summary reports in the answer using the "rtcp-xr" SDP attribute [88] and the "ecn-sum" parameter.

The use of ECN is disabled when a client sends an SDP without the 'ecn-capable-rtp' SDP attribute.

An MTSI client may initiate a session re-negotiation to disable ECN to resolve ECN-related error cases. An ECN-related error case may, for example, be detecting non-ECT in the received packets when ECT(0) was expected or detecting a very high packet loss rate when ECN is used.

SDP examples for offering and accepting ECT are shown in Annex A.12.

Session setup for sessions including speech and DTMF events is described in Annex G.

6.2.2.2 Generating SDP offers

When speech is offered, an MTSI client in terminal sending a first SDP offer in the initial offer-answer negotiation shall include at least one RTP payload type for AMR-NB according to RFC4867 [28] and the MTSI client in terminal shall support and offer a configuration, where the MTSI client in terminal includes the parameter settings as defined in Table 6.1. When EVS-NB is also offered, the MTSI client in terminal shall support and offer a configuration, where the MTSI client in terminal includes the parameter settings for EVS primary mode as defined in Table 6.2a and the RTP payload type for EVS shall also use the parameters for EVS AMR-WB IO mode as defined in Table 6.1, except for the 'ecn-capable-rtp' and 'leap ect' parameters.

If wideband speech is also offered, then the SDP offer shall also include at least one RTP payload type for AMR-WB according to RFC4867 [28] and the MTSI client in terminal shall support and offer a configuration, where the MTSI client in terminal includes the parameter settings as defined in Table 6.1. When EVS-WB is also offered, the MTSI client in terminal shall support and offer a configuration, where the MTSI client in terminal includes the parameter settings for EVS primary mode as defined in Table 6.2a and the RTP payload type for EVS shall also use parameters for EVS AMR-WB IO mode as defined in Table 6.1, except for the 'ecn-capable-rtp' and 'leap ect' parameters. AMR-WB and EVS (including the EVS AMR-WB IO mode) are thus offered using different RTP payload types.

If super-wideband speech is also offered, the SDP offer shall include at least one RTP payload type for EVS and the MTSI client in terminal shall support a configuration where the MTSI client in terminal includes the parameter settings as defined in Table 6.2a.

If fullband speech is also offered, the SDP offer shall include at least one RTP payload type for EVS and the MTSI client in terminal shall support a configuration where the MTSI client in terminal includes the parameter settings as defined in Table 6.2a.

When EVS is offered, the RTP payload type for EVS shall also use parameters for EVS AMR-WB IO mode as defined in Table 6.1, except for the 'ecn-capable-rtp' and 'leap ect' parameters. AMR-WB and EVS (including the EVS AMR-WB IO mode) are thus offered using different RTP payload types.

NOTE 1: RFC4867 can also be used for EVS AMR-WB IO when EVS is supported. This may happen after SRVCC when the EVS payload format is used between the ATGW and the MTSI client in terminal while RFC4867 is used between the CS-MGW and the ATGW.

NOTE 2: ECN-triggered adaptation is currently undefined for EVS. This does not prevent ECN-triggered adaptation from being negotiated and used for AMR or AMR-WB.

NOTE 3: When EVS is offered, the audio bandwidths may be different for different directions for the EVS Primary mode, even for 'sendrecv' media.

Clause 5.2.1.6 describes the preference order for how different configurations should be ordered in the list of payload type numbers that is given on the m= line.

Table 6.1: SDP parameters for AMR-NB or AMR-WB, when the MTSI client in terminal offers the bandwidth-efficient payload format

| Parameter | Usage |
|-----------------------------|--|
| octet-align | Shall not be included |
| mode-set | Shall not be included |
| mode-change-period | Shall not be included |
| mode-change-capability | Shall be set to 2 |
| mode-change-neighbor | Shall not be included |
| maxptime | Shall be set to 240, see also Table 7.1 |
| crc | Shall not be included |
| robust-sorting | Shall not be included |
| interleaving | Shall not be included |
| ptime | Shall be set according to Table 7.1 |
| channels | Shall either be set to 1 or be omitted |
| max-red | Shall be included and shall be set to 220 or less |
| ecn-capable-rtp: leap ect=0 | Shall be included if offering to use ECN and if the session setup allows for bit-rate adaptation |

Table 6.2: SDP parameters for AMR-NB or AMR-WB, when the MTSI client in terminal offers the octet-aligned payload format

| Parameter | Usage |
|-----------------------------|--|
| octet-align | Shall be set to 1 |
| mode-set | Shall not be included |
| mode-change-period | Shall not be included |
| mode-change-capability | Shall be set to 2 |
| mode-change-neighbor | Shall not be included |
| maxptime | Shall be set to 240, see also Table 7.1 |
| crc | Shall not be included |
| robust-sorting | Shall not be included |
| interleaving | Shall not be included |
| ptime | Shall be set according to Table 7.1 |
| channels | Shall either be set to 1 or be omitted |
| max-red | Shall be included and shall be set to 220 or less |
| ecn-capable-rtp: leap ect=0 | Shall be included if offering to use ECN and if the session setup allows for bit-rate adaptation |

Table 6.2a: SDP parameters for EVS Primary mode, when the MTSI client in terminal offers EVS

| Parameter | Usage |
|-----------------|--|
| ptime | Shall be set according to Table 7.1 |
| maxptime | Shall be set to 240, see also Table 7.1 |
| dtx | MTSI client in terminal shall not include dtx in the initial SDP offer. |
| dtx-recv | MTSI client in terminal shall not include dtx-recv. |
| hf-only | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| evs-mode-switch | MTSI client in terminal shall not include evs-mode-switch in the initial SDP offer. |
| br | An MTSI client in terminal supporting the EVS codec is required to support the entire bit-rate range but may offer a smaller bit-rate range or even a single bit-rate. |
| br-send | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| br-recv | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| bw | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| bw-send | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| bw-recv | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| ch-send | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| ch-recv | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| cmr | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| ch-aw-recv | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| channels | The SDP offer-answer considerations in TS 26.445 [125] apply. |
| max-red | Shall be included and shall be set to 220 or less. |

When the channels parameter is omitted then this means that one channel is being offered.

The mode-set parameter is omitted, allowing maximum freedom for the visited network.

The mode-change-capability parameter is included and set to 2, to support potential interworking with 2G radio access (GERAN).

An example of an SDP offer for AMR-NB is shown in Table A.1.1. An example of an SDP offer for both AMR-NB and AMR-WB is shown in Table A.1.2. An example of SDP offer for AMR-NB, AMR-WB, and EVS is shown in Table A.14.1.

An SDP example for offering and accepting a dual-mono session for EVS is shown in Annex A.14.1 and A.14.3.

An MTSI client in terminal may divide the offer-answer negotiation into several phases and offer different configurations in different SDP offers. If this is done then the first SDP offer in the initial offer-answer negotiation shall include the most preferable configurations. For AMR-NB, this means that the first SDP offer in the initial offer-answer negotiation shall include at least one RTP payload type for AMR-NB with the parameters as defined in Table 6.1. If wideband speech is offered then the first SDP offer in the initial offer-answer negotiation shall include also at least one RTP payload type for AMR-WB with the parameters as defined in Table 6.1. This also means that offers for octet-aligned payload format do not need to be included in the first SDP offer. If super-wideband or fullband speech is offered, the first SDP offer in the initial offer-answer negotiation shall include at least one RTP payload type for EVS with the parameters as defined in [125]. One example of dividing the offer-answer negotiation into two phases, and the corresponding SDP offers, is shown in clause A.1.1.2.2.

NOTE 4: Dividing the offer-answer negotiation into several phases may lead to never offering the less preferred configurations, if the other end-point accepts to use at least one of the configurations offered in the initial SDP offer.

If the speech media is re-negotiated during the session then the knowledge from earlier offer-answer negotiations should be used in order to shorten the session re-negotiation time. I.e., failed offer-answer transactions shall not be repeated.

6.2.2.3 Generating SDP answer

An MTSI client in terminal must understand all the payload format options that are defined in RFC 4867 [28], and in [125]. It does not have to support operating according to all these options but must be capable to properly accepting or rejecting all options.

The SDP answer depends on many factors, for example:

- what is included in the SDP offer and in what preference order that is defined. The SDP offer will probably be different if it is generated by another MTSI client in terminal, by an MTSI MGW, a TISPAN client or some other VoIP client that does not follow this specification;
- if terminal and/or network resources are available; and:
- if there are other configurations, for example defined with OMA-DM, that mandate, recommend or prevent some configurations.

Table 6.3 describes requirements and recommendations for handling of the AMR payload format parameters and for how to generate the SDP answer.

NOTE 1: An MTSI client in terminal may support more features than what is required by this specification, e.g. crc, robust sorting and interleaving. Table 6.3 describes the handling of the AMR payload format parameters when the MTSI client implementation supports only those features that are required by this specification. Tables 6.3a-6.3c describe the handling of the EVS payload format parameters.

Table 6.3: Handling of the AMR-NB and AMR-WB SDP parameters in the received SDP offer and in the SDP answer

| Parameter in the received SDP offer | Comments | Handling |
|-------------------------------------|---|--|
| Codec | Wide-band speech is preferable over narrow-band speech | <p>If both AMR-WB and AMR-NB are offered and if AMR-WB is supported by the answering MTSI client in terminal then it shall select to use the AMR-WB codec and include this codec in the SDP answer, unless another preference order is indicated in the SDP offer. If the MTSI client in terminal only supports AMR-NB then this codec shall be selected to be used and shall be included in the SDP answer.</p> <p>The SDP answer shall only include one RTP Payload Type for speech, see NOTE 1.</p> |
| octet-align | <p>Both the bandwidth-efficient and the octet-aligned payload formats are supported by the MTSI client in terminal.</p> <p>MTSI MGWs for GERAN or UTRAN are likely to either not include the octet-align parameter or to offer octet-align=0.</p> <p>The bandwidth-efficient payload format is preferable over the octet-aligned payload format.</p> | <p>The offer shall not be rejected purely based on the offered payload format variant.</p> <p>If both bandwidth-efficient and octet-aligned are included in the received SDP offer then the MTSI client in terminal shall select the bandwidth-efficient payload format and include it in the configuration in the SDP answer.</p> |
| mode-set | <p>The MTSI client in terminal can interoperate properly with whatever mode-set the other end-point offers or if no mode-set is offered.</p> <p>The possibilities to use the higher bit rate codec modes also depend on the offered bandwidth.</p> <p>MTSI MGWs for GERAN or UTRAN inter-working are likely to include the mode-set in the offer if in case the intention is to use TFO or TrFO.</p> <p>Mode sets that give more adaptation possibilities are preferable over mode-sets with fewer or no adaptation possibilities.</p> <p>An MTSI client in terminal may be configured with a preferred mode set. Otherwise, the preferred mode-set for AMR-NB is {12.2, 7.4, 5.9, 4.75} and for AMR-WB it is {12.65, 8.85 and 6.60}.</p> | <p>The offer shall not be rejected purely based on the offered mode-set.</p> <p>If only one mode-set is offered then the MTSI client in terminal shall select to use this and include the same mode-set in the SDP answer.</p> <p>If several different payload types for the same codec with different mode-sets (possibly including one or more payload type without mode set) are included in the received SDP offer, then the MTSI client in terminal should select in the first hand the mode-set that provides the largest degrees of freedom for codec mode adaptation and in the second hand the mode-set that is closest to the preferred mode sets.</p> <p>If only a payload type without mode-set has been offered, or if an MTSI client in terminal selects a payload type without mode-set from among the offered ones, and the MTSI client in terminal intends to use only some modes (e.g. one of the preferred mode sets defined at left), then the MTSI client in terminal should include these modes as the mode-set.</p> <p>There are also dependencies between the mode-set and the SDP b=AS bandwidth parameter; see Clause 6.2.5.2.</p> |
| mode-change-period | <p>The MTSI client in terminal can interoperate properly with whatever mode-change-period the other end-point offers.</p> <p>MTSI MGWs for GERAN or UTRAN inter-working are likely to include mode-change-period=2 in the offer if in case the intention is to use TFO or TrFO.</p> | <p>The offer shall not be rejected purely based on the offered mode-change-period.</p> <p>If the received SDP offer defines mode-change-period=2 then this information shall be used to determine the mode changes for AMR-NB or AMR-WB encoded media that the MTSI client in terminal sends.</p> <p>The MTSI client in terminal should not include the mode-change-period parameter in the SDP answer since it has no corresponding limitations.</p> |

| Parameter in the received SDP offer | Comments | Handling |
|-------------------------------------|--|---|
| mode-change-capability | The MTSI client in terminal can interoperate with whatever capabilities the other end-point declares. | <p>The offer shall not be rejected purely based on the offered mode-change-capability.</p> <p>The mode-change-capability information should be used to determine a proper value, or prevent using an improper value, for mode-change-period in the SDP answer, see above. If the offer includes mode-change-capability=1, then the MTSI client in terminal shall not offer mode-change-period=2 in the answer.</p> <p>The MTSI client in terminal shall include mode-change-capability=2 in the SDP answer since it is required to support restricting mode changes to every other frame.</p> |
| mode-change-neighbor | The MTSI client in terminal can interoperate with whatever limitations the other end-point offers. | <p>The offer shall not be rejected purely based on the offered mode-change-neighbor.</p> <p>The MTSI client in terminal shall use this information to determine how mode changes can be performed for AMR-NB or AMR-WB encoded media that the MTSI client in terminal sends.</p> <p>The MTSI client in terminal shall not include the mode-change-neighbor parameter in the SDP answer since it has no corresponding limitations.</p> |
| maxptime | <p>The MTSI client in terminal can interoperate with whatever value that is offered.</p> <p>The MTSI client in terminal may also use this information to determine a suitable value for max-red in the SDP answer.</p> | <p>The offer shall not be rejected purely based on the offered maxptime.</p> <p>The MTSI client in terminal shall use this information to control the packetization when sending RTP packets to the other end-point, see also clause 7.4.2.</p> <p>The maxptime parameter shall be included in the SDP answer and shall be an integer multiple of 20.</p> <p>If the received SDP offer includes both the max-red and ptime parameter then the MTSI client in terminal may choose to use this information to define a suitable value for maxptime in the SDP answer, see NOTE 2. The MTSI client in terminal may also choose to set the maxptime value to 240, regardless of the ptime and/or max-red parameters in the SDP offer.</p> <p>The maxptime value in the SDP answer shall not be smaller than ptime value in the SDP answer. The maxptime value should be selected to give at least some room for adaptation.</p> |
| crc | The MTSI client in terminal is not required to support this option. | The MTSI client in terminal may have to reject offered RTP payload types including this option. |
| robust-sorting | The MTSI client in terminal is not required to support this option. | The MTSI client in terminal may have to reject offered RTP payload types including this option. |
| interleaving | The MTSI client in terminal is not required to support this option. | The MTSI client in terminal may have to reject offered RTP payload types including this option. |

| Parameter in the received SDP offer | Comments | Handling |
|---|---|--|
| ptime | <p>The MTSI client in terminal can interoperate with whatever value that is offered.</p> | <p>The offer shall not be rejected purely based on the offered ptime.</p> <p>The MTSI client in terminal should use this information and should use the requested packetization when sending RTP packets to the other end-point. The MTSI client should use the ptime value to determine how many non-redundant speech frames that can be packed into the RTP packets. The requirements in clause 7.4.2 shall be followed even if ptime in the SDP offer is larger than 80.</p> <p>The ptime parameter shall be included in the SDP answer and shall be an integer multiple of 20.</p> <p>If the received SDP offer includes the ptime parameters then the MTSI client in terminal may choose to use this information to define a suitable value for ptime in the SDP answer, see NOTE 3. The MTSI client in terminal may also choose to set the ptime value in the SDP answer according to Table 7.1, regardless of the ptime parameter in the SDP offer.</p> <p>The ptime value in the SDP answer shall not be larger than the maxptime value in the SDP answer.</p> |
| channels | <p>The number of channels may either be explicitly indicated in the SDP by including '/1', '/2', etc. on the a=rtmpmap line, but the number of channels may also be omitted. When the number of channels is omitted then the default rule is that one channel is being offered.</p> <p>The MTSI client in terminal is only required to support audio media using one channel. Offered RTP payload types with more than one channel may therefore have to be rejected.</p> | <p>When the MTSI client in terminal accepts an offer for single-channel audio then the SDP answer shall either explicitly indicate '/1' or omit the channels parameter.</p> <p>When the MTSI client in terminal accepts an offer for multi-channel audio then the number of channels shall be included in the SDP answer.</p> |
| max-red | <p>The MTSI client in terminal may use this information to bound the delay for receiving redundant frames.</p> <p>The MTSI client in terminal may also use this information to determine a suitable value for maxptime in the SDP answer.</p> | <p>The max-red parameter shall be included in the SDP answer and shall be an integer multiple of 20.</p> <p>If the received SDP offer includes both the ptime and maxptime parameters then the MTSI client in terminal may choose to use this information to define a suitable value for max-red in the SDP answer, see NOTE 2. The MTSI client in terminal may also choose to set the max-red value to 220.</p> <p>The max-red value in the SDP answer should be selected to give at least some room for adaptation.</p> |
| ecn-capable-rtp: leap ect=0 | <p>An MTSI client in terminal uses this SDP attribute to offer ECN for RTP-transported media</p> | <p>Shall be included in the SDP answer if accepting an offer to use ECN and if the session setup allows for bit-rate adaptation</p> |
| <p>NOTE 1: An MTSI client may include both a speech coded, e.g. AMR-NB or AMR-WB, and 'telephone-events' for DTMF in the SDP answer, see TS 24.229 Clause 6.1, [7].</p> <p>NOTE 2: It is possible to use the following relationship between maxptime, ptime and max-red: $\text{maxptime} = \text{ptime} + \text{max-red}.$ There is however no mandatory requirement that these parameters must be aligned in this way.</p> <p>NOTE 3: It may be wise to use the same ptime value in the SDP answer as was given in the SDP offer, especially if the ptime in the SDP offer is larger than 20, since a value larger than the frame length indicates that the other end-point is somehow packet rate limited.</p> | | |

If an SDP offer is received from another MTSI client in terminal using the AMR-NB or AMR-WB codec, then the SDP offer will include configurations as described in Table 6.1 and Table 6.2. If the MTSI client in terminal chooses to accept the offer for using the AMR-NB or AMR-WB codec, as configured in Table 6.1 or Table 6.2 then the MTSI client in terminal shall support a configuration where the MTSI client in terminal creates an SDP answer containing an RTP payload type for the AMR-NB and AMR-WB codec as shown in Table 6.4.

Table 6.3a: Handling of SDP parameters common to EVS Primary and EVS AMR-WB IO in the received SDP offer and in the SDP answer

| Parameter | Comments | Handling |
|-----------------|---|--|
| ptime | | |
| maxptime | | |
| dtx | | MTSI client in terminal shall not include dtx in the initial SDP offer. MTSI MGW may modify SDP offer to include dtx in order to disable DTX in the session. |
| dtx-recv | | MTSI client in terminal shall not include dtx-recv. MTSI MGW may modify SDP offer or answer in order to disable DTX for the send direction of the receiver of dtx-recv. |
| hf-only | | - |
| evs-mode-switch | This parameter is used by MTSI MGW either when starting in EVS AMR-WB IO mode instead of EVS Primary mode or when switching between EVS Primary mode and EVS AMR-WB IO mode, e.g., for SRVCC. | MTSI client in terminal shall not include evs-mode-switch in the initial SDP offer. When including evs-mode-switch in the SDP offer during a session, the offerer shall use the requested mode when sending EVS packets. However, if a media stream is already being received, the offerer needs to be prepared to receive packets in both EVS primary and EVS AMR-WB IO modes until receiving the answer. When including evs-mode-switch in the SDP answer during a session, the answerer shall use the requested mode when sending EVS packets. When receiving SDP answer including evs-mode-switch during a session, the offerer shall use the requested mode when sending EVS packets. |
| max-red | See Table 6.3 | |
| channels | See Table 6.3 | |

Table 6.3b: Handling of the EVS Primary SDP parameters in the received SDP offer and in the SDP answer

| Parameter | Comments | Handling |
|------------|--|---|
| br | | An MTSI client in terminal supporting the EVS codec is required to support the entire bit-rate range but may offer a smaller bit-rate range or even a single bit-rate. |
| br-send | | |
| br-recv | | |
| bw | The session should start with the maximum bandwidth supported by the initial bit-rate up to the maximum negotiated bandwidth. If a range of bandwidth is negotiated, the codec can operate in any bandwidth in the session but the maximum bandwidth in the range should be used after the start of or update of the session. If a single audio bandwidth higher than narrowband is negotiated, the codec operates in the negotiated bandwidth but can use lower bandwidth(s) in the session, depending on the input signal. | Both the offerer and the answerer shall send according to the bandwidth parameter in the answer. |
| bw-send | | |
| bw-recv | | |
| ch-send | | |
| ch-recv | | |
| cmr | In EVS AMR-WB IO mode, CMR to the bit-rates of EVS AMR-WB IO mode and NO_REQ is always enabled. | If cmr=-1 and the session is in the EVS Primary mode, MTSI client in terminal shall not transmit CMR. If cmr=-1 and the session is in the EVS AMR-WB IO, MTSI client in terminal shall restrict CMR to values of EVS AMR-WB-IO bit-rates and NO_REQ in the session. MTSI client in terminal is required to accept CMR even when cmr=-1. MTSI client in terminal is required to accept RTP payload without CMR even when cmr=1. |
| ch-aw-recv | | If a positive (2, 3, 5, or 7) value of ch-aw-recv is declared for a payload type and the payload type is accepted, the receiver of the parameter shall send partial redundancy (channel-aware mode) at the start of the session using the value as the offset. If ch-aw-recv=0 is declared or not present for a payload type and the payload type is accepted, the receiver of the parameter shall not send partial redundancy (channel-aware mode) at the start of the session. If ch-aw-recv=-1 is declared for a payload type and the payload type is accepted, the receiver of the parameter shall not send partial redundancy (channel-aware mode) in the session. If not present or a non-negative (0, 2, 3, 5, or 7) value of ch-aw-recv is declared for a payload type and the payload type is accepted, partial redundancy (channel-aware mode) can be activated or deactivated during the session based on the expected or estimated channel condition through adaptation signaling, such as CMR (see Annex A.2 of [125]) or RTCP based signalling (see clause 10.2). If not present or a non-negative (0, 2, 3, 5, or 7) value of ch-aw-recv is declared for a payload type and the payload type is accepted, the partial redundancy offset value can also be adjusted during the session based on the expected or estimated channel condition through adaptation signaling. |

Table 6.3c: SDP parameters for the EVS AMR-WB IO parameters in the received SDP offer and in the SDP answer

| Parameter | Comments | Handling |
|------------------------|--|---|
| mode-set | See Table 6.3 | |
| mode-change-period | | |
| mode-change-neighbor | | |
| mode-change-capability | The default value is re-defined in comparison to that in [28]. | As the default and the only allowed value of mode-change-capability is 2 in EVS AMR-WB IO, it is not required to include this parameter in the SDP offer or answer. |

NOTE 2: ECN-triggered adaptation is currently undefined for EVS. This does not prevent ECN-triggered adaptation from being negotiated and used for AMR or AMR-WB.

Table 6.4: SDP parameters for AMR-NB or AMR-WB for SDP answer when the SDP offer is received from another MTSI client in terminal

| Parameter | Usage |
|-----------------------------|--|
| octet-align | Shall not be included |
| mode-set | See Table 6.3 |
| mode-change-period | Shall not be included |
| mode-change-capability | May be included. If it is included then it shall be set to 2 |
| mode-change-neighbor | Shall not be included |
| maxptime | Shall be set to 240, see also Table 7.1 |
| crc | Shall not be included |
| robust-sorting | Shall not be included |
| interleaving | Shall not be included |
| ptime | Shall be set according to Table 7.1 |
| channels | Shall either be set to 1 or be omitted |
| max-red | Shall be included and shall be set to 220 or less |
| ecn-capable-rtp: leap ect=0 | Shall be included in the SDP answer if accepting an offer to use ECN and if the session setup allows for bit-rate adaptation |

If an SDP offer is received from a MTSI MGW inter-working with CS GERAN/UTRAN, and when the MTSI MGW supports ECN (see also clause 12.3.3), then it is likely to be configured as shown in Table 6.5 if the MTSI MGW does not support redundancy.

Table 6.5: Expected configuration of SDP parameters for AMR-NB or AMR-WB in an SDP offer from an MTSI MGW inter-working with CS GERAN/UTRAN

| Parameter | Usage |
|-----------------------------|--|
| octet-align | Either not included or set to 0 |
| mode-set | Included and indicates the codec modes that are allowed in the CS network |
| mode-change-period | Set to 2 |
| mode-change-capability | Set to 2 |
| mode-change-neighbor | Set to 1 if the CS network is GERAN |
| maxptime | Set to 80, see also Table 12.1 |
| crc | Not included |
| robust-sorting | Not included |
| interleaving | Not included |
| ptime | Set according to Table 12.1 |
| channels | Set to 1 or parameter is omitted |
| max-red | Set to 0 |
| ecn-capable-rtp: leap ect=0 | Shall be included in the SDP answer if accepting an offer to use ECN and if the session setup allows for bit-rate adaptation |

If the MTSI client in terminal accepts the offer included in Table 6.5 then the MTSI client in terminal shall support a configuration where the MTSI client in terminal creates an SDP answer containing an RTP payload type for the AMR-NB and AMR-WB codecs as shown in Table 6.6.

Table 6.6: SDP parameters for AMR-NB or AMR-WB for SDP answer when the SDP offer is received from another MTSI MGW

| Parameter | Usage |
|-----------------------------|--|
| octet-align | Shall be set according to the offer |
| mode-set | See Table 6.3 |
| mode-change-period | Shall not be included |
| mode-change-capability | May be included. If it is included then it shall be set to 2 |
| mode-change-neighbor | Shall not be included |
| maxptime | Shall be set to 240, see also Table 7.1 |
| crc | Shall not be included |
| robust-sorting | Shall not be included |
| interleaving | Shall not be included |
| ptime | Shall be set according to Table 7.1 |
| channels | Shall be set according to the offer |
| max-red | Shall be included and shall be set to 220 or less |
| ecn-capable-rtp: leap ect=0 | Shall be included in the SDP answer if accepting an offer to use ECN and if the session setup allows for bit-rate adaptation |

6.2.3 Video

6.2.3.1 Introduction

The common session setup procedures for video are described in clause 6.2.3.2. Session setup procedures for Coordination of Video Orientation (CVO) and Video Region of Interest (ROI) are described in clauses 6.2.3.3 and 6.2.3.4, respectively. Session setup procedures for RTP Retransmission and Forward Error Correction (FEC) are described in clauses 6.2.3.5 and 6.2.3.6, respectively.

6.2.3.2 Common Session Setup Procedures

If video is used in a session, the session setup shall determine the applicable bandwidth(s) as defined in clause 6.2.5, RTP profile, video codec, profile and level. The "imageattr" attribute as specified in IETF RFC 6236 [76] should be supported. The "framesize" attribute as specified in [60] shall not be used in the session setup.

An MTSI client shall offer AVPF for all media streams containing video. RTP profile negotiation shall be done as described in clause 6.2.1a.

An MTSI client is required to support the AVPF feedback messages trr-int, NACK and PLI [40] and the CCM feedback messages FIR, TMMBR and TMMBN [43], see Clauses 7.3.3 and 10.3. These feedback messages can only be used together with AVPF and shall be negotiated in SDP offer/answer before they can be used in the session [40]. An MTSI client sending an SDP offer for AVPF shall also include these AVPF and CCM feedback messages in the offer. An MTSI client accepting an SDP offer for AVPF for video shall also accept these AVPF and CCM feedback messages if they are offered.

If an MTSI client offers to use ECN for video in RTP streams then the MTSI client shall offer ECN Capable Transport as defined below. If an MTSI client accepts an offer for ECN for video then the MTSI client shall declare ECN Capable Transport in the SDP answer as defined below. The SDP negotiation of ECN Capable Transport is described in [84].

The use of ECN for a video stream in RTP is negotiated with the "ecn-capable-rtp" SDP attribute, [84]. ECN is enabled when both clients agree to use ECN as configured below. An MTSI client using ECN shall therefore also include the following parameters and parameter values for the ECN attribute:

- 'leap', to indicate that the leap-of-faith initiation method shall be used;
- 'ect=0', to indicate that ECT(0) shall be set for every packet.

An MTSI client offering ECN for video shall indicate support of TMMBR [43] by including the "ccm tmmbr" value within an "rtcp-fb" SDP attribute [40]. An MTSI client offering ECN for video may indicate support for RTCP AVPF ECN feedback messages [84] using the "rtcp-fb" SDP attribute with the "nack" feedback parameter and the "ecn" feedback parameter value. An MTSI client offering ECN for video may indicate support for RTCP XR ECN summary reports [84] using the "rtcp-xr" SDP attribute and the "ecn-sum" parameter.

An MTSI client receiving an offer for ECN for video with an indication of support of TMMBR [43] within an "rtcp-fb" attribute should accept the offer if it supports ECN. It shall then indicate support for TMMBR using an "rtcp-fb" attribute in the SDP answer.

An MTSI client receiving an offer for ECN for video with an indication of support of RTCP AVPF ECN feedback message but without support for TMMBR should accept the offer if it supports ECN and also the RTCP AVPF ECN feedback message. It shall then indicate support of the RTCP AVPF ECN feedback message using the "rtcp-fb" attribute in the SDP answer.

An MTSI client receiving an offer for ECN for video with an indication of support of RTCP XR ECN summary reports [84] without support for TMMBR should accept the offer if it supports ECN and also the RTCP XR ECN summary reports. It shall then indicate support of RTCP XR ECN summary reports in the SDP answer.

The use of ECN is disabled when a client sends an SDP without the "ecn-capable-rtp" SDP attribute.

An MTSI client may initiate a session re-negotiation to disable ECN to resolve ECN-related error cases. An ECN-related error case may be, for example, detecting non-ECT in the received packets when ECT(0) was expected or detecting a very high packet loss rate when ECN is used.

Examples of SDP offers and answers for video can be found in clause A.4. SDP examples for offering and accepting ECT are shown in Annex A.12.2.

NOTE: For H.264 / MPEG-4 (Part 10) AVC, the optional max-rcmd-nalu-size receiver-capability parameter of RFC 6184 [25] should be set to the smaller of the MTU size (if known) minus header size or 1 400 bytes (otherwise).

The "framerate" attribute as specified in [8] indicates the maximum frame rate the offerer wishes to receive. If the "framerate" attribute is present in the SDP offer, its value may be modified in the SDP answer when the answerer wishes to receive video with a different maximum frame rate than what was indicated in the offer.

An MTSI client in terminal setting up asymmetric video streams with H.264 (AVC) should use both the 'level-asymmetry-allowed' parameter and the 'max-recv-level' parameter that are defined in the H.264 payload format, [25]. When the 'max-recv-level' parameter is used then the level offered for the receiving direction using the 'max-recv-level' parameter must be higher than the default level that is offered with the 'profile-level-id' parameter.

An SDP offer-answer example showing the usage of the 'level-asymmetry-allowed' and 'max-recv-level' parameters is included in Annex A.4.5.

An MTSI client in terminal setting up asymmetric video streams with H.265 (HEVC) should use the 'max-recv-level-id' parameter that is defined in the H.265 payload format, [120]. The level offered for the receiving direction using the 'max-recv-level-id' parameter must be higher than the default level that is offered with the 'level-id' parameter.

An SDP offer-answer example showing the usage of the 'max-recv-level-id' parameter is included in Annex A.4.8.

The resolutions in the "imageattr" attribute correspond to the image size information in the encoded video bitstream such that the x-component corresponds to the image width, and the y-component corresponds to the height component. When the bit-rate is being adapted, values of image width or image height smaller than the x- or y-component(s) in the negotiated "imageattr" attribute may be temporarily used.

MTSI clients should indicate all their preferred resolutions in the SDP offer and answer exchanges using the "imageattr" attribute. MTSI clients should not renegotiate SDP in case of no agreement on resolution, i.e., no new SDP offer-answer exchanges are expected in case of a mismatch of resolutions as indicated by the "imageattr" attribute. This is an MTSI-specific relaxation of the requirement in IETF RFC 6236 [76], according to which SDP renegotiation is expected in case of no agreed resolution. Related SDP offer-answer examples based on this expected behavior for MTSI clients can be found in Annex A.4.4a.

6.2.3.3 Coordination of Video Orientation (CVO)

An MTSI client should support Coordination of Video Orientation (CVO) as specified in clause 7.4.5.

An MTSI client supporting CVO shall offer Coordination of Video Orientation (CVO) in SDP for all media streams containing video. CVO is offered by including the a=extmap attribute [95] indicating the CVO URN under the relevant media line scope. The CVO URN is: urn:3gpp:video-orientation. Here is an example usage of this URN to signal CVO relative to a media line:

```
a=extmap:7 urn:3gpp:video-orientation
```

The number 7 in the example may be replaced with any number in the range 1-14. The above SDP line indicates 2 bits of granularity for rotation and shall be present when offering CVO.

Higher granularity CVO supports up to 6 bits of precision and may additionally be offered for the rotation value by also including the following line of SDP in the offer:

```
a=extmap:5 urn:3gpp:video-orientation:6
```

For terminals with asymmetric capability (e.g. the ability to process video orientation information but not detect orientation), the sendonly and recvonly attributes [95] may be used. Terminals should express their capability in each direction sufficiently clearly such that signals are only sent in each direction to the extent that they both express useful information and can be processed by the recipient; for example, 6-bit signals would not be sent when the sending terminal can only detect orientation to a precision of 2 bits, and terminals incapable of detecting orientation would not send the header.

An MTSI client supporting CVO shall respond to receive CVO when CVO is offered to be sent in SDP, by including exactly one of the offered extmap attributes. An MTSI client supporting CVO shall respond to send CVO when CVO is offered to be received in SDP, by including exactly one of the offered extmap attributes. An MTSI client shall not answer with CVO in a direction when not offered CVO in that direction in SDP.

6.2.3.4 Video Region-of-Interest (ROI)

An MTSI client should support Video Region-of-Interest (ROI) signaling as specified in clause 7.3.7.

An MTSI client supporting ROI shall support at least one of the following modes to request a desired region of interest (signalled from an MTSI receiver to an MTSI sender):

- Far End Camera Control' (FECC), as specified in [135]-[139] and clause 7.3.7

- 'Arbitrary ROI', as specified in clause 7.3.7
- 'Pre-defined ROI', as specified in clause 7.3.7

An MTSI client supporting FECC using H.224 shall offer FECC in SDP for all media streams containing video, where FECC capabilities are desired. FECC shall be offered via the syntax and semantics defined in IETF RFC 4573 [139]. The MIME type 'application/h224' corresponding to the RTP payload format for H.224 shall be used as in [139], which also defines the SDP parameters needed to indicate support for FECC using H.224.

An MTSI client supporting 'Arbitrary ROI' mode shall offer 'Arbitrary ROI' in SDP for all media streams containing video, where 'Arbitrary ROI' capabilities are desired. 'Arbitrary ROI' shall be offered by including the a=rtcp-fb attribute [40] with the 'Arbitrary ROI' type under the relevant media line scope. The 'Arbitrary ROI' type in conjunction with the RTCP feedback method shall be expressed with the following parameter: 3gpp-roi-arbitrary. A wildcard payload type ("*") may be used to indicate that the RTCP feedback attribute for 'Arbitrary ROI' signaling applies to all payload types. If several types of ROI signaling are supported and/or the same 'Arbitrary ROI' shall be specified for a subset of the payload types, several "a=rtcp-fb" lines can be used. Here is an example usage of this attribute to signal 'Arbitrary ROI' relative to a media line based on the RTCP feedback method:

```
a=rtcp-fb:* 3gpp-roi-arbitrary
```

An MTSI client supporting 'Pre-defined ROI' mode shall offer 'Pre-defined ROI' in SDP for all media streams containing video, where 'Pre-defined ROI' capabilities are desired. 'Pre-defined ROI' shall be offered by including the a=rtcp-fb attribute [40] with the 'Pre-defined ROI' type under the relevant media line scope. The 'Pre-defined ROI' type in conjunction with the RTCP feedback method shall be expressed with the following parameter: 3gpp-roi-predefined. A wildcard payload type ("*") may be used to indicate that the RTCP feedback attribute for 'Pre-defined ROI' signaling applies to all payload types. If several types of ROI signaling are supported and/or the same 'Pre-defined ROI' shall be specified for a subset of the payload types, several "a=rtcp-fb" lines can be used. Here is an example usage of this attribute to signal 'Pre-defined ROI' relative to a media line based on the RTCP feedback method:

```
a=rtcp-fb:* 3gpp-roi-predefined
```

The IANA registration information on the new RTCP feedback types for 'Arbitrary ROI' and 'Pre-defined ROI' are provided in Annex R.1.

The ABNF for rtcp-fb-val corresponding to the feedback types "3gpp-roi-arbitrary" and "3gpp-roi-predefined" is given as follows:

```
rtcp-fb-val =/ "3gpp-roi-arbitrary"
```

```
rtcp-fb-val =/ "3gpp-roi-predefined"
```

An MTSI sender supporting the 'Pre-defined ROI' feature shall offer detailed pre-defined ROI information in the initial offer-answer negotiation by carrying it in SDP. Pre-defined ROIs shall be offered by including the "a=predefined_ROI" attribute under the relevant media line. The following parameters shall be provided in the attribute for each pre-defined ROI:

- ROI_ID – identifies the pre-defined ROI
- Position_X - specifies the x-coordinate for the upper left corner of the ROI area covered in the original content (i.e., uncompressed captured content) in units of pixels
- Position_Y - specifies the y-coordinate for the upper left corner of the ROI area covered in the original content in units of pixels
- Size_X - specifies the horizontal size of the ROI area covered in the original content in units of pixels
- Size_Y - specifies the vertical size of the ROI area covered in the original content in units of pixels
- Name- specifies the name of the pre-defined ROI.

The syntax for the "a=predefined_ROI" attribute shall conform to the following ABNF:

```
predefined_ROI = "predefined_ROI:" PT 1*WSP attr-list
```

```
PT = 1*DIGIT / "*"
```

```
attr-list = ( set *(1*WSP set) ) / "*"
```

; WSP and DIGIT defined in [RFC5234]

```
set= "[" ROI_ID=" idvalue "," Position_X=" posvalue "," Position_Y=" posvalue "," Size_X=" sizevalue ","
"Size_Y=" sizevalue "," Name=" namevalue "]"
```

idvalue= onetone*2DIGIT

; Digit between 1 and 9 that is

; followed by 0 to 2 other digits

posvalue = sizevalue / "0"

; position may be "0"

sizevalue = onetone *5DIGIT

; Digit between 1 and 9 that is

; followed by 0 to 5 other digits

onetone = "1" / "2" / "3" / "4" / "5" / "6" / "7" / "8" / "9"

; Digit between 1 and 9

namevalue = byte-string

; byte-string defined in RFC 4566

The SDP offer with a=predefined_ROI parameter shall contain the full-size view of the video indicated via ROI_ID=0.

Here is an example use of the "a=predefined_ROI" attribute relative to a media line:

```
a=predefined_ROI:99
```

```
[ROI_ID=0,Position_X=1,Position_Y=1,Size_X=1080,Size_Y=720,Name=fullview]
[ROI_ID=1,Position_X=1,Position_Y=1,Size_X=540,Size_Y=360,Name=museum]
[ROI_ID=2,Position_X=541,Position_Y=1,Size_X=540,Size_Y=360,Name=cinema]
[ROI_ID=3,Position_X=1,Position_Y=361,Size_X=540,Size_Y=360,Name=park]
[ROI_ID=4,Position_X=541,Position_Y=361,Size_X=540,Size_Y=360,Name= zoo]
```

The IANA registration information for the "a=predefined_ROI" SDP attribute is provided in Annex M.5.

In response to the SDP offer with the set of offered pre-defined ROIs provided using the "a=predefined_ROI" line(s), an MTSI client accepting 'Pre-defined ROI' shall provide an SDP answer using the "a=predefined_ROI" line(s) containing the accepted set of pre-defined ROIs. Such an SDP answer shall also contain the "a=rtcp-fb:* 3gpp-roi-predefined" line. The accepted set of pre-defined ROIs shall be a subset of the offered set of pre-defined ROIs. If the SDP answer contains the a=rtcp-fb:* 3gpp-roi-predefined" line, but does not contain a "a=predefined_ROI" line, this indicates that the MTSI client supports the 'Pre-defined ROI' mode, but none of the ROIs in the offered set of pre-defined ROIs is acceptable for this MTSI client. Following the successful negotiation of 'Pre-defined ROI', the MTSI receiver uses the RTCP feedback method to request from the accepted set of pre-defined ROIs and MTSI sender encodes the sent video accordingly to provide the requested pre-defined ROI.

If the SDP offer just provides the "a=predefined_ROI" but not "a=rtcp-fb:* 3gpp-roi-predefined", then the "a=predefined_ROI" lines should be ignored.

A new SDP offer-answer negotiation can be performed to modify the set of pre-defined ROIs. The MTSI sender may update all the content of pre-defined ROIs, including the total number of pre-defined ROIs, and the position, size and name of each of the pre-defined ROIs.

The ROI information parameters exchanged via the a=predefined_ROI parameter in the SDP signalling defined above are independent of the negotiated video resolution for the encoded content. Instead, the ROI information parameters defined above take as reference the original video content, i.e., uncompressed captured video content. Therefore, no modifications or remappings of ROI parameters are necessary during any transcoding that results in changes in video resolution or during potential dynamic adaptations of encoded video resolution at the sender.

An MTSI client supporting 'Arbitrary ROI' or 'Pre-defined ROI' should also offer 'Sent ROI' in SDP for all media streams containing video. An MTSI sender accepting "Arbitrary ROI" or 'Pre-defined ROI' shall also accept an

accompanying 'Sent ROI' offer. An MTSI sender accepting 'FECC' should also accept an accompanying 'Sent ROI' offer. 'Sent ROI' is specified in clause 7.3.7 and is offered by including the `a=extmap` attribute [95] indicating the 'Sent ROI' URN under the relevant media line scope. The 'Sent ROI' URN corresponding to an arbitrary ROI is: `urn:3gpp:roi-sent`, on which the IANA registration information is provided in Annex O.4. The 'Sent ROI' URN corresponding to a pre-defined ROI is: `urn:3gpp:predefined-roi-sent`, on which the IANA registration information is provided in Annex O.5. Here is an example usage of this URN to signal 'Sent ROI' relative to a media line:

```
a=extmap:7 urn:3gpp:roi-sent
```

The number 7 in the example may be replaced with any number in the range 1-14.

6.2.3.5 RTP Retransmission

An MTSI client should support RTP Retransmission as specified in clause 7.4.6.

An MTSI client supporting RTP Retransmission shall offer retransmission for all media streams containing video. The binding used for retransmission stream to the payload type number is indicated by an `rtptime` attribute. The MIME subtype name used in the binding is "rtx". The "apt" (associated payload type) parameter shall be used to map the retransmission payload type to the associated original payload type. The "rtx-time" payload-format-specific parameter indicates the maximum time a sender will keep an original RTP packet in its buffers available for retransmission [140]. An MTSI client offering RTP retransmission shall specify "rtx-time" parameter.

An SDP offer/answer example showing the usage of the "apt" and "rtx-time" is included in Annex A.4.2c.

Bandwidth allocation for video with RTP Retransmission is discussed in clause 6.2.5.3.

6.2.3.6 Forward Error Correction (FEC)

An MTSI client should support Forward Error Correction (FEC) as specified in clause 7.4.7.

An MTSI client supporting FEC shall offer FEC for all media streams containing video. The MIME subtype name used for FEC stream is "flexfec". The "ssrc-group" attribute is used to designate FEC grouping association according to "ssrc" identifiers along with the "FEC-FR" grouping semantics for FEC Framework. The "repair-window" parameter indicates the time span of the source and repair packets [141] [143]. An example for FEC grouping relative to a media line is:

```
a=ssrc:1234
a=ssrc:2345
a=ssrc-group:FEC-FR 1234 2345
```

An SDP offer/answer example showing the usage of the "flexfec", "repair-window" and "ssrc-group" is included in Annex A.4.2d.

Bandwidth allocation for video with FEC is discussed in clause 6.2.5.3.

6.2.4 Text

An MTSI client should offer AVP for all media streams containing text. Only in cases where there is an explicit demand for the AVPF RTCP reporting timing or feedback messages AVPF shall be used. If AVPF is offered then RTP profile negotiation shall be done as described in clause 6.2.1a.

Examples of SDP offers for text can be found in clause A.5.

An MTSI client configured to automatically enable global text telephony (GTT), e.g. because the MTSI client is used by a deaf or hearing-impaired person or a person wanting to communicate with such an impaired person, shall accept an initial INVITE request for a SIP dialogue if the SDP offer does not include real time text media. It shall then send a new SDP offer (e.g. in a SIP UPDATE request during call establishment) adding text media for real time text conversation.

NOTE: As one example, incoming calls from a PSTN interworked by an MGCF will not contain media for real time text conversation in the initial SDP offer. The new offer adding media for real time text conversation enables the transport of real time text towards the MTSI client.

6.2.5 Bandwidth negotiation

6.2.5.1 General

The SDP shall include bandwidth information for each media stream and also for the session in total. The bandwidth information for each media stream and for the session is defined by the Application Specific (AS) bandwidth modifier as defined in RFC 4566 [8].

An MTSI client in terminal should include the 'a=bw-info' attribute in the SDP offer. When accepting a media type where the 'a=bw-info' attribute is included the MTSI client in terminal shall include the 'a=bw-info' attribute in the SDP answer if it supports the attribute. The 'a=bw-info' attribute and the below used bandwidth properties are defined in clause 19.

When the 'a=bw-info' attribute is supported, the following bandwidth properties shall be included for each RTP payload type in the SDP:

- Maximum Supported Bandwidth for sending direction.
- Maximum Desired Bandwidth for sending direction.
- Minimum Desired Bandwidth for sending direction.
- Minimum Supported Bandwidth for sending direction.
- Maximum Supported Bandwidth for receiving direction with the following exception:
 - The b=AS bandwidth modifier indicates the bandwidth needed for the RTP payload type that requires the highest bandwidth. The Maximum Supported Bandwidth for this RTP payload type is therefore indicated with the b=AS bandwidth modifier and does not need to be indicated with the 'a=bw-info' attribute for this RTP payload type. It is still allowed to include the 'a=bw-info' attribute for this RTP payload type but the value shall then be aligned with the b=AS value when sending the SDP. When receiving the SDP, the b=AS bandwidth modifier and the Maximum Supported Bandwidth for the receiving direction may not be aligned. In this case, the maximum sending rate is determined as defined below.
- Maximum Desired Bandwidth for receiving direction.
- Minimum Desired Bandwidth for receiving direction.
- Minimum Supported Bandwidth for receiving direction.

Recommended bandwidths for several codec configurations are provided in the media-specific sections.

For a media stream that has been removed by either the offerer or answerer, the inclusion of bandwidth information is optional. This is in accordance with clause 8.2 of RFC 3264 [58].

SDP examples incorporating bandwidth modifiers are shown in annex A. SDP examples using the 'a=bw-info' attribute are shown in annex A.6.3.

When an MTSI client in terminal receives an SDP offer or answer it shall determine the maximum sending rate for the selected codec by selecting the smallest of the following:

- the bandwidth value, if the b=AS parameter was included in the received SDP offer or answer
- the Maximum Supported Bandwidth for the receiving direction, if included in the received SDP
- the preconfigured data rate for the selected codec, if the MTSI client has been preconfigured by the operator to use a particular data rate for the selected codec
- the maximum data rate for the selected codec as determined by examining the codec information (e.g., codec, mode, profile, level) and any other media information (e.g.,ptime and maxptime) included in the received SDP offer or answer. This maximum data rate is determined assuming no extra bandwidth is allowed for redundancy.

The maximum sending rate may be further updated by the MTSI client in terminal based on receiving an indication of the granted QoS (see clause 6.2.7).

The MTSI client in terminal shall not transmit at a rate above the maximum sending rate. For speech, the MTSI client should transmit using the codec mode with the highest data rate allowed by the maximum sending rate, except if limited to a lower codec mode by the initial codec mode procedures (see clause 7.5.2.1.6) or by the adaptation procedures (see clause 10.2).

The MTSI client in terminal should support access network bitrate recommendation (ANBR, see clause 10.7). SDP offer/answer re-negotiation shall not be used as a replacement for dynamic media bitrate adaptation. ANBR contains information on short-term bandwidth and SDP offer/answer re-negotiations should be avoided or minimized since they consume network resources. Therefore, SDP offer/answer re-negotiation (e.g. in SIP UPDATE) shall not be initiated based on ANBR information other than in the following cases:

If;

1. The received ANBR from the access network is below the established GBR; and
2. The received ANBR cannot be supported by any of the negotiated codec configurations; and
3. Potentially increased loss and/or delay due to not lowering the bitrate are not acceptable; and
4. The MTSI client in terminal supports one or more codec configurations that supports the received ANBR; and
5. ANBR messages with values meeting all conditions in 1-4 above are received consistently for an extensive period of time (e.g. 5 seconds or more, see also clause 10.7.2)

then the MTSI client in terminal:

- may re-negotiate the session
 - To switch to a codec or codec configuration that can support the lower bitrate in the ANBR (if any); and/or
 - To reduce the number of used RTP streams (e.g. turning off the affected media); and
- If the session re-negotiation fails, shall not initiate further re-negotiation based on ANBR for that bearer in the session.

For video, if:

- TMMBR/TMMBN are not supported in the session; and
- For an extensive period of time (e.g. 5 seconds), the MTSI client in terminal consistently receives ANBR messages with values significantly below the video bitrate sent (as estimated by the receiving MTSI client in terminal) from the remote peer

Then the MTSI client in terminal may re-negotiate the session:

- To set the session bitrate for video (see clause 6.2.5) to a value corresponding to the minimum of the received ANBR and GBR (if > 0); or
- To turn video off

NOTE 1: An ANBR below GBR does not change the GBR semantics in any way. GBR is defined as a guarantee that for a packet stream not exceeding the GBR, 98 percent of the packets do not experience a delay exceeding the QCI's Packet Delay Budget (see clause 6.1.7.2 of TS 23.303 [90]). Temporarily reducing bitrate below GBR to comply with an ANBR can increase the probability that loss and/or delay can be kept within the bounds set by the used QCI.

NOTE 2: For GBR=MBR bearers, an ANBR below the GBR can frequently be supported by the negotiated codec configuration.

NOTE 3: For GBR<MBR bearers, an ANBR below the GBR can typically not be supported by the negotiated codec configuration.

NOTE 4: If the above conditions are not met, the MTSI client in terminal will ignore ANBR values below the GBR (see also clause 10.7.2).

6.2.5.2 Speech

If an MTSI client includes an AMR or AMR-WB mode-set, or EVS Primary mode br or br-recv parameter in the SDP offer or answer, the MTSI client shall set the b=AS parameter to a value matching the maximum codec mode in the mode-set or the highest bit-rate in the br or br-recv, the packetization time (ptime), and the intended redundancy level. For example, b=AS for AMR-WB at IPv6 should be set to 38 if mode-set includes {6.60, 8.85, 12.65}, the packetization time is 20, and if no extra bandwidth is allocated for redundancy. Likewise, b=AS for EVS Primary mode at IPv4 should be set to 42 if br=7.2-24.4, the packetization is header-full payload format, ptime=20, and no extra bandwidth is allocated for redundancy.

If an MTSI client does not include an AMR or AMR-WB mode-set, or EVS Primary mode br or br-recv parameter in the SDP offer or answer, the MTSI client shall set the b=AS parameter in the SDP to a value matching the highest AMR/AMR-WB mode, i.e., AMR 12.2 and AMR-WB 23.85, or the highest bit-rate of EVS Primary mode depending on negotiated bandwidth(s), i.e., EVS 24.4 for NB and EVS 128 for WB, SWB and FB, respectively.

NOTE 1: When no mode-set is defined, then this should be understood as that the offerer or answerer is capable of sending and receiving all codec modes of AMR or AMR-WB. An MTSI client in terminal will not include the mode-set parameter in SDP offer in the initial offer-answer negotiation. See Clause 6.2.2.2, Tables 6.1 and 6.2. It is however expected that the mode-set is defined when an SDP offer is received from an MTSI MGW inter-working with CS GERAN/UTRAN, see Clause 6.2.2.3, Table 6.5.

The bandwidth to use for b=AS for AMR and AMR-WB, and EVS Primary mode should be computed as shown in Annexes K and Q respectively. Tables 6.7 and 6.8 shows the bandwidth for the respective AMR and AMR-WB codec when the packetization time is 20 and no extra bandwidth is allocated for redundancy. The b=AS value is computed without taking statistical variations, e.g., the effects of DTX, into account. Such variations can be considered in the scheduling and call admission control. Detailed procedures to compute b=AS of AMR and AMR-WB, and EVS Primary mode can be found in Annexes K and Q.

NOTE 2: For any payload format, b=AS of EVS Primary mode at 5.9 kbps source controlled variable bit-rate (SC-VBR) coding is computed as the b=AS of its highest component bit-rate, 8 kbps.

NOTE 3: b=AS of EVS AMR-WB IO mode can be computed as in the octet-aligned payload format of AMR-WB as shown in Annex K.

b=AS of EVS shall be equal to the maximum of b=AS of the highest included EVS primary mode and b=AS of the highest included EVS AMR-WB IO mode, regardless of the presence and configuration of evs-mode-switch.

Table 6.7: b=AS for each codec mode of AMR when ptime is 20

| Payload format | | Codec mode | | | | | | | |
|---------------------|------|------------|------|-----|-----|-----|------|------|------|
| | | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 |
| Bandwidth-efficient | IPv4 | 22 | 22 | 23 | 24 | 24 | 25 | 27 | 29 |
| | IPv6 | 30 | 30 | 31 | 32 | 32 | 33 | 35 | 37 |
| Octet-aligned | IPv4 | 22 | 22 | 23 | 24 | 25 | 25 | 28 | 30 |
| | IPv6 | 30 | 30 | 31 | 32 | 33 | 33 | 36 | 38 |

Table 6.8: b=AS for each codec mode of AMR-WB when ptime is 20

| Payload format | | Codec Mode | | | | | | | | |
|---------------------|------|------------|------|-------|-------|-------|-------|-------|-------|-------|
| | | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 |
| Bandwidth-efficient | IPv4 | 24 | 26 | 30 | 31 | 33 | 35 | 37 | 40 | 41 |
| | IPv6 | 32 | 34 | 38 | 39 | 41 | 43 | 45 | 48 | 49 |
| Octet-aligned | IPv4 | 24 | 26 | 30 | 32 | 33 | 36 | 37 | 40 | 41 |
| | IPv6 | 32 | 34 | 38 | 40 | 41 | 44 | 45 | 48 | 49 |

Table 6.9: b=AS for each bit-rate of EVS Primary mode when ptime is 20

| Payload format | Bit-rate |
|----------------|----------|
|----------------|----------|

| | | | | | | | | | | | | |
|-------------|------|-----|----|-----|------|------|------|----|----|----|-----|-----|
| | | 7.2 | 8 | 9.6 | 13.2 | 16.4 | 24.4 | 32 | 48 | 64 | 96 | 128 |
| Header-full | IPv4 | 24 | 25 | 27 | 30 | 34 | 42 | 49 | 65 | 81 | 113 | 145 |
| | IPv6 | 32 | 33 | 35 | 38 | 42 | 50 | 57 | 73 | 89 | 121 | 153 |

Tables 6.10-1 to 6.10-3 describe the setting of the bandwidth properties that should be used for the 'a=bw-info' attribute for a few possible combinations of codec, codec rate, packetization schemes and redundancy levels. The Minimum Supported Bandwidth does not prevent encoding the speech with an even lower bitrate, for example when EVS is used in the 5.9 kbps VBR mode or during DTX periods when SID frames are encoded with a very low bit rate and are generated with a reduced frame rate. Bit rates lower than the Minimum Supported Bandwidth may also be used when sending DTMF. Additional combinations and corresponding bandwidth properties are found in Annex K for AMR, AMR-WB and EVS AMR-WB IO mode and in Annex Q for EVS primary mode.

Table 6.10-1: Recommended bandwidth properties for AMR to be used with the 'a=bw-info' attribute when codec modes up to 12.2 are negotiated

| Parameter | Assumed setting |
|---|--------------------------------|
| Negotiated codec modes | 4.75, 5.9, 7.4, 12.2 |
| Codec mode used without redundancy | 12.2 |
| Codec mode used with redundancy | 5.9 |
| Payload format | AMR/AMR-WB bandwidth-efficient |
| Minimum frame aggregation | 1 frame per packet |
| Maximum frame aggregation | 4 frames per packet |
| Maximum redundancy level | 100% |
| Redundancy offset | 0 |
| IP version | 6 |
| Bandwidth property | Value |
| Maximum Supported Bandwidth | 37 (see NOTE 1) |
| Maximum Desired Bandwidth | 37 |
| Minimum Desired Bandwidth | 31 |
| Minimum Supported Bandwidth | 13 (see NOTE 2) |
| NOTE 1: If redundancy is needed for higher codec modes, if additional redundancy is needed, or if redundancy offset is needed then the Maximum Supported Bandwidth needs to be set to a higher value. | |
| NOTE 2: The Minimum Supported Bandwidth is calculated based on the lowest codec rate when maximum frame aggregation is used. | |

Table 6.10-2: Recommended bandwidth properties for AMR-WB to be used with the ‘a=bw-info’ attribute when codec modes up to 12.65 are negotiated

| Parameter | Assumed setting |
|---|--------------------------------|
| Negotiated codec modes | 6.6, 8.85, 12.65 |
| Codec mode used without redundancy | 12.65 |
| Codec mode used with redundancy | 6.6 |
| Payload format | AMR/AMR-WB bandwidth-efficient |
| Minimum frame aggregation | 1 frame per packet |
| Maximum frame aggregation | 4 frames per packet |
| Maximum redundancy level | 100% |
| Redundancy offset | 0 |
| IP version | 6 |
| Bandwidth property | Value |
| Maximum Supported Bandwidth | 38 (see NOTE 1) |
| Maximum Desired Bandwidth | 38 |
| Minimum Desired Bandwidth | 32 |
| Minimum Supported Bandwidth | 13 (see NOTE 2) |
| NOTE 1: If redundancy is needed for higher codec modes, if additional redundancy is needed, or if redundancy offset is needed then the Maximum Supported Bandwidth needs to be set to a higher value. | |
| NOTE 2: The Minimum Supported Bandwidth is calculated based on the lowest codec rate when maximum frame aggregation is used. | |

Table 6.10-3: Recommended bandwidth properties for EVS to be used with the ‘a=bw-info’ attribute when codec modes up to 13.2 are negotiated

| Parameter | Assumed setting |
|---|----------------------------|
| Negotiated codec modes | 5.9 – 13.2 |
| Codec mode used without redundancy | 13.2 |
| Codec mode used with redundancy | 7.2 |
| Payload format | EVS |
| Minimum frame aggregation | 1 frame per packet |
| Maximum frame aggregation | 4 frames per packet |
| Maximum redundancy level | 100% |
| Redundancy offset | 0 |
| IP version | 6 |
| Bandwidth property | Value |
| Maximum Supported Bandwidth | 40 (see NOTE 1 and NOTE 2) |
| Maximum Desired Bandwidth | 38 |
| Minimum Desired Bandwidth | 32 (see NOTE 2) |
| Minimum Supported Bandwidth | 14 (see NOTE 2) |
| NOTE 1: If redundancy is needed for higher codec modes, if additional redundancy is needed, or if redundancy offset is needed then the Maximum Supported Bandwidth needs to be set to a higher value. | |
| NOTE 2: The bandwidth is calculated assuming EVS 7.2 kbps when maximum frame aggregation is used. | |

SDP examples using the ‘a=bw-info’ attribute for speech are shown in annex A.6.3.

6.2.5.3 Video

The b=AS parameter is set to match the maximum bit rate desired for the media in the receiving direction.

Video codecs are usually capable of adapting the bit rate over a large bit rate range. This allows for dynamic addition of redundancy (see clauses 6.2.3.5 and 6.2.3.6) without explicitly allocating any additional bandwidth for the redundancy information. Therefore, when the 'a=bw-info' attribute is used, the Maximum Supported Bandwidth and the Maximum Desired Bandwidth properties should be set to the same value.

SDP examples using the 'a=bw-info' attribute for video are shown in annex A.6.3.

6.2.6 The Synchronization Info attribute "3gpp_sync_info"

Synchronization jitter (also known as synchronization or inter-media skew) is defined as the amount of synchronization delay between media streams that needs to be maintained during the synchronization process (at the receiver side), which is acceptable to a session (or the sender of the multimedia streams) for a good user experience.

Tight synchronization between the constituent streams is not necessary for all types of MTSI sessions. For instance, during a VoIP call, one of the call participants may wish to share a video clip or share his/her camera view. In this situation, the sender may want to relax the requirement on the receiver to synchronize the audio and the video streams in order to maintain a good video quality without stressing on tight audio/video synchronization. The Synchronization Info attribute defined in the present document is not just limited to lip-sync between audio/video streams, but is also applicable to any two media streams that need to be synchronized during an MTSI session. This attribute allows an MTSI client to specify whether or not media streams should be synchronized. In case the choice is to have synchronization between different streams, it is up to the implementation, use case and application to decide the exact amount of synchronization jitter allowed between the streams to synchronize.

The ABNF for the synchronization info attribute is described as follows:

```
Synchronization-Info      = "a" "=" "3gpp_sync_info" ":" sync-value
sync-value                 = "Sync" / "No Sync"
```

The value "Sync" indicates that synchronization between media shall be maintained. The value "No Sync" indicates that No Synchronization is required between the media.

The parameter "3gpp_sync_info" should be included in the SDP at the session level and/or at the media level. Its usage is governed by the following rules:

1. At the session level, the "3gpp_sync_info" attribute shall be used with the group attribute defined in RFC 3388 [48]. The group attribute indicates to the receiver which streams (identified by their mid attributes) that are to be synchronized. The "3gpp_sync_info" attribute shall follow the "group: LS" line in the SDP.
2. At the media level, the "3gpp_sync_info" attribute shall assume a value of "No Sync" only. It indicates to the receiver that this particular media stream is not required to be synchronized with any other media stream in the session. The use of the "mid" attribute of RFC 3388 [48] is optional in this case. If the "mid" attribute is used for any other media in the session, then "mid" with this media line shall be used also according to RFC 3388 [48]. Otherwise, it is not necessary to tie the "3gpp_sync_info" attribute with the "mid" attribute.
3. When the "3gpp_sync_info" attribute is defined at both session level (with the "group" attribute) and media level, then the media level attribute shall override the session level attribute. Thus if the "3gpp_sync_info" attribute is defined at the media level, then that particular media stream is not to be synchronized with any other media stream in the session (even if the "3gpp_sync_info" is defined at the session level for this media stream).

The calling party (or the initiator or offerer of the multimedia stream) should include the "3gpp_sync_info" attribute in the SDP which is carried in the initial INVITE message. Upon reception of the INVITE message that includes the "3gpp_sync_info" attribute, the other party in the session should include its own "3gpp_sync_info" attribute (with its own wish for synchronization or no synchronization) in the 200/OK response message.

There are no offer/answer implications on the "3gpp_sync_info" attribute; it provides synchronization requirement between the specified media streams to the receiver. The "3gpp_sync_info" attribute in the calling party SDP is only an indication to the called party of the synchronization requirement that should be maintained between the specified media streams that it receives. Similarly the "3gpp_sync_info" attribute value from the called party is an indication to the calling party of the synchronization requirements between specified media streams. The "3gpp_sync_info" attribute value can be different for the calling and the called parties.

SDP examples using the "3gpp_sync_info" attribute are given in clause A.7.

NOTE: Default operation in the absence of the "3gpp_sync_info" attribute in SDP is to maintain synchronization between media streams.

6.2.7 Negotiated QoS parameters

6.2.7.1 General

The MTSI client in the terminal may support the negotiation of the QoS. The term "negotiated" in the present document describes the end result of a QoS negotiation between an MTSI client in terminal and the network (or the end result of what the network grants to the MTSI client in terminal even if no negotiation takes place).

An MTSI client in terminal supporting the transport level QoS negotiation should verify that the QoS parameters, e.g. MBR and GBR, are aligned with the media configurations negotiated in SDP. While checking whether the different bandwidth parameters are aligned or not, the MTSI client in terminal should take into account the following areas where differences are likely to occur: parameter encoding; units used; packetization schemes related to the IP/UDP/RTP overhead; the amount of RTCP bandwidth. The MTSI client in terminal needs to compensate for the differences if found.

NOTE: The transport level QoS negotiation applies only to 3GPP accesses.

In the following sections it is assumed that the MTSI client in terminal supports the QoS negotiation and is made aware of the negotiated bandwidth properties and other QoS hints (if supported and used, see clause 6.2.7.4).

6.2.7.2 Alignment of negotiated QoS parameters and b=AS bandwidth modifier

In SDP offer-answer, the b=AS bandwidth modifier is used to describe the maximum bandwidth for the receiving direction. The IP/UDP/RTP overhead is included in the bandwidth value for RTP-based media but the RTCP bandwidth is not included. The IP/UDP/DTLS/SCTP overhead is included in the bandwidth value for SCTP-based media such as the data channel (see clause 6.2.10). The bandwidth value is an integer and the unit is kbps.

If, at session setup or at session re-negotiation, the MTSI client in terminal detects that the negotiated downlink QoS Maximum Bit Rate (MBR) is not aligned with the b=AS bandwidth modifier in the sent SDP then it should try to align the bandwidth properties in a subsequent SDP offer-answer.

If, during the session, the negotiated downlink Maximum Bit Rate(s) (MBR) for the bearer(s) has been updated from the network then the MTSI client in terminal should check if the bandwidth(s) it sent within b=AS bandwidth modifiers in previous SDP (e.g. during the initial session setup or earlier session re-negotiation, if any) are aligned with the downlink MBR(s) allocated for the bearer(s) and its receiving capabilities.

The rules for alignment are different depending on how many media streams that are handled by the bearer, as follows:

- When a bearer carries a single media stream, then it is the media-level b=AS bandwidth for that media stream that should be aligned with the MBR of the bearer.
- When a bearer carries several media streams, then it is the sum of the media-level b=AS bandwidths for those media streams that should be aligned with the MBR of the bearer.

The rules for alignment are also different depending on whether the media stream(s) are bi-directional (sendrecv) or uni-direction (sendonly or recvonly), as follows:

- If the MTSI terminal receives (and possibly sends) a media stream, it should consider the sent b=AS bandwidth(s) in SDP for that media stream in comparison with the downlink MBR.
- If the MTSI terminal only sends a media stream, it should not consider the sent b=AS bandwidth(s) in SDP for that media stream in comparison with the downlink MBR.

NOTE 1: The b=AS bandwidth and the MBR bandwidth are not directly comparable since the b=AS bandwidth does not include the RTCP bandwidth while the MBR bandwidth allocation for RTP media must include some headroom for RTCP. The bandwidths will therefore almost always be different. The MBR bandwidth may also differ from the b=AS bandwidth because of other reasons, for example: bearer allocation and header compression. It is an implementation consideration to handle such impacts and how to judge whether the bandwidth values differ and what bandwidth value to send in the UPDATE message.

If the MTSI client in terminal determines that the b=AS bandwidth(s) are not aligned with the MBR and the receiving capabilities of the MTSI client, then it should align the media-level b=AS bandwidth(s) to the MBR and its receiving capabilities by sending to the other party an SDP offer with the new b=AS bandwidth value(s). In the process of this alignment it is also likely that the session-level b=AS bandwidth needs to be updated. In addition, the MTSI client in terminal may modify other parts of the SDP, e.g., to replace the codecs or adjust codec parameters (such as the AMR or AMR-WB mode-set).

NOTE 2: It could be necessary to reconfigure the codec(s), or even to drop some media streams, to be able to operate within the bandwidth constraints defined with the MBR of the bearer.

NOTE 3: A situation when the MTSI client in terminal could not align the b=AS bandwidth(s) with the MBR, due to its receiving capabilities, is when it does not support rate adaptation.

This alignment may require re-negotiation, which should preferably be performed when there is session re-negotiation for other reasons. When additional re-negotiation is required, which adds load to the SIP bearer and the SIP servers, it is not desirable to repeat the re-negotiation multiple times. Therefore, if re-negotiation fails to align the b=AS bandwidth modifier and the QoS parameters then it should not be repeated unless new re-negotiation is needed for other reasons, e.g. to add or remove media components.

If an MTSI client in a terminal receives a new SDP offer with new b=AS bandwidth value(s) (e.g., in a SIP UPDATE or in a SIP re-INVITE) and it accepts the session update then it shall generate an SDP answer as described in clause 6.2.

Any subsequent QoS changes indicated to the MTSI client in terminal during an MTSI session (including the cases described in Clause 10.3) shall be signalled by the MTSI client in terminal (subject to the QoS update procedure) to the other party using the same signalling described above.

Examples of SDP using negotiated QoS are given in clause A.8.

When the MTSI client in terminal receives an indication that the negotiated uplink Maximum Bit Rate (MBR) is less than the current maximum sending rate of its sender, the MTSI client in terminal should configure the maximum sending rate of its sender to align with the negotiated uplink Maximum Bit Rate (MBR).

When the MTSI client in terminal receives an indication that the negotiated uplink Maximum Bit Rate (MBR) is greater than the current maximum sending rate of its sender and rate adaptation is possible for the session, the MTSI client in terminal should configure the maximum sending rate of its sender to align with the smallest of the following:

- the negotiated uplink Maximum Bit Rate (MBR)
- the bandwidth value, if the b=AS parameter was included in the last SDP offer or answer received by the client
- the preconfigured data rate for the selected codec, if the MTSI client has been preconfigured by the operator to use a particular data rate for the selected codec

6.2.7.3 Alignment of negotiated QoS parameters and a=bw-info attribute

The Maximum Supported Bandwidth for the sending direction should be aligned with the uplink MBR.

The Minimum Desired Bandwidth for the sending direction should be aligned with the uplink GBR.

The Maximum Supported Bandwidth for the receiving direction should be aligned with the downlink MBR.

The Minimum Desired Bandwidth for the receiving direction should be aligned with the downlink GBR.

The procedures for aligning the above listed bandwidth properties with the above listed QoS parameters are the same as given above in clauses 6.2.7.1 and 6.2.7.2 for other bandwidth-related information.

6.2.7.4 The a=3gpp-qos-hint SDP attribute

6.2.7.4.1 General

In some cases, it is not possible to uniquely map a media type in SDP to an appropriate QoS without additional information on the intended usage of that media from the application or the end-user of that application.

The MTSI client in terminal may include a non-authoritative QoS hint in one or more SDP media description(s), for use by local and remote policy control functions, by adding an "a=3gpp-qos-hint" media-level SDP attribute with a value that is set based on the intended media usage. The QoS hints included on the "a=3gpp-qos-hint" line applies jointly to the aggregate of all packets (e.g. if more than one media stream is part of the same media description), and equally in both directions (uplink and downlink) unless restricted by the inclusion of "a=sendonly" or "a=recvonly" in the same media description.

If this attribute is included in an SDP media description, policy control can choose to take this additional information into account for the affected media.

An "a=3gpp-qos-hint" attribute shall not occur more than once for an SDP media description.

SDP examples are provided in Annex A.16.

NOTE: If a media GBR bearer that was set up assisted by 3gpp-qos-hint information is dropped by the network due to insufficient resources, an attempt to re-establish the bearer can be made through a re-offer in an UPDATE or re-INVITE with less demanding QoS values included in the 3gpp-qos-hint and/or bandwidth attributes (e.g. "b=AS" and "a=bw-info"; see clause 19) for that media. How to get information on what values that will be acceptable to include in such re-offer is not specified.

6.2.7.4.2 3gpp-qos-hint ABNF syntax and semantics

3gpp-qos-hint-value = qos-hint *(";" qos-hint)

qos-hint = qos-hint-property ["=" qos-hint-end-to-end-value *(qos-hint-split)]

qos-hint-property = "loss" / "latency" / token

qos-hint-end-to-end-value = qos-hint-value

qos-hint-split = "/" qos-hint-split-method ":" qos-hint-split-value

qos-hint-split-method = "local" / token

qos-hint-split-value = qos-hint-value

qos-hint-value = zero-based-integer / non-zero-real / token

; token as defined by IETF RFC 4566

; zero-based-integer and non-zero-real as defined by IETF RFC 8866The IANA registration information for this attribute is provided in Annex M.11.

A qos-hint-property value shall only occur once on the "a=3gpp-qos-hint" line. If a qos-hint-property value is not included on the "a=3gpp-qos-hint" line, it should be interpreted as the UE and application have no preference of any qos-hint-value for that qos-hint-property but anything the network can provide is equally acceptable.

If a qos-hint-property has no qos-hint-end-to-end-value, it is of boolean (on/off) type.

If the qos-hint-property has a qos-hint-end-to-end-value but doesn't include any explicit qos-hint-split, the qos-hint-end-to-end-value is split equally between the SDP offerer and the SDP answerer. If an explicit qos-hint-split is included, it specifies how the split of the qos-hint-end-to-end-value between SDP offerer and SDP answerer is made. The following qos-hint-split-method values are currently defined:

"local": The qos-hint-split-value specifies the SDP sender's part of the qos-hint-end-to-end-value that is applied across its local link, in the same format and units used by the associated qos-hint-end-to-end-value, the value being the same in both send and receive directions. The SDP sender of the SDP offer is the SDP offerer and the SDP sender of the SDP answer is the SDP answerer.

The following qos-hint-property values are currently defined:

"loss": This qos-hint-property qos-hint-end-to-end-value describes the maximum desirable end-to-end transport level packet loss rate in percent (without "%" sign) as a zero-based-integer or as a non-zero-real value.

"latency": This qos-hint-property qos-hint-end-to-end-value describes the maximum desirable end-to-end transport level packet latency in milliseconds as a zero-based-integer or as a non-zero-real value. The value excludes any application-level processing in the sender and receiver, such as e.g. application-level retransmission or encoding/decoding.

6.2.7.4.3 Creating an SDP offer

An "a=3gpp-qos-hint" line may be included in any media description in an SDP offer.

6.2.7.4.4 Creating an SDP answer

If there is no "a=3gpp-qos-hint" line included in a media description in the SDP offer, it shall not be included in the corresponding media description in the SDP answer.

If a qos-hint with an unknown or malformed qos-hint-property or qos-hint-value is received in an SDP offer, that qos-hint shall be ignored, shall be omitted from the SDP answer, and shall neither cause the "a=3gpp-qos-hint" line nor the entire media description to be rejected in the SDP answer.

An SDP answerer shall not add any qos-hint-property values on the "a=3gpp-qos-hint" line that are not present in the received SDP offer.

A "loss" qos-hint-end-to-end-value received in the SDP offer that is accepted by the SDP answerer shall be kept unmodified in the SDP answer. A "loss" qos-hint-end-to-end-value received in the SDP offer that cannot be supported by the SDP answerer even when making use of an allowable qos-hint-split-value (see below) may be increased in the SDP answer to a value that can be supported by the SDP answerer. A "loss" qos-hint-end-to-end-value received in the SDP offer that is higher than required by the SDP answerer may be decreased in the SDP answer to a value that is required by the SDP answerer. Figures 6.2.7.4.4-1 and 6.2.4.4.4-2 provide illustrated examples of these principles. If the "loss" qos-hint-property is known by the SDP answerer but if there are no known, supported qos-hint-values for it, the entire qos-hint shall be omitted from the SDP answer.

A "latency" qos-hint-value received in the SDP offer that cannot be supported by the SDP answerer as the desirable maximum end-to-end packet latency should be increased in the SDP answer to a value that can be supported by the SDP answerer. A "latency" qos-hint-end-to-end-value received in the SDP offer that is higher than required by the SDP answerer may be decreased in the SDP answer to a value that is required by the SDP answerer. Figures 6.2.7.4.4-1 and 6.2.7.4.4-2 provide illustrated examples of the principles for "loss" that is also applicable for "latency". If the "latency" qos-hint-property is known by the SDP answerer but if there are no known, supported qos-hint-values for it, the entire qos-hint shall be omitted from the SDP answer.

If, as a result of the above procedures, there are no qos-hints to be included in the SDP answer, the entire "a=3gpp-qos-hint" line shall be omitted from the SDP answer.

NOTE: If the qos-hint-end-to-end-value is changed from what was included in the offer, or if the implicit or explicit qos-hint-split-value applicable to the SDP offerer is decreased between the SDP offer and the SDP answer, there's a risk that the resulting QoS will not be acceptable to the SDP offerer or, if anyway accepted, at least cause sub-optimal user experience.

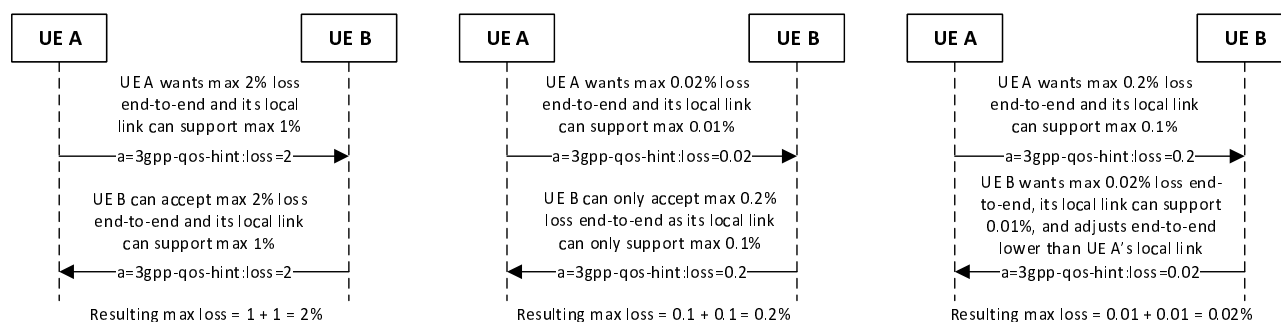


Figure 6.2.7.4.4-1 Illustration of 3gpp-qos-hint "loss" UE-to-UE offer/answer

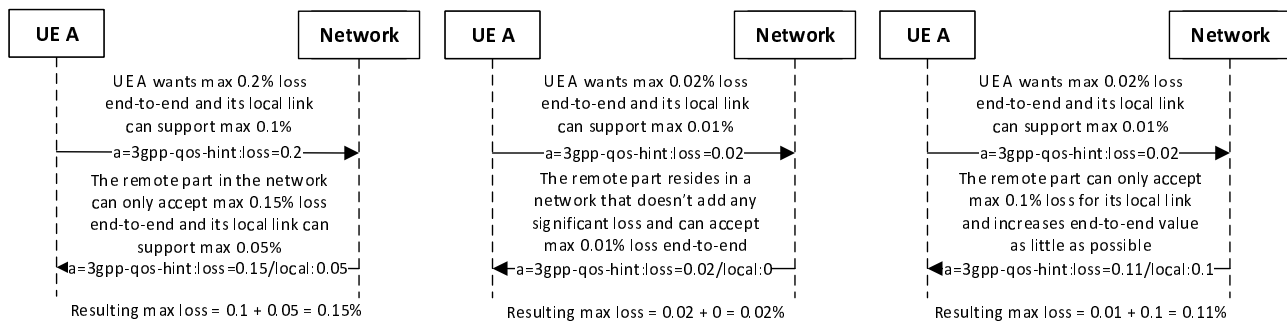


Figure 6.2.7.4.4-2 Illustration of 3gpp-qos-hint "loss" UE-to-Network offer/answer

If a qos-hint with an unknown or malformed qos-hint-split-method or qos-hint-split-value is received in an SDP offer, the entire qos-hint-split shall be ignored, shall be omitted from the SDP answer, and shall neither cause the "a=3gpp-qos-hint" line nor the entire media description to be rejected in the SDP answer.

The qos-hint-split provides the SDP offerer's opinion on how that qos-hint-end-to-end-value should be split between the SDP offerer's and the SDP answerer's local links. As stated above, if no qos-hint-split is provided the qos-hint-end-to-end-value is split equally between SDP offerer and SDP answerer local links and is equivalent to a qos-hint-split being provided with a value that is half of the qos-hint-end-to-end-value.

If the SDP answerer accepts a default split (without explicit qos-hint-split in the SDP offer) it shall not include any qos-hint-split in the SDP answer.

If the SDP answerer accepts an explicitly provided qos-hint-split proposed by the SDP offer, it shall include a qos-hint-split in the SDP answer with a qos-split-hint-value being equal to qos-hint-end-to-end-value in the SDP answer minus the corresponding qos-hint-split-value from the SDP offer.

If the SDP answerer does not accept the qos-hint-split-value proposed in the SDP offer, regardless if that qos-hint-split-value is explicit or not, it can make one of the following choices for the SDP answer:

- If the SDP offerer qos-hint-split-value is smaller than what the SDP answerer requires (i.e., qos-hint-end-to-end-value in the SDP offer minus the corresponding qos-hint-split-value from the SDP offer is larger than required across the SDP answerer's local link), the SDP answerer may include a qos-hint-split-value in the SDP answer that is less than the qos-hint-end-to-end-value in the SDP answer minus the corresponding qos-hint-split-value from the SDP offer. The SDP answerer may use the value 0 if the actual (non-zero) qos-hint-split-value can be considered insignificant compared to the qos-hint-split-value in the SDP offer.
- If the SDP offerer qos-hint-split-value is larger than what the SDP answerer considers feasible (i.e., qos-hint-end-to-end-value in the SDP offer minus the corresponding qos-hint-split-value from the SDP offer is smaller than feasible across the SDP answerer's local link), the SDP answerer may include a qos-hint-split-value in the SDP answer that is larger than the qos-hint-end-to-end-value included in the SDP offer minus the corresponding qos-hint-split-value from the SDP offer, but the included value shall then also be less than or equal to half of the qos-hint-end-to-end-value included in the SDP answer. The exception to that rule is when the qos-hint-end-to-end-value included in the SDP answer is less than the qos-hint-end-to-end-value included in the SDP offer (see also above), in which case the qos-hint-split-value should instead be the qos-hint-end-to-end-value included in the SDP answer minus the corresponding qos-hint-split-value from the SDP offer, unless the SDP answerer cannot support such qos-hint-split-value. In that case the included value shall be less than or equal to half of the qos-hint-end-to-end-value included in the SDP answer.

6.2.7.4.5 Offerer receiving an SDP answer

If there is no "a=3gpp-qos-hint" line included in a media description in the SDP answer, it shall be interpreted as the answerer either does not support the attribute at all or cannot accept any of the qos-hint-property values from the offer, and QoS hints will therefore not be used for that media description.

If a qos-hint with an unknown or malformed qos-hint-property or qos-hint-value is received in an SDP answer, that qos-hint shall be ignored, and shall neither cause the "a=3gpp-qos-hint" line nor the entire media description to be rejected (e.g. by issuing a new SDP offer with that "m=" line disabled).

A "loss" property value received in the SDP answer that is identical to the SDP offer shall be taken as the SDP answerer accepting to share end-to-end packet loss budget equally. A "loss" property value received in the SDP answer that is

larger than in the SDP offer shall be taken as the SDP answerer being incapable of sharing end-to-end packet loss budget with the proposed split from the SDP offer while keeping the end-to-end packet loss value that was included in the SDP offer (see illustrative examples in Figures 6.2.7.4.4-1 and 6.2.7.4.4-2). A "loss" property value received in the SDP answer that is smaller than in the SDP offer shall be taken as the SDP answerer requiring a lower end-to-end packet loss for the media.

A "latency" property value received in the SDP answer that is identical to the SDP offer shall be taken as the SDP answerer accepting to share end-to-end packet latency budget equally. A "latency" property value received in the SDP answer that is larger than in the SDP offer shall be taken as the SDP answerer being incapable of sharing end-to-end packet latency budget with the proposed split from the SDP offer while keeping the end-to-end packet latency value that was included in the SDP offer (the principles in the illustrative examples for "loss" in Figures 6.2.7.4.4-1 and 6.2.7.4.4-2 apply also for "latency"). A "latency" property value received in the SDP answer that is smaller than in the SDP offer shall be taken as the SDP answerer requiring a lower end-to-end latency for the media.

If a qos-hint with an unknown or malformed qos-hint-split-method or qos-hint-split value is received in an SDP answer, the entire qos-hint-split shall be ignored, and shall neither cause the "a=3gpp-qos-hint" line nor the entire media description to be rejected (e.g. by issuing a new SDP offer with that "m=" line disabled).

If no qos-hint-split is included in the received SDP answer and if no qos-hint-split was included in the SDP offer, the SDP answerer has accepted the offered default, equal split.

If a qos-hint-split is included in the received SDP answer with a qos-split-hint-value equal to the qos-hint-end-to-end-value in the SDP answer minus the corresponding qos-hint-split-value from the SDP offer, the SDP answerer has accepted the qos-hint-split proposed by the SDP offer.

If a qos-hint-split is included in the received SDP answer with a qos-split-hint-value different than the above, the SDP answerer has modified the qos-hint-split-value allocation from the SDP offer, regardless if that qos-hint-split-value was explicit or not in the SDP offer, and the resulting split is described by the SDP answer.

Table 6.2.7.4.5-1 below shows what resulting QoS hint values from the SDP answer that can be used during resource reservation at the SDP offerer and SDP answerer sides, respectively, for each of the defined qos-hint-property-values.

Table 6.2.7.4.5-1: Resulting summary of QoS hint values from SDP answer

| QoS hint condition | SDP offerer side part | SDP answerer side part |
|----------------------------|--|--|
| No qos-hint-split included | $0.5 * \text{qos-hint-end-to-end-value}$ | $0.5 * \text{qos-hint-end-to-end-value}$ |
| qos-hint-split included | $\text{qos-hint-end-to-end-value} - \text{qos-hint-split-value}$ | qos-hint-split-value |

NOTE: The following *non-authoritative* mapping of the qos-hint-end-to-end-values (no explicit split applied, for simplicity) to, *for example*, the 5 defined FLUS-related 5QI/QCI would then be possible for the PCF/PCRF (other mappings are possible, left for the originating and terminating policy control to decide, and need not even be exactly the same for originating and terminating network for a UE-to-UE session):

loss=0.0002;latency=300 => QCI 71 (150 ms, 10^{-6})
 loss=0.02;latency=600 => QCI 72 (300 ms, 10^{-4})
 loss=0.000002;latency=600 => QCI 73 (300 ms, 10^{-8})
 loss=0.000002;latency=1000 => QCI 74 (500 ms, 10^{-8})
 loss=0.02;latency=1000 => QCI 76 (500 ms, 10^{-4})

6.2.8 Delay Budget Information (DBI) Signaling

An MTSI client may support Delay Budget Information (DBI) signaling as specified in sub-clause 7.3.8.

An MTSI client supporting DBI shall offer 'Delay Budget Information' (DBI) signaling in SDP for all media streams containing speech. An MTSI client supporting DBI may also offer 'Delay Budget Information' (DBI) signaling in SDP for all media streams containing video. DBI shall be offered by including the a=rtcp-fb attribute [40] with the DBI type under the relevant media line scope. The DBI type in conjunction with the RTCP feedback method shall be expressed with the following parameter: 3gpp-delay-budget. A wildcard payload type ("*") shall be used to indicate that the RTCP feedback attribute for DBI signaling applies to all payload types. Here is an example usage of this attribute to signal DBI relative to a media line based on the RTCP feedback method:

a=rtcp-fb:* 3gpp-delay-budget

The IANA registration information on the new RTCP feedback type for DBI signaling is provided in Annex R.2.

The ABNF for `rtcp-fb-val` corresponding to the feedback type "3gpp-delay-budget" is given as follows:

```
rtcp-fb-val =/ "3gpp-delay-budget"
```

As described in sub-clause 7.3.8, DBI signalling involves RTCP feedback signalling to carry both (i) available additional delay budget from the MTSI receiver to the MTSI sender, and (ii) requested additional delay budget from the MTSI sender to the MTSI receiver.

Annex V.3 presents SDP examples on DBI signalling capability.

6.2.9 ANBR Support attribute "anbr"

Access network bitrate recommendation (ANBR) is described in clause 10.7. Use of ANBR with dynamic bitrate adaptation is described in clause 10.7.3 and related adaptation of sent and received media is described in clauses 10.7.3.2 and 10.7.3.3, respectively. At the radio signalling level, ANBR signaling capability, also known as RAN-assisted codec adaptation, is specified in TS 36.321 [157] for LTE access and TS 38.321 [166] for NR access respectively.

The media-level SDP attribute "anbr" is specified in this clause to indicate ANBR support. The MTSI client in terminal supporting "anbr" shall only signal this attribute in the SDP offer and answer if all of the following are true:

- MTSI client in terminal supports ANBR as described in clause 10.7, including the use of ANBR with dynamic bitrate adaptation as described in clause 10.7.3.
- The UE of the MTSI client in terminal is capable of RAN-assisted codec adaptation specified in TS 36.321 for LTE access and/or TS 38.321 for NR access. For LTE access, inclusion of "anbr" in the SDP indicates that the UE is able to query and receive ANBR information (for both downlink and uplink ANBR) from its eNB. Likewise, for NR access inclusion of this attribute indicates that the UE is able to query and receive ANBR information (for both downlink and uplink ANBR) from its gNB.
- The P-CSCF has indicated to the UE of the MTSI client in terminal its ability to handle this SDP attribute, as described in clause E.10 of TS 23.228 [167], and specified in TS 24.229 [7] through the respective SIP registration procedures via the corresponding feature capability indicator `g.3gpp.anbr`.

Signalling of ANBR capabilities in the SDP via "a=anbr" enables end-to-end coordination of ANBR capabilities across the UEs, access networks, and PCF/PCRF. In particular, such ANBR capability signalling can be useful for the PCF/PCRF when setting GBR<MBR bearers.

Here is an example use of the "a=anbr" attribute relative to a media line:

```
a=anbr
```

The IANA registration information for the "a=anbr" SDP attribute is provided in Annex M.8.

SDP examples on ANBR capability signalling are provided in Annex A.15.

6.2.10 Data channel

6.2.10.1 General

Support of data channel media is optional for an MTSI client and an MTSI client in terminal. For brevity, an MTSI client supporting data channel is henceforth denoted as a DCMTSI client or DCMTSI client in terminal, respectively.

To indicate support for the procedures in this clause, a DCMTSI client shall when including media feature tags as specified in TS 24.229 [7] include a `+sip.app-subtype` media feature tag, as specified by RFC 5688 [177], with a value of "webrtc-datachannel" (the application media format used by [172]), regardless of data channel media being part of the SDP or not.

One or more data channel SDP media descriptions formatted according to [172] may be added to the SDP, alongside other SDP media descriptions such as e.g. speech, video, and text. A data channel SDP media description shall not be placed before the first SDP speech media description. SDP examples are provided in Annex A.17.

If data channels are used in a session, the session setup shall determine the applicable bandwidth limit(s) as defined in clause 6.2.5.

Multiple data channels may be mapped to a single data channel SDP media description, each with a corresponding "a=dcmap" SDP attribute and stream IDs that are unique within that media description. There is no limit to the number of data channels in an SDP media description, but the aggregate of all defined data channels shall keep within the set bandwidth limit and care should be taken to avoid excessive SDP size. If the session is re-negotiated to include a changed number of data channels in an SDP media description, the bandwidth limit may either be kept constant, changing the share of bandwidth available to each individual data channel, or the bandwidth limit may be changed to accommodate the changed number of data channels, keeping individual data channel bandwidth shares. Regardless of what approach is used when changing number of used data channels in a media description, the aggregate of all defined data channels shall keep within the re-negotiated bandwidth limit.

If there is a need to use data channels with either different transport IP addresses, different UDP ports, or different SCTP ports, separate data channel SDP media descriptions shall be used, as IP address, UDP port and SCTP port are all constant per SDP media description. Multiple SCTP associations for a single channel, commonly denoted as "multi-homing", defined in IETF RFC 4960 [173] for reasons of redundancy and basically using one destination transport address at a time, is not described for use with WebRTC data channel and shall therefore not be used in this specification.

NOTE 1: The main reasons to not specify multi-homing are because it cannot use the needed separation of signalling paths for redundancy purposes in the applicable usage scenarios, and it is also not considered feasible when using SCTP on top of DTLS.

To ease data channel media implementation and ease interworking with WebRTC data channels, DCMTSI clients shall support ICE Lite and may support full ICE [184], for data channel media. DCMTSI clients supporting full ICE shall only use host candidate addresses. SDP "a=candidate" line host address information shall match corresponding SDP "c=" and "m=" line information.

NOTE 2: In typical IMS deployments, it is expected that DCMTSI clients have no need to use STUN or TURN servers with ICE. This is in line with what constitutes an ICE Lite agent.

A "data channel application" consists of an HTML web page including JavaScript(s), and optionally image(s) and style sheet(s). A "bootstrap data channel" is henceforth defined as a data channel used to retrieve data channel application(s) for a DCMTSI client in terminal, with a data channel stream ID below 1000, and using the HTTP [73] protocol as data channel subprotocol. The data channel application accessible at the HTTP root ("/") URL through a bootstrap data channel describes the graphical user interface and the logic needed to handle any further data channel usage beyond the bootstrap data channel itself. The meaning of the "authority" (host) part of the URL and consequently the "Host" HTTP header are not defined, shall be ignored on reception, and shall be set to the empty value by a DCMTSI client in terminal.

NOTE 3: Data channel stream IDs below 1000 may use a well-defined subprotocol for other features than retrieving data channel application(s). For example, the "mpeg-sd" subprotocol can be used for a data channel stream ID below 1000 for scene description-based overlays as specified in Annex Y.6.9.

The data channel application is created prior to the DCMTSI call where it is intended to be used, by means left out of scope for this specification. The data channel application workflow is depicted by Figure 6.2.10.1-1 below.

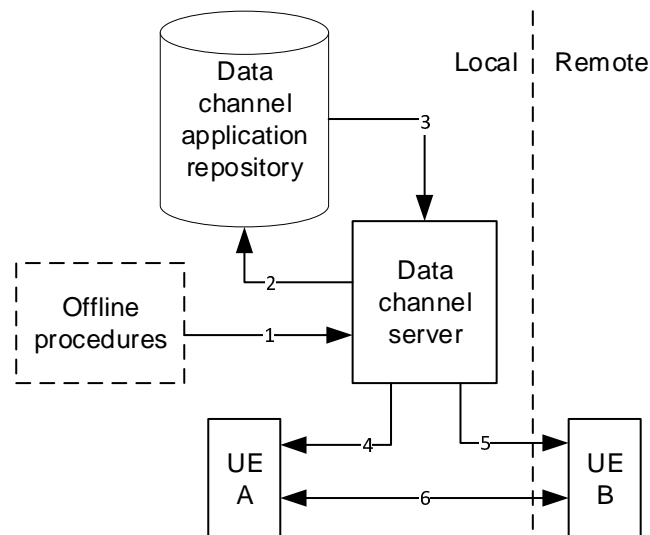


Figure 6.2.10.1-1: Data Channel Workflow

The data channel application is, referring to the numbered arrows in Figure 6.2.10.1-1:

1. Uploaded to the network, by the UE user or some other authorized party.
2. Stored in a data channel application repository in the network.
3. During the DCMTSI call where it should be used, retrieved from the repository.
4. Sent through a bootstrap data channel to the local UE A.
5. Sent through a bootstrap data channel to the remote UE B. This may happen in parallel with and rather independent of step 4.
6. Any additional data channels created and used by the data channel application itself are established (logically) between UE A and UE B. Data transmission on data channels shall not start until there is confirmation that both peers have instantiated the data channel, using the same procedures as described for WebRTC in section 6.5 of [172]. The traffic may effectively go through the Data Channel Server, e.g., when the bootstrap and end-to-end data channels have the same anchoring point. This traffic may pass across an inter-operator border if UE A and UE B belong to different operators' networks.

The bootstrap data channel is not intended for use directly between DCMTSI clients in terminal. DCMTSI clients in terminal that receive HTTP requests on a bootstrap data channel shall ignore such request and shall update the session by removing the SDP "a=dcmap" line with the stream ID where such HTTP request was received, and closing that stream ID.

The data channel application sent in a bootstrap data channel may be updated at any time, automatically or interactively, using normal HTTP procedures.

A bootstrap data channel shall be configured as ordered, reliable, with normal SCTP multiplexing priority. The sub-protocol for a bootstrap data channel shall be HTTP (not encapsulating HTTP in TCP), represented by the following, example SDP "a=dcmap" line, which therefore shall be present in each data channel media description in an SDP offer from a DCMTSI client in terminal:

```
a=dcmap:0 subprotocol="http"
```

When the HTTP subprotocol is used, any other data channels used by the data channel application JavaScript(s) sent in the bootstrap data channel shall be represented in an updated SDP as additional "a=dcmap" lines with stream ID values starting from 1000, using stream ID numbers from the JavaScript(s).

There are multiple, possible providers of data channel applications. In Figure 6.2.10.1-1, assume that UE A is local to the operator hosting the data channel server. Further assume that UE B belongs to a different operator (remote). The user of UE A can create and use data channel applications (steps 1-4), which can also be sent to UE B (step 5). Similarly, some other authorized part associated with UE A's operator can create data channel applications for use by UE A (steps 1-4), which can also be sent to UE B (step 5). For simplicity, there's no data channel server and data

channel application repository depicted for UE B in Figure 6.2.10.1-1, but those could be present in a more general case. Seen from the perspective of a single UE, there are then at least four possible data channel application providers:

1. The local UE user.
2. Other authorized parties associated with the local network (e.g. the local operator).
3. The remote UE user.
4. Other authorized parties associated with the remote network (e.g. the remote operator).

The HTML web content making up a data channel application in each bootstrap data channel represents a different context of user interaction and should open in a separate tab, or some corresponding user interface construct, but the details are out of scope for this specification and left open for individual implementations. It shall be possible to use and navigate between different data channel applications from different bootstrap data channels with different stream IDs that are open simultaneously.

Table 6.2.10.1-2 describes a mandatory mapping between stream ID and bootstrap channel data channel application content sources, as seen from a single (local) DCMTSI client in terminal, each of which shall be listed as separate "a=dcmap" lines with "http" subprotocol in SDP when the DCMTSI client in terminal supports receiving data channel application content from that source.

Table 6.2.10.1-2: Bootstrap Data Channel Content Sources

| Stream ID | Content Source |
|-----------|-------------------------|
| 0 | Local network provider |
| 10 | Local user |
| 100 | Remote network provider |
| 110 | Remote user |

NOTE 4: When the local user has defined and stored multiple, different data channel applications in the local data channel application repository, the local network provider may provide functionality in the stream ID 0 data channel application that enables a dynamic choice of which user-defined data channel application to use with stream ID 10 in the DCMTSI call.

Figure 6.2.10.1-3, referring to Figure 6.2.10.1-1 and Table 6.2.10.1-2, is depicting the stream IDs used for distribution of a data channel application owned by UE A from its local data channel repository to both UE A (stream ID 10) and its remote UE B (stream ID 110).

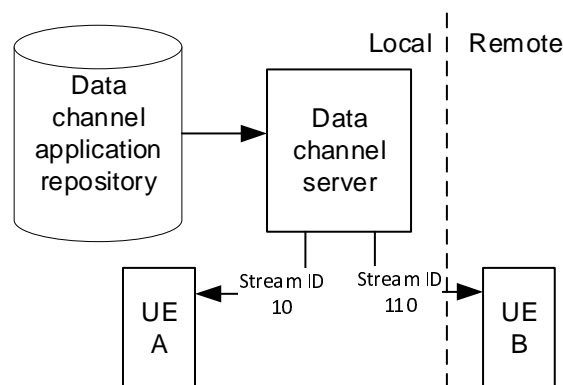


Figure 6.2.10.1-3: Distribution of local data channel application to both UE

6.2.10.2 Generating SDP offer

A DCMTSI client in terminal may include a data channel media description for the "bootstrap" data channels in the initial SDP offer, as described above and according to [172] [184]. A DCMTSI client in terminal may add or disable (by setting port 0, as for RTP media) additional data channel media descriptions as needed in subsequent SDP offers.

A DCMTSI client in terminal that desires to use data channels with stream IDs from a data channel application retrieved from its local "bootstrap" data channel stream ID 0 or 10, shall initiate a subsequent SDP offer after the initial SDP offer, opening those data channels by adding corresponding "a=dcmap" and (optionally) "a=dcsa" lines. A DCMTSI

client in terminal that retrieves a data channel application from a stream ID different than 0 or 10 (e.g. a data channel application from the peer), shall not initiate any subsequent offer to open data channels used by that data channel application.

A data channel media description with specific loss or latency requirements should use "a=3gpp-qos-hint" in the SDP offer, as detailed in section 6.2.7.4. If subsequent SDP offers or answers adds data channels with more strict loss or latency requirements that cannot be met by keeping current "a=3gpp-qos-hint" and providing suitable SCTP "a=dcmmap" parameters, the existing "a=3gpp-qos-hint" should be modified accordingly. Similarly, if subsequent SDP offers or answers closes (removes) data channels that are known to be the limiting factor for choosing the existing "a=3gpp-qos-hint", a more relaxed "a=3gpp-qos-hint" should be chosen to better fit the remaining data channels.

6.2.10.3 Generating SDP answer

An answering DCMTSI client in terminal may accept an SDP offer with data channel as described by [172] [184].

An answering DCMTSI client in terminal that desires to reject the entire SCTP association for all offered data channels shall set the port to 0 (zero) on the corresponding "m=application" line in SDP, as described in [172]. An SCTP association that initially, or as a result of session modification, has no open data channels ("a=dcmmap" lines) should be rejected or closed by modifying the session, setting port number to 0 (zero).

An answering DCMTSI client in terminal that desires to accept some offered data channels and reject others shall indicate this by removing the non-desired data channel "a=dcmmap" and "a=dcsa" lines from the SDP answer, as described in [172]. The DCMTSI client in terminal accepting a data channel must also accept the corresponding, supported "bootstrap" data channels with stream ID <1000 (e.g. a=dcmmap:0 ...).

6.2.10.4 Receiving SDP answer

An offering DCMTSI client in terminal receiving an SDP answer where the data channel SCTP association is accepted (port is not 0) may use any offered stream ID that has a corresponding "a=dcmmap" line in the SDP answer, as described by section 6.5 in [172]. Data channels with "a=dcmmap" lines in the SDP offer that are not included in the SDP answer must be considered as rejected and shall not be used, as described by section 6.5 in [172].

6.2.11 Still Images

If still images are used in a session, then the imageseq SDP attribute shall be present in the offer of the corresponding video media session. The syntax is defined below

$$\text{imageseq} = \text{"a=imageseq:" PT [SP item_count]}$$

PT is the payload type number to which the attribute applies to as indicated by the "m=video" line. Optional parameters can be ignored by an MTSI client if not understood. The parameters have the following semantics:

item_count: provides the number of images in the corresponding image collection or image sequence. For ITT4RT, the parameter is informational from the ITT4RT-Tx client to the ITT4RT-Rx client. An ITT4RT-Rx client shall not include item_count in the SDP offer. If an ITT4RT-Rx client receives item_count in the SDP offer, it should include the same value in the response.

An MTSI client that supports still images shall support long and varying time distances between RTP time stamps. An MTSI client that supports still images should support the HEVC display orientation SEI message as defined in clause D.3.17 and HEVC VUI parameters. An MTSI client that supports and desires to use still images shall in the SDP offer media description of such a bitstream include the "a=imageseq" attribute. An MTSI client that supports still images and that receives the "a=imageseq" attribute in the SDP offer shall keep the "a=imageseq" attribute in the corresponding media description in the SDP answer.

An MTSI client that doesn't support still images shall remove the image attribute in the answer. An MTSI client that sent the "imageseq" attribute in the SDP offer but does not receive it in the corresponding SDP answer, shall not use still images. In such case, the sender should transmit the still image as a regular continuous video stream that conforms to the requirements in 7.4.3 for an HEVC compressed video stream. If that is not possible, the sender shall initiate a session re-negotiation to remove the still image media line.

The "imageattr" attribute as specified in IETF RFC 6236 [76] shall be supported and shall indicate the image size.

6.3 Session control procedures

Addition and removal of media components shall be performed based on the SDP-based offer-answer model as specified in RFC 3264 [58].

During session renegotiation for adding or removing media components, the SDP offerer should continue to use the same media (m=) line(s) from the previously negotiated SDP for the media components that are not being added or removed.

An MTSI client in terminal may support multiple media components including media components of the same media type. An MTSI client in terminal may support adding one or more media components to an on-going session which already contains a media component of the same media type. If an MTSI client in terminal needs to have multiple media components of the same media type in a single MTSI session, then the MTSI client in terminal should use the SDP content attributes as defined in [81] for identifying different media components.

SDP examples for adding a second video stream to an ongoing video telephony session and removing a video stream from an ongoing video telephony session are given in Annex A.11.

The content attribute can be used in combination with the group attributes defined in RFC 3388 [48] and also in combination with the synchronization attributes defined in Clause 6.2.6, for example to identify two (or more) media components are related to each other and if synchronization is needed.

7 Data transport

7.1 General

MTSI clients shall support an IP-based network interface for the transport of session control and media data. Control-plane signalling is sent using SIP; see TS 24.229 [7] for further details. Real-time user plane media data is sent over RTP/UDP/IP. Real-time interaction is using data channels over SCTP/DTLS/UDP/IP. Non-real-time media may use other transport protocols, for example UDP/IP or TCP/IP. An overview of the user plane protocol stack can be found in figure 4.3 of the present document.

7.2 RTP profiles

MTSI clients shall transport speech, video and real-time text using RTP (RFC 3550 [9]) over UDP (RFC 0768 [39]). The following profiles of RTP shall be supported for all media types:

- RTP Profile for Audio and Video Conferences with Minimal Control (RFC 3551 [10]), also called RTP/AVP;

The following profiles of RTP shall be supported for video and should be supported for all other media types:

- Extended RTP Profile for RTCP-based Feedback (RTP/AVPF) (RFC 4585 [40]), also called RTP/AVPF.

The support of AVPF requires an MTSI client in terminal to implement the RTCP transmission rules, the signalling mechanism for SDP and the feedback messages explicitly mentioned in the present document.

For a given RTP based media stream, the MTSI client in terminal shall use the same port number for sending and receiving RTP packets. This facilitates interworking with fixed/broadband access. However, the MTSI client shall accept RTP packets that are not received from the same remote port where RTP packets are sent by the MTSI client.

7.3 RTCP usage

7.3.1 General

The RTP implementation shall include an RTCP implementation.

For a given RTP based media stream, the MTSI client in terminal shall use the same port number for sending and receiving RTCP packets. This facilitates interworking with fixed/broadband access. However, the MTSI client shall accept RTCP packets that are not received from the same remote port where RTCP packets are sent by the MTSI client.

The bandwidth for RTCP traffic shall be described using the "RS" and "RR" SDP bandwidth modifiers at media level, as specified by RFC 3556 [42]. Therefore, an MTSI client shall include the "b=RS:" and "b=RR:" fields in SDP, and shall be able to interpret them. There shall be an upper limit on the allowed RTCP bandwidth for each RTP session signalled by the MTSI client. This limit is defined as follows:

- 8 000 bps for the RS field (at media level);
- 6 000 bps for the RR field (at media level).

The RS and RR values included in the SDP answer should be treated as the negotiated values for the session and should be used to calculate the total RTCP bandwidth for all terminals in the session.

If the session described in the SDP is a point-to-point speech only session, the MTSI client may request the deactivation of RTCP by setting its RTCP bandwidth modifiers to zero.

If a MTSI client receives SDP bandwidth modifiers for RTCP equal to zero from the originating MTSI client, it should reply (via the SIP protocol) by setting its RTCP bandwidth using SDP bandwidth modifiers with values equal to zero.

RTCP packets should be sent for all types of multimedia sessions to enable synchronization with other RTP transported media, remote end-point aliveness information, monitoring of the transmission quality, and carriage of feedback messages such as TMMBR for video and RTCP APP for speech. The RR value should be set greater than zero to enable RTCP packets to be sent when media is put on hold and during active RTP media transmission, including real-time text sessions which may have infrequent RTP media transmissions.

Point-to-point speech only sessions may not require the above functionalities and may therefore turn off RTCP by setting the SDP bandwidth modifiers (RR and RS) to zero. When RTCP is turned off (for point-to-point speech only sessions) and the media is put on hold, the MTSI client should re-negotiate the RTCP bandwidth with the SDP bandwidth modifier RR value set greater than zero, and send RTCP packets (i.e., Receiver Reports) to the other end. This allows the remote end to detect link aliveness during hold. When media is resumed, the resuming MTSI client should request to turn off the RTCP sending again through a re-negotiation of the RTCP bandwidth with SDP bandwidth modifiers equal to zero.

When RTCP is turned off (for point-to-point speech only sessions) and if sending of an additional associated RTP stream becomes required and both RTP streams need to be synchronized, or if transport feedback due to lack of end-to-end QoS guarantees is needed, a MTSI client should re-negotiate the bandwidth for RTCP by sending an SDP with the RR bandwidth modifier greater than zero. Setting the RR bandwidth modifier greater than zero allows sending of RTCP Receiver Reports even when the session is put on hold and neither terminal is actively sending RTP media.

NOTE: Deactivating RTCP will disable the adaptation mechanism for speech defined in clause 10.2.

7.3.2 Speech

MTSI clients in terminals offering speech should support AVPF (RFC 4585 [40]). When allocating RTCP bandwidth, it is recommended to allocate RTCP bandwidth and set the values for the "b=RR:" and the "b=RS:" parameters such that a good compromise between the RTCP reporting needs for the application and bandwidth utilization is achieved, see also Annex A.6. The value of "tr-int" should be set to zero or not transmitted at all (in which case the default "tr-int" value of zero will be assumed) when Reduced-Size RTCP (see clause 7.3.6) is not used.

For speech sessions it is beneficial to keep the size of RTCP packets as small as possible in order to reduce the potential disruption of RTCP onto the RTP stream in bandwidth-limited channels. RTCP packet sizes can be minimized by using Reduced-Size RTCP packets or using the parts of RTCP compound packets (according to RFC 3550 [9]) which are required by the application. RTCP compound packet sizes should be at most as large as 1 time and, at the same time, shall be at most as large as 4 times the size of the RTP packets (including UDP/IP headers) corresponding to the highest bit rate of the speech codec modes used in the session. Reduced-Size RTCP and semi-compound RTCP packet sizes should be at most as large as 1 time and, at the same time, shall be at most as large as 2 times the size of the RTP packets (including UDP/IP headers) corresponding to the highest bit rate of the speech codec modes used in the session.

An MTSI client using ECN for speech in RTP sessions may support the RTCP AVPF ECN feedback message and the RTCP XR ECN summary report [84]. If the MTSI client supports the RTCP AVPF ECN feedback message then the MTSI client shall also support the RTCP XR ECN summary report.

NOTE 1: This can improve the interworking with non-MTSI ECN peers.

When an MTSI client that has negotiated the use of ECN and then receives RTP packets with ECN-CE marks, the MTSI client shall send application specific adaptation requests (RTP CMR [28] or RTCP-APP CMR, as defined in Subclause 10.2.1.5) and shall not send RTCP AVPF ECN feedback messages, even if RTCP AVPF ECN feedback messages were negotiated.

NOTE 2: RTP CMR is mandated to be supported by any AMR or AMR-WB implementation using the RTP profile [28].

When an MTSI client in terminal that has negotiated the use of ECN for speech and RTCP AVPF ECN feedback messages receives both application specific requests and RTCP AVPF ECN feedback messages, the MTSI client should follow the application specific requests for performing media bit rate adaptation.

When an MTSI client in terminal that has negotiated the use of ECN for speech and RTCP XR ECN summary reports receives an RTCP XR ECN summary report, the MTSI client should use the RTCP XR ECN summary report as specified in [84]. If the MTSI client received and acted upon a recent application specific adaptation request, then the MTSI client shall not perform any additional rate adaptation based on the received RTCP XR ECN summary report.

If ANBR (see clause 10.7) is available to the MTSI client in terminal, it should use this information when performing media bitrate adaptation. In addition, a media receiving MTSI client in terminal may send RTCP-APP or RTP CMR messages for speech rate adaptation based on adaptation decisions, including ANBR information.

For speech, RTCP APP packets are used for adaptation (see clause 10.2). If the MTSI client determines that RTCP APP cannot be used or does not work then the MTSI client may use CMR in the AMR RTP payload [28] inband CMR or other RTCP mechanisms for adaptation.

An MTSI client that requests mode adaptation shall use the CMR in the AMR/AMR-WB RTP payload [28] when using the AMR or the AMR-WB codec or in the EVS payload [125] when using the EVS codec, respectively, when:

- the RTCP bandwidth is set to zero,
- the MTSI client detected that the remote end-point does not respond to adaptation requests sent with RTCP APP during the session, or
- the support for RTCP APP was not negotiated for the session.

If RTCP-APP was negotiated, an MTSI client that requests mode adaptation for EVS shall use RTCP-APP when the CMR in the EVS RTP payload has been disabled for the session.

NOTE 3: It is not possible to send adaptation requests if both CMR in the EVS RTP payload has been disabled and if RTCP-APP is not negotiated for the session.

An MTSI client using AMR or AMR-WB that requests mode adaptation when no MTSI feature tag was received (see clause 5.2 of [57]) may use the CMR in the AMR/AMR-WB RTP payload, [28], when AMR or AMR-WB is used and may use the CMR in the EVS RTP payload, [125], when EVS is used, respectively. If ECN-triggered adaptation is used and an MTSI client requests mode adaptation when no MTSI feature tag was received it should use the CMR in the AMR RTP payload, [28].

NOTE 4: Other procedures by which the MTSI client determines that RTCP APP cannot be used or does not work is implementation specific.

If ECN-triggered adaptation is used with AVP then the RTCP APP signalling could be too slow and CMR in the AMR RTP payload [28] should be used for faster feedback.

An MTSI client that requests mode adaptation in combination with other codec control requests (as defined in clause 10.2.1) shall use RTCP APP.

An MTSI client that requests rate adaptation for unidirectional streams shall use RTCP-based adaptation signaling (RTCP APP or RTCP SR/RR) since CMR in the AMR RTP payload, [28] is not usable for unidirectional streams.

7.3.3 Video

MTSI clients offering video shall support AVPF (RFC 4585 [40]). The behaviour can be controlled by allocating enough RTCP bandwidth using "b=RR:" and "b=RS:" (see clause 7.3.1) and setting the value of "trr-int".

MTSI clients offering video shall support transmission and reception of AVPF NACK messages, as an indication of non-received media packets. MTSI terminals offering video shall also support transmission and reception of AVPF Picture Loss Indication (PLI). The actions of an MTSI client receiving NACK or PLI to improve the situation for the MTSI client that sent NACK or PLI is defined in clause 9.3. Note that by setting the bitmask of following lost packets (BLP) the frequency of transmitting NACK can be reduced, but the repairing action by the MTSI client receiving the message can be delayed correspondingly.

The Temporary Maximum Media Bit-rate Request (TMMBR) and Temporary Maximum Media Bit-rate Notification (TMMBN) messages of Codec-Control Messages (CCM) [43] shall be supported by MTSI clients in terminals supporting video. The TMMBR notification messages along with RTCP sender reports and receiver reports are used for dynamic video rate adaptation. See clause 10.3 for usage and Annexes B and C for examples of bitrate adaptation.

MTSI clients supporting video shall support Full Intra Request (FIR) of CCM [43]. A sender should ignore FIR messages that arrive within Response Wait Time (RWT) duration after responding to a previous FIR message. Response Wait Time (RWT) is defined as RTP-level round-trip time, estimated by RTCP or some other means, plus twice the frame duration.

MTSI clients in terminals shall not use SIP INFO message, as specified in [96], for video picture fast update.

The usage of the AVPF and CCM feedback messages is negotiated in SDP offer/answer, see Clause 6.2.3.2. Any AVPF or CCM feedback messages that have not been agreed in the SDP offer/answer negotiation shall not be used in the session, [40].

An MTSI client using ECN for video in RTP sessions may support the RTCP AVPF ECN feedback message and the RTCP XR ECN summary report [84]. If the MTSI client supports the RTCP AVPF ECN feedback message then the MTSI client shall also support the RTCP XR ECN summary report.

NOTE: This can improve the interworking with non-MTSI ECN-capable peers.

When an MTSI client that has negotiated the use of ECN and TMMBR receives RTP packets with ECN-CE marks, the MTSI client shall send application specific adaptation requests (TMMBR) and shall not send RTCP AVPF ECN feedback messages, even if RTCP AVPF ECN feedback messages were negotiated in addition to TMMBR.

When an MTSI client that has negotiated the use of ECN for video and RTCP AVPF ECN feedback messages receives both application specific requests and RTCP AVPF ECN feedback messages, the MTSI client should follow the application specific requests for performing media bit rate adaptation.

When an MTSI client that has negotiated the use of ECN for video and RTCP XR ECN summary reports receives an RTCP XR ECN summary report, the MTSI client should use the RTCP XR ECN summary report as specified in [84]. If the MTSI client received and acted upon a recent application specific adaptation request, then the MTSI client shall not perform any additional rate adaptation based on the received RTCP XR ECN summary report.

If ANBR (see clause 10.7) information is available to the MTSI client in terminal, it should use this information when performing media bitrate adaptation. In addition, a media receiving MTSI client in terminal may send RTCP feedback messages (e.g., TMMBR, TMMBN messages of CCM, etc.) for video rate adaptation based on adaptation decisions, including ANBR information.

7.3.4 Real-time text

For real-time text, RTCP reporting should be used according to general recommendations for RTCP.

7.3.5 Void

7.3.6 Reduced-Size RTCP

MTSI clients should support the use of Reduced-Size RTCP reports [87]. A Reduced-Size RTCP packet is an RTCP packet that does not follow the sending rules outlined in RFC 3550 [9] in the aspect that it does not necessarily contain the mandated RR/SR report blocks and SDES CNAME items.

As specified in RFC5506 [87], a client that support Reduced-Size RTCP shall also support AVPF, see clause 7.2 An SDP offer to use Reduced-Size RTCP shall also offer using AVPF.

When Reduced-Size RTCP is used, the following requirements apply on the RTCP receiver:

- The RTCP receiver shall be capable of parsing and decoding report blocks of the RTCP packet correctly even though some of the items mandated by RFC3550 [9] are missing.
- An SDP attribute "a=rtcp-rsize" is used to enable Reduced-Size RTCP. A receiver that accepts the use of Reduced-Size RTCP shall include the attribute in the SDP answer. If this attribute is not set in offer/answer, then Reduced-Size RTCP shall not be used in any direction.

When Reduced-Size RTCP is used, an RTCP sender transmitting Reduced-Size RTCP packets shall follow the requirements listed below:

- AVPF early or immediate mode shall be used according to RFC4585 [40].
- The "a=rtcp-rsize" attribute shall be included in the SDP offer, see Annex A.9a.
- Reduced-Size RTCP packets should be used for transmission of adaptation feedback messages, for example APP packets as defined in Clause 10.2 and TMMBR as defined in Clause 10.3. When regular feedback packets are transmitted, the individual packets that would belong to a compound RTCP packet shall be transmitted in a serial fashion, although adaptation feedback packets shall take precedence.
- Two or more RTCP packets should be stacked together, within the limits allowed by the maximum size of Reduced-Size RTCP packets (see clause 7.3.2) (i.e., to form a semi-compound RTCP packet which is smaller than a compound RTCP packet). The RTCP sender should not send Reduced-Size RTCP packets that are larger than the regularly scheduled compound RTCP packets.
- Compound RTCP packets with an SR/RR report block and CNAME SDES item should be transmitted on a regular basis as outlined in RFC 3550 [9] and RFC 4585 [40]. In order to control the allocation of bandwidth between Reduced-Size RTCP and compound RTCP, the AVPF "tr-int" parameter should be used to set the minimum report interval for compound RTCP packets.
- The first transmitted RTCP packet shall be a compound RTCP packet as defined in RFC3550 [9] without the size restrictions defined in clause 7.3.2.

The application should verify that the Reduced-Size RTCP packets are successfully received by the other end-point. Verification can be done by implicit means, for instance the RTCP sender that sends an adaptation feedback requests is expected to detect some kind of a response to the requests in the media stream. If verification fails then the RTCP sender shall switch to the use of compound RTCP packets according to the rules outlined in RFC3550 [9].

Examples of SDP negotiation for Reduced-Size RTCP given in Clause A.9a.

7.3.7 Video Region-of-Interest (ROI) Signaling

Video Region-of-Interest (ROI) consists of signalling the currently requested region-of-interest (ROI) of the video on the receiver side to the sender for appropriate encoding and transmission.

Video ROI is composed of three modes of signalling from an MTSI receiver to an MTSI sender in order to request a desired region of interest, and an MTSI client supporting ROI shall support at least one of these modes:

- 'FECC' mode, in which the MTSI client uses the FECC protocol based on ITU-T H.281 over H.224 [135]-[138] to signal ROI information as a sequence of 'Pan', 'Tilt', 'Zoom' and 'Focus' (PTZF) commands.
- 'Arbitrary ROI' mode, in which the MTSI receiver determines a specific ROI and signals this ROI to the MTSI sender.
- 'Pre-defined ROI' mode, in which the MTSI receiver selects one of the ROIs pre-determined by the MTSI sender and signals this ROI to the MTSI sender. In this mode, the MTSI receiver obtains the set of pre-defined ROIs from the MTSI sender during the SDP capability negotiation.

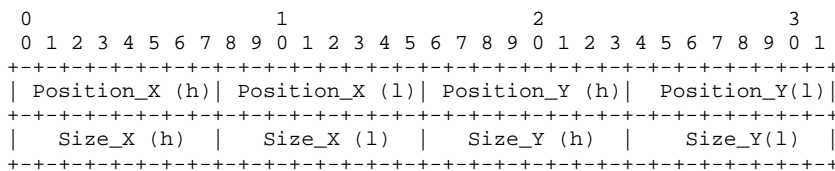
In the FECC mode, the ROI information shall be signaled by the MTSI client via RTP packets that carry H.224 frames using the stack IP/UDP/RTP/H.224/H.281. FECC is internal to the H.224 frame and is identified by the client ID field of the H.224 packet. The zooming to a particular region of interest is enabled by the H.281 protocol that supports the 4 basic camera movements "PTZF" (Pan, Tilt, Zoom, and Focus). In case of a fixed camera without pan/tilt capabilities, the pan command should be mapped to left/right movements/translations and tilt command should be mapped to up/down movements/translations over the 2D image plane. As such, a combination of PTZ commands can still allow for zooming into an arbitrary ROI.

The signalling of ‘Arbitrary ROI’ and ‘Pre-defined ROI’ requests uses RTCP feedback messages as specified in IETF RFC 4585 [40]. The RTCP feedback message is identified by PT (payload type) = PSFB (206) which refers to payload-specific feedback message. FMT (feedback message type) shall be set to the value ‘9’ for ROI feedback messages. The IANA registration information for the FMT value for ROI is provided in Annex R.1. The RTCP feedback method may involve signaling of ROI information in both of the immediate feedback and early RTCP modes.

The FCI (feedback control information) format for ROI shall be as follows. The FCI shall contain exactly one ROI. The ROI information is composed of the following parameters:

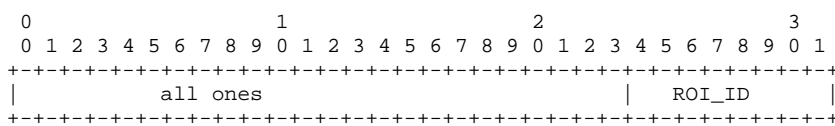
- Position_X - specifies the x-coordinate for the upper left corner of the ROI area covered in the original content (i.e., uncompressed captured content) in units of pixels
- Position_Y - specifies the y-coordinate for the upper left corner of the ROI area covered in the original content in units of pixels
- Size_X - specifies the horizontal size of the ROI area covered in the original content in units of pixels
- Size_Y - specifies the vertical size of the ROI area covered in the original content in units of pixels
- ROI_ID – identifies the pre-defined ROI selected by the MTSI receiver

For ‘Arbitrary ROI’ requests, the RTCP feedback message for ROI shall contain the parameters Position_X, Position_Y, Size_X and Size_Y. The values for the each of the parameters Position_X, Position_Y, Size_X and Size_Y shall each be indicated using two bytes. The MTSI sender shall ignore ROI requests describing regions outside the original video. The FCI for the RTCP feedback message for ‘Arbitrary ROI’ shall follow the following format:



For each two-byte indication of the Position_X, Position_Y, Size_X and Size_Y parameters, the high byte (indicated by ‘h’) above) shall be followed by the low byte (indicated by ‘l’) above), where the low byte holds the least significant bits.

For ‘Pre-defined ROI’ requests, the RTCP feedback message for ROI shall contain the ROI_ID parameter. The value of ROI_ID shall be acquired from the "a=predefined_ROI" attributes that are indicated in the SDP offer-answer negotiation (see clause 6.2.3.4 for the related SDP-based procedures). The value for the ROI_ID parameter shall be indicated using one byte. The FCI for the RTCP feedback message for ‘Pre-defined ROI’ shall follow the following format:



If ‘Arbitrary ROI’ and ‘Pre-defined ROI’ are both successfully negotiated, then the RTCP feedback message from the MTSI receiver shall conform to one of the two message formats specified above for ‘Arbitrary ROI’ or ‘Pre-defined ROI’, respectively. The MTSI sender should distinguish between the two RTCP feedback message formats by parsing the first 24 bits, which is uniquely set to all ones in case of ‘Pre-defined ROI’ requests.

The semantics of the ROI feedback messages is independent of the payload type.

‘Sent ROI’ involves signalling from the MTSI sender to the MTSI receiver and this helps the MTSI receiver to know the actually sent ROI corresponding to the video transmitted by the MTSI sender, i.e., which may or may not agree with the ROI requested by the MTSI receiver, but shall contain it so that the end user is still able to see the desired ROI. When ‘Sent ROI’ is successfully negotiated, it shall be signalled by the MTSI sender.

If the sent ROI corresponds to an arbitrary ROI (indicated via the URN urn:3gpp:roi-sent in the SDP negotiaton, see clause 6.2.3.4), the signalling of the ROI shall use RTP header extensions as specified in IETF 5285 [95] and shall carry the Position_X, Position_Y, Size_X and Size_Y parameters corresponding to the actually sent ROI. The one-byte form of the header should be used. The values for the parameters Position_X, Position_Y, Size_X and Size_Y shall each be indicated using two bytes, with the following format:


```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|  ID  | len=7 | Position_X (h)| Position_X (l)| Position_Y (h)|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Position_Y (l)| Size_X (h) | Size_X (l) | Size_Y (h) |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Size_Y (l) | zero padding |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

The 4-bit ID is the local identifier as defined in [95]. The length field takes the value 7 to indicate that 8 bytes follow. For each two-byte indication of the Position_X, Position_Y, Size_X and Size_Y parameters, the high byte (indicated by ‘h’ above) shall be followed by the low byte (indicated by ‘l’ above), where the low byte holds the least significant bits.

If the sent ROI corresponds to one of the pre-defined ROIs (indicated via the URN urn:3gpp:predefined-roi-sent in the SDP negotiation, see clause 6.2.3.4), then the signalling of the ROI shall again use the RTP header extensions and shall carry the ROI_ID parameter corresponding to the actually sent pre-defined ROI. The one-byte form of the header should be used. The value for the ROI_ID parameter shall be indicated using one byte, with the following format:

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|  ID  | len=0 | ROI_ID | zero padding |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

In this case, the length field takes the value 0 to indicate that only a single byte follows.

‘Arbitrary ROI’ and ‘Pre-defined ROI’ may be supported bi-directionally or uni-directionally depending on how clients negotiate to support the feature during SDP capability negotiations. For terminals with asymmetric capability (e.g. the ability to process ROI information but not detect/signal ROI information), the sendonly and recvonly attributes may be used. Terminals should express their capability in each direction sufficiently clearly such that signals are only sent in each direction to the extent that they both express useful information and can be processed by the recipient.

‘Arbitrary ROI’ and ‘Pre-defined ROI’ support may be offered at the same time, or only one of them may be offered. When both capabilities are successfully negotiated by the MTSI sender and receiver, it is the MTSI receiver’s decision to request an arbitrary ROI or one of the pre-defined ROIs at a given time. When pre-defined ROIs are offered by the MTSI sender, it is also the responsibility of the MTSI sender to detect and track any movements of the ROI, e.g., the ROI could be a moving car, or moving person, etc. and refine the content encoding accordingly.

The presence of ROI signalling should not impact the negotiated resolutions (based on SDP imageattr attribute) between the sending and receiving terminals. The only difference is that the sending terminal should encode only the ROI with the negotiated resolution rather than the whole captured frame, and this would lead to a higher overall resolution and better user experience than having the receiving terminal zoom in on the ROI and crop out the rest of the frame.

The ROI information parameters exchanged via the RTP/RTCP signalling defined above are independent of the negotiated video resolution for the encoded content. Instead, the ROI information parameters defined above take as reference the original video content, i.e., uncompressed captured video content. Therefore, no modifications or remappings of ROI parameters are necessary during any transcoding that results in changes in video resolution or during potential dynamic adaptations of encoded video resolution at the sender.

An MTSI sender may have to handle multiple simultaneously received ROI requests. The encoder at the MTSI sender may consider the multiple ROI requests to determine a proximity ROI that is a larger area that contains all the requested ROIs, and encode the transmitted video stream according to the proximity ROI. The encoder may iteratively adjust the proximity ROI based on the interactive additional ROI requests received from the remote clients. These additional ROI requests can be in the form of PTZF commands (using the FECC protocol) corresponding to the desired translation of the proximity ROI each MTSI receiver wishes the MTSI sender to make. Alternatively, the MTSI sender may offer the set of candidate proximity ROIs to the MTSI receivers using the pre-defined ROI signalling framework, and collect responses from the MTSI receivers to determine their preferred proximity ROIs. By considering these additional ROI requests, the MTSI sender can make a better decision on the proximity ROI to fulfil the requests of as many MTSI receivers as possible.

When the MTSI sender is not able to derive a proximity ROI from the received concurrent ROI requests, the MTSI sender should transmit the full-size view of the video to those users whose ROI requests cannot be satisfied. In case of ‘Pre-defined ROI’, this can be achieved by including the full-size view of the video in the list of pre-defined ROIs. Then, the MTSI sender can transmit the full-size view of the video and also signal the corresponding ROI_ID (via the

RTP header extension using ‘Sent ROI’) if a specific pre-defined ROI request cannot be satisfied. In case of ‘Arbitrary ROI’, the MTSI sender can transmit the full-size view of the video and also signal the corresponding coordinates of the full-size view (via the RTP header extension using ‘Sent ROI’) during times when an ROI request cannot be satisfied.

7.3.8 Delay Budget Information (DBI) Signaling

RAN delay budget reporting is specified in TS 36.331 [160] for E-UTRA and TS 38.331 [163] for NR while the use of RAN delay budget reporting is specified for coverage enhancements only in E-UTRA.. RAN delay budget reporting through the use of RRC signalling to eNB / gNB allows UEs to locally adjust air interface delay. Based on the reported delay budget information, a good coverage UE on the receiving end (i.e., the UE that contains the MTSI receiver) can reduce its air interface delay, e.g., by turning off CDRX or via other means. This additional delay budget can then be made available for the sending UE (i.e., the UE that contains the MTSI sender), and can be quite beneficial for the sending UE when it suffers from poor coverage. When the sending UE is in bad coverage, it would request the additional delay from its local eNB / gNB, and if granted, it would utilize the additional delay budget to improve the reliability of its uplink transmissions in order to reduce packet loss, e.g., via suitable repetition or retransmission mechanisms, and thereby improve end-to-end delay and quality performance.

While RAN-level delay budget reporting as defined in TS 36.331 [160] and TS 38.331 [163] allows UEs (i.e., MTSI sender and MTSI receiver) to locally adjust air interface delay, such a mechanism does not provide coordination between the UEs on an end-to-end basis. To alleviate this issue, this clause defines RTCP signalling to realize the following capabilities on signalling of delay budget information (DBI) across UEs: (i) an MTSI receiver can indicate available delay budget to an MTSI sender, and (ii) an MTSI sender can explicitly request delay budget from an MTSI receiver.

More specifically, the RTCP-based signalling of DBI is composed of a dedicated RTCP feedback (FB) message type to carry available additional delay budget during the RTP streaming of media, signalled from the MTSI receiver to the MTSI sender. In addition, the defined RTCP feedback message type may also be used to carry requested additional delay budget during the RTP streaming of media, signalled from the MTSI sender to the MTSI receiver.

A corresponding dedicated SDP parameter on the RTCP-based ability to signal available or requested additional delay budget during the IMS/SIP based capability negotiations is also defined, as described in sub-clause 6.2.8.

Such RTCP-based signaling of DBI can also be used by an MTSI receiver to indicate delay budget availability created via other means such as jitter buffer size adaptation as mentioned in clause 8.2.1.

The signalling of available or requested additional delay budget information (DBI) shall use RTCP feedback messages as specified in IETF RFC 4585 [40]. The RTCP feedback message is identified by PT (payload type) = RTPFB (205) which refers to RTP-specific feedback message. FMT (feedback message type) shall be set to the value '10' for delay budget information (DBI). The RTCP feedback method may involve signalling of available or requested additional delay budget in both of the immediate feedback and early RTCP modes.

As such, the RTCP feedback message shall be sent from the MTSI receiver to the MTSI sender to convey to the sender the available additional delay budget from the perspective of the receiver. The recipient UE of the RTCP feedback message (i.e., the UE containing the MTSI sender) may then use this information in determining how much delay budget it may request from its eNB / gNB over the RAN interface, e.g. by using RRC signalling based on *UEAssistanceInformation* as defined in TS 36.331 [160] and TS 38.331 [163].

The FCI (feedback control information) format shall be as follows. The FCI shall contain exactly one instance of the available additional delay budget information, composed of the following parameters:

- Available additional delay budget *delay* - specified in milliseconds (16 bits)
- Sign 's' for the additional delay budget *delay* and whether this is positive or negative– specified as a Boolean (1 bit)
- Query 'q' for additional delay budget – specified as a Boolean (1 bit)

The sign value, 's' may be positive, indicated by '1' or negative, indicated by '0'. Essentially, when the additional delay parameter takes on a positive value, the UE indicates that there is additional delay budget available. In case the additional delay parameter takes on a negative value, the UE indicates that the available delay budget has been reduced. A sequence of RTCP feedback messages may be sent by the UE to report on the additional delay budget availability in increments.

When the MTSI receiver sends RTCP feedback messages indicating the available delay budget for the received RTP stream, the query parameter shall be set to '0'. When the MTSI sender sends RTCP feedback messages indicating the requested delay budget for the RTP stream sent from the MTSI sender to the MTSI receiver, the query parameter shall be set to '1'. In this case, the value of `delay` indicates the additional delay budget requested by the sender of the RTCP feedback message (i.e., the MTSI sender) for the RTP stream sent from the MTSI sender to the MTSI receiver.

The FCI for the proposed RTCP feedback message shall follow the following format where (i) 's' stands for the single-bit message on the sign of the additional delay parameter and (ii) 'q' stands for the single-bit message on query:

```

0          1          2          3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|          delay          |s|q|      zero padding      |
+-----+-----+-----+-----+-----+-----+-----+

```

The high byte of `delay` shall be followed by the low byte, where the low byte holds the least significant bits.

Annex V presents example signalling flows on RAN delay budget reporting usage for voice in MTSI with and without DBI signalling.

An MTSI receiver shall not indicate available delay budget to an MTSI sender via DBI signalling more frequently than once every `T_DBI` seconds, provided that the necessary amount of RTCP bandwidth is available. If an MTSI receiver indicates available delay budget to an MTSI sender via DBI signalling, this shall mean that the indicated delay budget amount is available to the MTSI sender for at least the duration of `T_DBI` seconds. An MTSI sender shall not request delay budget from an MTSI receiver via DBI signalling more frequently than once every `T_DBI` seconds. `T_DBI` shall be set to a value between 1 – 3 seconds.

NOTE: The requirement on how to set the exact value of `T_DBI` is FFS.

Timing-wise, it is possible that DBI signalling may happen concurrently or asynchronously between the MTSI sender and MTSI receiver, i.e., the MTSI receiver may indicate available delay budget to the MTSI sender, while the MTSI sender may request delay budget from an MTSI receiver.

If the MTSI sender receives available delay budget information from an MTSI receiver via DBI signaling, this delay budget is available for its uplink over the duration of at least `T_DBI` seconds. Thus, if an MTSI receiver has already indicated available delay budget to the MTSI sender via DBI signalling, reception of a DBI request from the MTSI sender during any time within the time window of `T_DBI` seconds shall not trigger any further DBI signalling from the MTSI receiver to the MTSI sender on the available delay budget at any time sooner than `T_DBI` seconds following the last indication of the available delay budget.

Once the period of `T_DBI` seconds following the last indication of the available delay budget is over, if the available delay budget has changed, the MTSI receiver shall inform the MTSI sender on the new delay budget availability (as a relative value as explained above) using DBI signalling. If the MTSI sender does not receive any new DBI signalling on the available delay budget from the MTSI receiver after the `T_DBI` second period is over, it shall mean the continued availability of the same amount of delay budget indicated to the MTSI sender via the latest DBI signalling.

Likewise, if the MTSI sender no longer needs the additional delay budget it has requested earlier or has a delay budget request that is different from what it had requested earlier, it shall inform the MTSI receiver about the new delay budget request (as a relative value as explained above) via DBI signalling. If the MTSI receiver does not receive any new DBI signalling on the requested delay budget from the MTSI sender after the `T_DBI` second period is over, this shall mean that the MTSI sender is still requesting the same amount of delay budget indicated to the MTSI receiver via the latest DBI signalling.

It should be noted that the *delayBudgetReportingProhibitTimer* parameter for RAN delay budget reporting as defined in TS 36.331 [160] for E-UTRA and TS 38.331 [163] for NR may take any of the values among 0, 0.4, 0.8, 1.6, 3, 6, 12 and 30 seconds, as set by the local eNB / gNB. Hence, if an MTSI receiver is to provide additional delay budget by locally adjusting air interface delay via RAN delay budget reporting (as supposed to adjusting its jitter buffer size, which can be set independently from the *delayBudgetReportingProhibitTimer* parameter), the frequency of its signalling to eNB / gNB is subject to the *delayBudgetReportingProhibitTimer* parameter. Likewise, when an MTSI sender requests delay budget from its local eNB / gNB via RAN delay budget reporting, the frequency of this signalling is subject to the *delayBudgetReportingProhibitTimer* parameter. Therefore, it should be observed that end-to-end delay adaptation through the use of RAN delay budget reporting and DBI signalling may be limited when the eNB / gNB sets the *delayBudgetReportingProhibitTimer* parameter to a large value. In particular, if *delayBudgetReportingProhibitTimer* is set to a value larger than `T_DBI` seconds, then DBI signaling cannot be used in conjunction with RAN delay budget reporting.

Provided that the *delayBudgetReportingProhibitTimer* configurations over the uplink and downlink access networks of the respective MTSI sender and MTSI receiver both do not exceed 3 seconds, T_DBI should be set to a value greater than or equal to the maximum of the *delayBudgetReportingProhibitTimer* configurations over uplink and downlink access networks. In case an MTSI receiver adjusts its jitter buffer size and does not use RAN delay budget reporting, *delayBudgetReportingProhibitTimer* parameter for downlink may be considered to be set to zero as part of this recommendation. Typical *delayBudgetReportingProhibitTimer* configurations will be in the values of 0, 0.4, 0.8, 1.6 seconds, so setting T_DBI to 1.6 seconds is recommended to operate with typical *delayBudgetReportingProhibitTimer* configurations.

When transcoding is present on the media path between the MTSI sender and MTSI receiver in the packet-switched domain, the end-to-end delay and quality performance enhancements realized by DBI signalling are still applicable as long as the media gateway in between passes the RTCP feedback messages carrying DBI. There may be a possible reduction however on the end-to-end performance gains, due to the additional delays incurred from transcoding.

When transcoding is present on the media path between an MTSI sender in the packet-switched domain and a media receiver in the circuit-switched domain, the end-to-end delay and quality performance enhancements realized by DBI signalling may still be applicable if the media gateway is able to offer additional delay budget, e.g., by extending its jitter buffer size, while also considering the fixed delay over the circuit-switched domain. In this case, the media gateway may receive delay budget request from the MTSI sender via DBI signalling, and the media gateway may further inform the MTSI sender about available delay budget via DBI signalling (note that no DBI signalling happens in the circuit switched domain).

In case of multiparty conferencing, DBI signalling may also be useful to improve end-to-end delay and quality performance of the RTP streams exchanged between the clients and conferencing server. In particular, an MSMTSI client (as defined in Annex S) and an MSMTSI MRF (as defined in Annex S) may negotiate DBI signalling using the SDP based procedures described in sub-clause 6.2.8. An MSMTSI client may then use DBI signalling to indicate available additional delay budget for the RTP streams received from the MSMTSI MRF and also request additional delay budget for the RTP streams it sends to the MSMTSI MRF. Likewise, an MSMTSI MRF may then use DBI signalling to indicate available additional delay budget for the RTP streams received from the MSMTSI client and also request additional delay budget for the RTP streams it sends to the MSMTSI client.

7.4 RTP payload formats for MTSI clients

7.4.1 General

This clause specifies RTP payload formats for MTSI clients, except for MTSI media gateways that is specified in clause 12.3.2, for all codecs supported by MTSI in clause 5.2. Note that each RTP payload format also specifies media type signalling for usage in SDP.

7.4.2 Speech

When the AMR codec is selected in the SDP offer-answer negotiation the AMR payload format [28] shall be used between RTP termination points.

When the AMR-WB is selected in the SDP offer-answer negotiation the AMR-WB payload format [28] shall be used between RTP termination points.

NOTE 1: It may happen that EVS AMR-WB IO encoded speech is transported using the AMR-WB payload format between an EVS-capable MTSI client and a legacy (not EVS capable) MTSI client. This may also happen after SRVCC (see Clause 12.3.4) when an EVS-capable MTSI client sends EVS AMR-WB IO encoded speech in EVS payload format to the ATGW and the ATGW then re-packetizes the EVS AMR-WB IO packet into AMR-WB payload format without performing transcoding of the media.

When the EVS codec is selected in the SDP offer-answer negotiation the EVS payload format [125] shall be used between RTP termination points.

NOTE 2: After SRVCC when a CS UE (not EVS capable) sends AMR-WB encoded speech to the ATGW, it may happen that the ATGW then re-packetizes this AMR-WB packet into the EVS payload format without performing transcoding of the media, see clause 12.3.4.

In case of ambiguity the present specification shall take precedence over RFC 4867 [28].

MTSI clients (except MTSI MGW) shall support both the bandwidth-efficient and the octet-aligned payload format of the AMR/AMR-WB payload format [28]. The bandwidth-efficient payload format shall be preferred over the octet-aligned payload format.

When sending AMR or AMR-WB encoded media, the RTP Marker Bit shall be set according to Section 4.1 of the AMR/AMR-WB payload format [28]. When sending EVS encoded media, the RTP Marker Bit shall be set as described in the EVS payload format [125].

The MTSI clients (except MTSI MGW) should use the SDP parameters defined in table 7.1 for the session. For all access technologies, and for normal operating conditions, the MTSI client should encapsulate the number of non-redundant (a.k.a. primary) speech frames in the RTP packets that corresponds to the ptime value received in SDP from the other MTSI client, or if no ptime value has been received then according to "Recommended encapsulation" defined in table 7.1. The MTSI client may encapsulate more non-redundant speech frames in the RTP packet but shall not encapsulate more than 4 non-redundant speech frames in the RTP packets. The MTSI client may encapsulate any number of redundant speech frames in an RTP packet but the length of an RTP packet, measured in ms, shall never exceed the maxptime value.

NOTE 3: The terminology "non-redundant speech frames" refers to speech frames that have not been transmitted in any preceding packet.

Table 7.1: Encapsulation parameters (to be used as defined above)

| Radio access bearer technology | Recommended encapsulation (if no ptime and no RTCP_APP_REQ_AGG has been received) | ptime | maxptime |
|--------------------------------|--|------------------|----------|
| Default | 1 non-redundant speech frame per RTP packet Max 12 speech frames in total but not more than a received maxptime value requires | 20 | 240 |
| HSPA E-UTRAN NR | 1 non-redundant speech frame per RTP packet Max 12 speech frames in total but not more than a received maxptime value requires | 20 | 240 |
| EGPRS | 2 non-redundant speech frames per RTP packet, but not more than a received maxptime value requires Max 12 speech frames in total but not more than a received maxptime value requires | 40 | 240 |
| GIP | 1 to 4 non-redundant speech frames per RTP packet but not more than a received maxptime value requires. Max 12 speech frames in total but not more than a received maxptime | 20, 40, 60 or 80 | 240 |

NOTE 4: It is possible to send only redundant speech frames in one RTP packet.

When the radio access bearer technology is not known to the MTSI client, the default encapsulation parameters defined in Table 7.1 shall be used.

When the AMR/AMR-WB payload formats are used, the bandwidth-efficient payload format should be used unless the session setup concludes that the octet-aligned payload format is the only payload format that all parties support. The SDP offer shall include an RTP payload type where octet-align=0 is defined or where octet-align is not specified and should include another RTP payload type with octet-align=1. MTSI client offering wide-band speech shall offer these parameters and parameter settings also for the RTP payload types used for wide-band speech.

For examples of SDP offers and answers, see annex A.

The RTP payload format for DTMF events is described in Annex G.

7.4.3 Video

The following RTP payload formats shall be used:

- H.264 (AVC) video codec RTP payload format according to RFC 6184 [25], where the interleaved packetization mode shall not be used. Receivers shall support both the single NAL unit packetization mode and the non-interleaved packetization mode of RFC 6184 [25], and transmitters may use either one of these packetization modes.
- H.265 (HEVC) video codec RTP payload format according to [120].

7.4.4 Real-time text

The following RTP payload format shall be used:

- T.140 text conversation RTP payload format according to RFC 4103 [31].

Real-time text shall be the only payload type in its RTP stream because the RTP sequence numbers are used for loss detection and recovery. The redundant transmission format shall be used for keeping the effect of packet loss low.

Media type signalling for usage in SDP is specified in section 10 of RFC 4103 [31] and section 3 of RFC 4102 [49].

7.4.5 Coordination of Video Orientation

Coordination of Video Orientation consists in signalling of the current orientation of the image captured on the sender side to the receiver for appropriate rendering and displaying. When CVO is successfully negotiated it shall be signalled by the MTSI client. The signalling of the CVO uses RTP Header Extensions as specified in IETF RFC 5285 [95]. The one-byte form of the header should be used. CVO information for a 2 bit granularity of Rotation (corresponding to urn:3gpp:video-orientation) is carried as a byte formatted as follows:

```

Bit#      7   6   5   4   3   2   1   0(LSB)
Definition 0   0   0   0   C   F   R1  R0

```

With the following definitions:

C = Camera: indicates the direction of the camera used for this video stream. It can be used by the MTSI client in receiver to e.g. display the received video differently depending on the source camera.

0: Front-facing camera, facing the user. If camera direction is unknown by the sending MTSI client in the terminal then this is the default value used.

1: Back-facing camera, facing away from the user.

F = Flip: indicates a horizontal (left-right flip) mirror operation on the video as sent on the link.

0: No flip operation. If the sending MTSI client in terminal does not know if a horizontal mirror operation is necessary, then this is the default value used.

1: Horizontal flip operation

R1, R0 = Rotation: indicates the rotation of the video as transmitted on the link. The receiver should rotate the video to compensate that rotation. E.g. a 90° Counter Clockwise rotation should be compensated by the receiver with a 90° Clockwise rotation prior to displaying.

Table 7.2: Rotation signalling for 2 bit granularity

| R1 | R0 | Rotation of the video as sent on the link | Rotation on the receiver before display |
|----|----|--|---|
| 0 | 0 | 0° rotation | None |
| 0 | 1 | 90° Counter Clockwise (CCW) rotation or 270° Clockwise (CW) rotation | 90° CW rotation |
| 1 | 0 | 180° CCW rotation or 180° CW rotation | 180° CW rotation |
| 1 | 1 | 270° CCW rotation or 90° CW rotation | 90° CCW rotation |

CVO information for a higher granularity of Rotation (corresponding to urn:3GPP:video-orientation:6) is carried as a byte formatted as follows:

```

Bit#      7   6   5   4   3   2   1   0(LSB)
Definition R5  R4  R3  R2  C   F   R1  R0

```

where C and F are as defined above and the bits R5,R4,R3,R2,R1,R0 represent the Rotation, which indicates the rotation of the video as transmitted on the link. Table 7.3 describes the rotation to be applied by the receiver based on the rotation bits.

Table 7.3: Rotation signalling for 6 bit granularity

| R 1 | R 0 | R 5 | R 4 | R 3 | R 2 | Rotation of the video as sent on the link | Rotation on the receiver before display |
|-----|-----|-----|-----|-----|-----|--|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0° rotation | None |
| 0 | 0 | 0 | 0 | 0 | 1 | (360/64)° Counter Clockwise (CCW) rotation | (360/64)° CW rotation |
| 0 | 0 | 0 | 0 | 1 | 0 | (2*360/64)° CCW rotation | (2*360/64)° CW rotation |
| . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . |
| 1 | 1 | 1 | 1 | 1 | 0 | (62*360/64)° CCW rotation | (2*360/64)° CCW rotation |
| 1 | 1 | 1 | 1 | 1 | 1 | (63*360/64)° CCW rotation | (360/64)° CCW rotation |

The sending MTSI client in the terminal using a camera as source and equipped with appropriate orientation sensor(s) should compute the image orientation from the sensor(s) that indicate the rotation of the device with respect to the default camera orientation. It is recommended that appropriate filtering on the time and angular domain is applied onto the sensor's indications to prevent a "ping-pong" effect between two quantization levels in the case where the measured value is fluctuating between two quantization levels. The sending MTSI client may choose to send any orientation information not necessarily based on orientation sensor(s).

For higher granularity CVO, a terminal shall send a report at least as frequently as it would have sent a 2-bit report. A report interval shorter than this requirement should only be used when the report contains a value that differs significantly from the previous report, i.e. after taking noise removal, sensor precision, and any other relevant factors into account.

The rotation is a quantized value of the angle between the earth vertical projected onto the plane of the image as sent on the link and the image vertical. The earth vertical is a radial line starting at the center of the earth and passing through the depicted scene while the image vertical is a line passing from the middle of the bottom to the middle of the top of the image. For the case where the camera is pointing vertical or nearly vertical, the last valid value used for rotation should be used. In case there is no previous valid value, a suitable default value should be chosen.

When compensating for both rotation and flip at the receiving MTSI client, the operations shall be performed in the order of rotation compensation followed by flipping, because the order of flip and rotation operations matters when rotating 90° or 270°. The sending MTSI client shall correspondingly, when the transmitted image is both flipped and rotated, include information in the RTP Header Extension as if the transmitted image on the link was first flipped (mirrored) and then rotated, using an image perceived as upright (regardless if using portrait or landscape format) as starting point.

The MTSI client shall add the payload bytes as defined in this clause onto the last RTP packet in each group of packets which make up a key frame (I-frame or IDR frame in H.264 (AVC), or an IRAP picture in H.265 (HEVC)). The MTSI client may also add the payload bytes onto the last RTP packet in each group of packets which make up another type of frame (e.g. a P-Frame) only if the current value is different from the previous value sent.

If this is the only header extension present, a total of 8 bytes are appended to the RTP header, and the last packet in the sequence of RTP packets will be marked with both the marker bit and the Extension bit, as defined in RFC3550 [9].

When CVO is not successfully negotiated the MTSI clients are said to be in non-CVO operation. The sender in non-CVO operation should operate as follows to compensate for image rotation and potential misalignment.

If the receiver has explicitly indicated support for both [x,y] and [y,x] resolutions via the imageattr attribute during SDP negotiation (see clause 6.2.3.3 and an example in clause A.4.6), and when video is negotiated for the session, the sender should rotate the image prior to video encoding and compensate image rotation by changing the signaled Sequence Parameter Set in the video bitstream between [x,y] and [y,x] as applicable.

If the receiver has not explicitly indicated support for both [x,y] and [y,x] resolutions via the imageattr attribute during SDP negotiation, then the sender should apply rotation/padding/cropping/resizing prior to video encoding as the sender considers appropriate while keeping the resolution unchanged. As for CVO operation, the sending MTSI client in the terminal using a camera as source and equipped with appropriate orientation sensor(s) should compute the image orientation from the output of the sensor(s) that indicates the rotation of the device with respect to the default camera orientation. It is recommended that appropriate filtering on the time and angular domain is applied onto the sensor's

indications to prevent a "ping-pong" effect in the case where the measured value is fluctuating between two quantization levels. The decision of MTSI client transmitting video to change the image size needs not necessarily be based on input from orientation sensor(s).

7.4.6 RTP Retransmission

AVPF NACK messages are used by MTSI clients to indicate non-received RTP packets for video (see clause 7.3.3). The RTP Retransmission Payload Format RFC 4588 [140] supports retransmission of lost packets based on NACK feedback. Retransmission is useful if retransmitted packets arrive within the end to end delay requirements of the system. It is suitable for low RTT networks with relatively low observed packet loss [142]. If support for RTP retransmission payload format has been negotiated, the receivers shall support handling of RTP retransmission packets defined in RFC 4588 sent using SSRC multiplexing. Similarly, senders shall use RTP retransmission packets defined in RFC 4588 for packets it retransmits using SSRC multiplexing.

7.4.7 Forward Error Correction (FEC)

Forward Error Correction (FEC) can provide effective error resiliency under certain packet loss and network RTT conditions [142]. If support for FEC is negotiated, then use of a separate SSRC multiplexed FEC stream with the RTP payload defined in [141] shall be supported at both the receiver and the sender. The receiver can demultiplex the incoming stream by the SSRC field and map it to the source by using the *ssrc-group* mechanism defined in RFC 5956 [143]. The systematic FEC scheme defined in [141] is a flexible parity FEC scheme that supports various signalling of source packets used to generate the parity packets.

Other types of FEC schemes may be supported. The use of a particular FEC scheme shall be negotiated before it is used.

7.4.8 Still Images

The RTP payload format for HEVC as defined in [120] shall be used for the delivery of images and image sequences.

NOTE 1: The time distance between RTP timestamps for HEVC encoded images/image sequence may be very varying and very long compared to typical HEVC encoded video, in the order of several seconds.

7.5 Media flow

7.5.1 General

This clause contains considerations on how to use media in RTP, packetization guidelines, and other transport considerations. The use of ECN for RTP sessions is also described for speech in this clause.

The general handling of bitrate variations is described in clause 7.5.5.1. Media specific handling is described in clause 7.5.5.2 for video.

7.5.2 Media specific

7.5.2.1 Speech

7.5.2.1.1 General

This clause describes how the speech media should be packetized during a session. It includes definitions both for the cases where the access type is known and one default operation for the case when the access type is not known.

Requirements for transmission of DTMF events are described in Annex G.

7.5.2.1.2 Default operation

7.5.2.1.2.1 General

When the radio access bearer technology is not known to the MTSI client, the default encapsulation parameters defined in Table 7.1 shall be used.

The codec modes and the other codec parameters (mode-change-capability, mode-change-period, mode-change-neighbor, etc), applicable for each session, are negotiated as described in clauses 6.2.2.2 and 6.2.2.3.

When transmitting AMR or AMR-WB encoded media, codec mode changes should be aligned to every other frame border and should be performed to one of the neighbouring codec modes in the negotiated mode set, except for a MTSI media gateway, see clause 12.3.1.1. In the transmitted media, the highest codec mode of the negotiated mode-set (or of all modes, if no mode-set was included in the SDP answer) should be used, unless it is restricted by the most recently received CMR. In the received media, codec mode changes shall be accepted at any frame border and to any codec mode within the negotiated mode set.

The bandwidth-efficient payload format should be used for AMR and AMR-WB encoded media unless the session setup determines that the octet-aligned payload format must be used.

The adaptation of codec mode, aggregation and redundancy is defined in clause 10.2.

7.5.2.1.2.2 Codec Mode Requests

For AMR and AMR-WB, if the highest mode within the negotiated mode-set is acceptable for media reception, then the MTSI client in terminal shall either indicate that no codec mode request is present (i.e. value 15) or shall indicate the CMR value corresponding to the highest mode within the negotiated mode-set in the CMR bits in the AMR and/or AMR-WB payload format [28] in every outgoing RTP packet. Otherwise the highest acceptable mode within the negotiated mode-set shall be sent in CMR in each outgoing RTP packet.

NOTE 1: The MTSI client sending CMR values relies on that the remote media-sender will not send media with higher codec modes than requested by CMR, after some reaction time (round trip delay). However the remote party can send with lower modes within the negotiated mode-set, because there could be other mode limiting effects in the voice path.

For AMR and AMR-WB, the MTSI client shall accept that the remote party sends with lower modes within the negotiated mode-set than requested by the CMR. The MTSI client shall follow each received CMR and shall not use higher modes in media-encoding than indicated by the most recently received CMR, while lower modes within the negotiated mode-set are allowed any time.

NOTE 2: The codec modes in media-sending and media-receiving direction may differ in general. Received CMR values have no influence on or relation to received media-frames.

The MTSI client shall accept Codec Mode Requests signalled with the CMR bits in the AMR and/or AMR-WB payload format in every incoming RTP packet.

For EVS, the CMR related procedures in subclause A.2.2.1.1 of TS 26.445 [125] apply.

If the MTSI client supports RTCP-APP packets, it shall also accept CMR in every incoming RTCP-APP packet.

The MTSI client shall follow each received CMR as soon as possible.

NOTE 3: There is no upper limit defined for the reaction time; it is expected that typically the media-sender reacts within less than 40ms after the reception of a new CMR value.

7.5.2.1.2.3 Frame aggregation and redundancy

The MTSI client should send one speech frame encapsulated in each RTP packet unless the session setup or adaptation request defines that the other MTSI client wants to receive another encapsulation variant.

The MTSI client should request to receive one speech frame encapsulated in each RTP packet but shall accept any number of frames per RTP packet up to the maximum limit of 12 speech frames per RTP packet.

For application-layer redundancy, see clause 9.2.

7.5.2.1.3 HSPA

Use default operation as defined in clause 7.5.2.1.2.

NOTE: The RLC PDU sizes defined in TR 25.993 [33] have been optimized for the codec modes, payload formats and frame encapsulations defined in the default operation in clause 7.5.2.1.2.

7.5.2.1.4 EGPRS

Use default operation as defined in clause 7.5.2.1.2, except that the MTSI client in terminal

- should send two speech frames encapsulated in each RTP packet unless the session setup or adaptation request defines that the other PS end-point want to receive another encapsulation variant;
- should request receiving two speech frames encapsulated in each RTP packet but shall accept any number of frames per RTP packet up to the maximum limit of 12 speech frames per RTP packet.

7.5.2.1.5 GIP

Use default operation as defined in clause 7.5.2.1.2, except that the MTSI client in terminal:

- should send 0, 1, 2, 3 or 4 non-redundant speech frames encapsulated in each RTP packet unless the session setup or adaptation request defines that other PS end-point want to receive another encapsulation variant;
- should request receiving 1 to 4 speech frames in each RTP packet but shall accept any number of frames per RTP packet up to the maximum limit of 12 speech frames per RTP packet;
- may use application layer redundancy, in which case the MTSI client in terminal may encapsulate up to 12 speech frames in each RTP packet, with a maximum of four non-redundant speech frames.

7.5.2.1.6 Initial codec mode for AMR and AMR-WB

To avoid congestion on the link and to improve inter-working with CS GERAN when AMR or AMR-WB is used and when more than one codec mode is allowed in the session, the MTSI client in terminal should limit the initial codec mode (ICM) to one of the lowest codec modes for an Initial Waiting Time from the beginning of the RTP stream, or until it receives one of the following:

- a frame-block with rate control information; or
- an RTCP message with rate control information; or
- reception quality feedback information, e.g. PLR or jitter in RTCP Sender Reports or Receiver Reports, indicating that the currently used codec mode is too high for the current operating condition.

The value for the Initial Waiting Time is 600 ms when ECN is not used and 500 ms when ECN is used, unless configured differently by the MTSI Media Adaptation Management as described in Clause 17.

The rate control information can either be: a CMR with a value other than '15' in the RTP payload; or a CMR with a value other than '15' in an RTCP_APP message (see Clause 10.2.1).

NOTE 1: A CMR with a value of '15' means that no mode request is present [28].

If no rate control information is received within the Initial Waiting Time, then the sending MTSI client in terminal should gradually increase the codec mode from the ICM towards the highest codec mode allowed in the session. While not detecting poor transmission performance or not receiving rate control information, the sending MTSI client in terminal should use step-wise up-switch to avoid introducing congestion during the upwards adaptation. The step-wise up-switch should be performed by switching to the next higher codec mode in the allowed mode set and then waiting for an Initial Up-switch Waiting Time before each subsequent up-switch until the first down-switch occurs.

The value for the Initial Up-switch Waiting Time is 600 ms when ECN is not used and 500 ms when ECN is used, unless configured differently by the MTSI Media Adaptation Management as described in Clause 17.

The following rules can be used for determining the ICM:

- If 1 codec mode is included in the mode-set then this should be the ICM.

- If 2 or 3 codec modes are included in the mode-set then the ICM should be the codec mode with the lowest rate.
- If 4 or more codec modes are included in the mode-set then the ICM should be the codec mode with the 2nd lowest rate.

NOTE 2: Without ECN, the Initial Waiting Time needs to be long enough to allow the receiver to collect reliable statistics for the adaptation, e.g. for PLR-triggered or jitter-triggered adaptation. With ECN, a congested network can immediately mark IP packets with ECN-CE, which allows the ECN-triggered adaptation react sooner. The Initial Waiting Time can therefore be shorter when ECN is used. The same applies for the Initial Up-switch Waiting Time.

7.5.2.1.7 E-UTRAN and NR

Use the default operation as defined in Clause 7.5.2.1.2.

7.5.2.1.8 Initial codec mode for EVS

When the EVS AMR-WB IO mode is used from the start of the session, the Initial Codec Mode (ICM) should be selected as defined in Clause 7.5.2.1.6 for AMR-WB.

When EVS Primary mode is used from the start of the session, the following principles apply for the selection of the Initial Codec Mode bit-rate (ICMbr):

- If GBR is known and if GBR is less than MBR, the ICMbr should be aligned with the GBR or should be lower than GBR.

When EVS Primary mode is used from the start of the session, the Initial Codec Mode audio bandwidth (ICMab) should be the highest audio bandwidth negotiated for the Initial Codec Mode bit-rate (ICMbr).

7.5.2.1.9 Dual-mono

An MTSI client may support dual-mono operation for EVS.

The packetization of dual-mono for EVS is described in [125]. When the EVS Primary mode is used for dual-mono encoding, the Header-full format must be used for all RTP packets.

When offering dual-mono for an RTP payload type number, the number of channels is set to 2, see SDP example in Annex A.14.

7.5.2.2 Video

An MTSI client should follow general strategies for error-resilient coding (segmentation) and packetization as specified by each codec [24][119] and RTP payload format [25][120] specification. Further guidelines on how the video media data should be packetized during a session are provided in this clause.

Coded pictures should be encoded into individual segments:

- For H.264 (AVC), a slice corresponds to such a segment.
- For H.265 (HEVC), a slice segment corresponds to such a segment.

Each individual segment should be encapsulated in one RTP packet. Each RTP packet should be smaller than the Maximum Transfer Unit (MTU) size.

NOTE 1: Unnecessary video segmentation, e.g. within RTP packets, may reduce coding efficiency.

NOTE 2: RTP packet fragmentation, e.g. across UDP boundaries, may decrease transport overhead and reduce error robustness. Hence, packet size granularity is a trade-off between error robustness and overhead that may be tuned according to bearer access characteristics if available.

NOTE 3: In most cases, the MTU-size has a direct relationship with the bearer of the radio network.

7.5.2.3 Text

Real-time text is intended for human conversation applications. Text shall not be transferred with higher rate than 30 characters per second (as defined for cps in section 6 of RFC 4103 [31]). A text-capable MTSI client shall be able to receive text with cps set up to 30.

7.5.3 Media synchronization

7.5.3.1 General

RTCP SR shall be used for media synchronization by setting the NTP and RTP timestamps according to RFC 3550 [9]. To enable quick media synchronization when a new media component is added, or an MTSI session is initiated, the RTP sender should send RTCP Sender Reports for all newly started media components as early as possible.

NOTE: An MTSI sender can signal in SDP that no synchronization between media components is required. See clause 6.2.6 and clause A.7.

7.5.3.2 Text

The media synchronization requirements for real-time text are relaxed. A synchronization error between text and other media of a maximum of 3 seconds is accepted. Since this is longer than the maximum accepted latency, no specific methods need to be applied to assure to meet the requirement

7.5.4 ECN usage in RTP sessions

Once the ECN negotiation has been completed as defined in [84], then only ECT(0) shall be used when marking packets with ECT, [83]. When ECN is used for an RTP stream then the sending MTSI client shall mark every packet with ECT until the end of the session or until the session is re-negotiated to no longer use ECN. The leap-of-faith method is used for the ECN initiation.

Handling of ECN Congestion Experience (ECN-CE) marked packets is described in clause 10.

7.5.5 Handling of bit-rate variations

7.5.5.1 General

An MTSI client in terminal using variable bitrate encoding shall ensure for speech, and should ensure for video, that the transmitted bandwidth does not exceed the negotiated bandwidth ($b=AS$) nor the QoS bandwidth parameters (MBR, GBR) for the bearer, if defined and known. This can, in general, be done in two ways: either by controlling the bit-rate used for encoding each media frame or by controlling when the generated packets are transmitted.

NOTE 1: This bit-rate control is not to be confused with rate adaptation, which is described in clause 10. The main purpose of the bitrate control is to reduce the risk that policing functions in the network drop media packets since the bandwidth of packets is temporarily exceeding the allowed bandwidth, which is typically configured in a static fashion. Rate adaptation is instead used to ensure that the used bandwidth is sufficiently low to avoid packet losses and extended delays when operating conditions change.

NOTE 2: Controlling when packets are transmitted is called "packet pacing". The drawback with packet pacing is the increase of end-to-end delay. Excessive use of packet pacing can lead to not fulfilling the service requirements defined in TS 22.105 [34]. Packet pacing therefore needs to be used with care.

The method to control the encoding bitrate and/or the transmission of packets is left for the discretion of the implementation. However, the average bandwidth of transmitted packets shall be calculated over a sliding window that is no longer than T seconds. Different media types may use different window lengths. The default window length T is 2 seconds if nothing else is specified below.

The bitrate control ensures that the average bandwidth of transmitted media packets does not exceed the maximum allowed bandwidth in the sending direction.

The maximum allowed bandwidth for the sending direction is the smaller of:

- The $b=AS$ bandwidth,
- The MBR for the bearer, after compensating for the RTCP bandwidth, if known.

7.5.5.2 Video

Video encoders use variable bitrate encoding to ensure a sufficiently low average bit-rate, which allow temporarily encoding a few frames at high bit-rate to match the video content with high motion activity or complex scenes, or to send an Intra frame, e.g. , when a Full Intra Request (FIR) is received. The bitrate control should ensure that the average bandwidth of transmitted media packets does not exceed the maximum allowed bandwidth in the sending direction.

The bit-rate control requires managing the following properties:

- The size of large frame to transmit;
- The proportion of neighboring frames that can be reduced in size;
- The length of averaging window.

When any two of the properties are known, it is possible to determine the third.

TR 26.924 [144] Annex A describes a method for determining the length of the averaging window from the other two properties. The same method can also be used to determine the portion of the bit-rate for frames neighboring the large frame, which needs to be reduced depending on the size of the frame and the length of the averaging window.

The recommended procedure consists of reducing the encoding bit-rate of the frames neighboring the large frame in a balanced fashion, taking into account the spatio-temporal tradeoff of video quality when the encoding parameters such as quantization step or frame rate are adjusted. This may even require dropping some frames before they are encoded.

When a FIR is received, the MTSI client in terminal may delay the generation and transmission of an Intra frame by up to a half of the averaging window.

7.5.5.3 Speech

When speech is operated in a Source-Controlled Variable Bit Rate mode (e.g., EVS 5.9VBR), both the GBR and MBR need to be set at least as high as the highest rate used by the codec in this mode (e.g., 8.0kbps for EVS 5.9VBR). These rates may be set higher for a session if, in addition to the VBR mode, a higher rate codec mode is also negotiated for the session. Therefore, packet policing on the MBR or GBR will not prevent the transport of VBR media packets regardless of the averaging window used.

8 Jitter buffer management in MTSI clients in terminals

8.1 General

This clause specifies mechanisms to handle delay jitter in MTSI clients in terminals.

8.2 Speech

8.2.1 Terminology

In the following paragraph(s), Jitter Buffer Management (JBM) denotes the actual buffer as well as any control, adaptation and media processing algorithm (excluding speech decoder) used in the management of the jitter induced in the transport channel. An illustration of an exemplary structure of an MTSI speech receiver with adaptive jitter buffer is shown in figure 8.1 to clarify the terminology and the relation between different functional components.

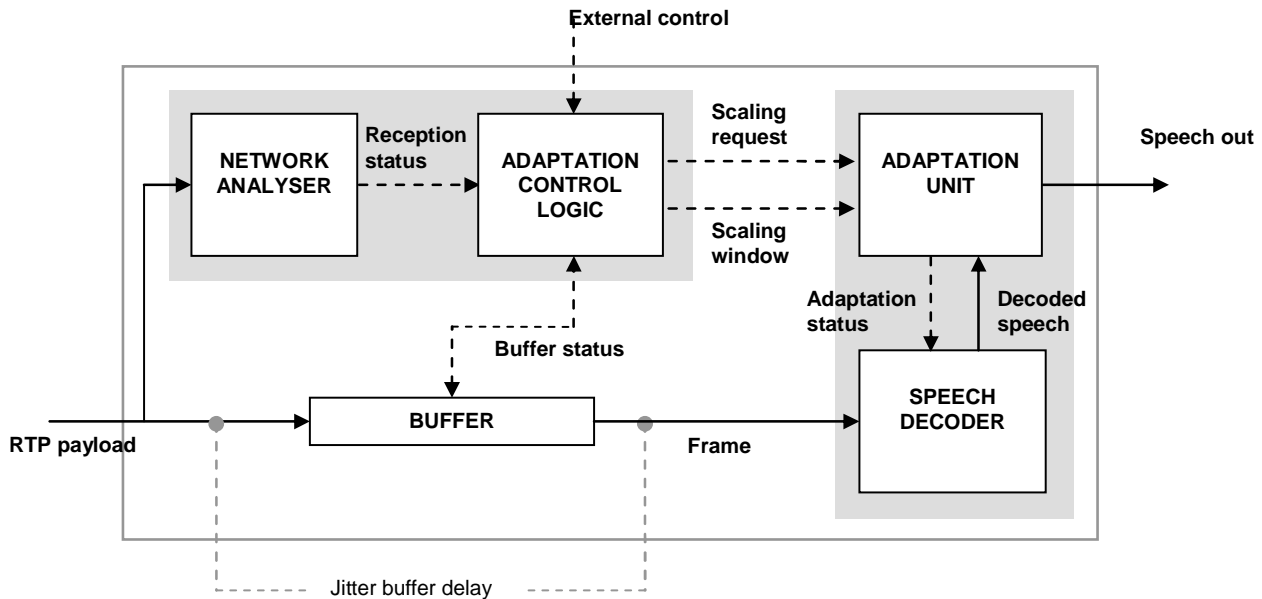


Figure 8.1: Example structure of an MTSI speech receiver

The blocks "network analyzer" and "adaptation control logic" together with the information on buffer status form the actual buffer control functionality, whereas "speech decoder" and "adaptation unit" provide the media processing functionality. Note that the external playback device control driving the media processing is not shown in figure 8.1.

The grey dashed lines indicate the measurement points for the jitter buffer delay, i.e. the difference between the decoder consumption time and the arrival time of the speech frame to the JBM.

The functional processing blocks are as follows:

- **Buffer:** The jitter buffer unpacks the incoming RTP payloads and stores the received speech frames. The buffer status may be used as input to the adaptation decision logic. Furthermore, the buffer is also linked to the speech decoder to provide frames for decoding when they are requested for decoding.
- **Network analyser:** The network analysis functionality is used to monitor the incoming packet stream and to collect reception statistics (e.g. jitter, packet loss) that are needed for jitter buffer adaptation. Note that this block can also include e.g. the functionality needed to maintain statistics required by the RTCP if it is being used.
- **Adaptation control logic:** The control logic adjusting playback delay and operating the adaptation functionality makes decisions on the buffering delay adjustments and required media adaptation actions based on the buffer status (e.g. average buffering delay, buffer occupancy, etc.) and input from the network analyser. Furthermore, external control input, including RTCP from the sender, can be used e.g. to enable inter-media synchronisation, to adapt the jitter buffer, or other external scaling requests. The control logic may utilize different adaptation strategies such as fixed jitter buffer (without adaptation and time scaling), simple adaptation during comfort noise periods or buffer adaptation also during active speech. The general operation is controlled with desired proportion of frames arriving late, adaptation strategy and adaptation rate.
- **Speech decoder:** The standard AMR, AMR-WB or EVS speech decoder. Note that the speech decoder is also assumed to include error concealment / bad frame handling functionality. Speech decoder may be used with or without the adaptation unit.
- **Adaptation unit:** The adaptation unit shortens or extends the output signal length according to requests given by the adaptation control logic to enable buffer delay adjustment in a transparent manner. The adaptation is performed using the frame based or sample based time scaling on the decoder output signal during comfort noise periods only or during active speech and comfort noise. The buffer control logic should have a mechanism to limit the maximum scaling ratio. Providing a scaling window in which the targeted time scale modifications are performed improves the situation in certain scenarios - e.g. when reacting to the clock drift or to a request of inter-media (re)synchronization - by allowing flexibility in allocating the scaling request on several frames and performing the scaling on a content-aware manner. The adaptation unit may be implemented either in a separate entity from the speech decoder or embedded within the decoder.

8.2.2 Functional requirements for jitter-buffer management

The functional requirements for the speech JBM guarantee appropriate management of jitter which shall be the same for all speech JBM implementations used in MTSI clients in terminals. A JBM implementation used in MTSI shall support the following requirements, but is not limited in functionality to these requirements. They are to be seen as a minimum set of functional requirements supported by every speech JBM used in MTSI.

Speech JBM used in MTSI shall:

- support source-controlled rate operation as well as non-source-controlled rate operation;
- be able to receive the de-packetized frames out of order and present them in order for decoder consumption;
- be able to receive duplicate speech frames and only present unique speech frames for decoder consumption;
- be able to handle clock drift between the encoding and decoding end-points.

8.2.3 Minimum performance requirements for jitter-buffer management

8.2.3.1 General

An MTSI client in terminal supporting speech shall use a JBM fulfilling the minimum performance requirements defined in this clause. The JBM specified in [128] fulfils these minimum performance requirements and should be used for EVS. The EVS JBM may also be used for other codecs.

The jitter buffering time is the time spent by a speech frame in the JBM. It is measured as the difference between the decoding start time and the arrival time of the speech frame to the JBM. The frames that are discarded by the JBM are not counted in the measure.

The minimum performance requirements consist of objective criteria for delay and jitter-induced concealment operations. In order for a JBM implementation to pass the minimum performance requirements all objective criteria shall be met.

A JBM implementation used in MTSI shall comply with the following design guidelines:

1. The overall design of the JBM shall be to minimize the buffering time at all times while still conforming to the minimum performance requirements of jitter induced concealment operations and the design guidelines for sample-based timescaling (as set in bullet point 3);
2. If the limit of jitter induced concealment operations cannot be met, it is always preferred to increase the buffering time in order to avoid growing jitter induced concealment operations going beyond the stated limit above. This guideline applies even if that means that end-to-end delay requirement given in TS 22.105 [34] can no longer be met;
3. If sample-based time scaling is used (after speech decoder), then artefacts caused by time scaling operation shall be kept to a minimum. Time scaling means the modification of the signal by stretching and/or compressing it over the time axis. The following guidelines on time scaling apply:
 - Use of a high-quality time scaling algorithm is recommended;
 - The amount of scaling should be as low as possible;
 - Scaling should be applied as infrequently as possible;
 - Oscillating behaviour is not allowed.

NOTE: If the end-to-end delay for the ongoing session is known to the MTSI client in terminal and measured to be less than 150 ms (as defined in TS 22.105 [34]), the JBM may relax its buffering time minimization criteria in favour of reduced JBM adaptation artefacts if such a relaxation will improve the media quality. Note that a relaxation is not allowed when testing for compliance with the minimum performance requirements specified in clauses 8.2.3.2.2 and 8.2.3.2.3.

8.2.3.2 Objective performance requirements

8.2.3.2.1 General

The objective performance requirements consist of criteria for delay, time scaling and jitter-induced concealment operations.

The objective minimum performance requirements are divided into three parts:

1. Limiting the jitter buffering time to provide as low end-to-end delay as possible.
2. Limiting the jitter induced concealment operations, i.e. setting limits on the allowed induced losses in the jitter buffer due to late losses, re-bufferings, and buffer overflows.
3. Limiting the use of time scaling to adapt the buffering depth in order to avoid introducing time scaling artefacts on the speech media.

In order to fulfil the objective performance requirements, the JBM under test needs to pass the respective criteria using the six channels as defined in clause 8.2.3.3. Note that in order to pass the criteria for a specific channel, all three requirements must be fulfilled.

8.2.3.2.2 Jitter buffer delay criteria

The reference delay computation algorithm in Annex D defines the performance requirements for the set of delay and error profiles described in clause 8.2.3.3. The JBM algorithm under test shall meet these performance requirements. The performance requirements shall be a threshold for the Cumulative Distribution Function (CDF) of the speech-frame delay introduced by the reference delay computation algorithm. A CDF threshold is set by shifting the reference delay computation algorithm CDF 60 ms. The speech-frame delay CDF is defined as:

$$P(x) = \text{Probability}(\text{delay_compensation_by_JBM} \leq x)$$

The relation between the reference delay computation algorithm and the CDF threshold is outlined in figure 8.2.

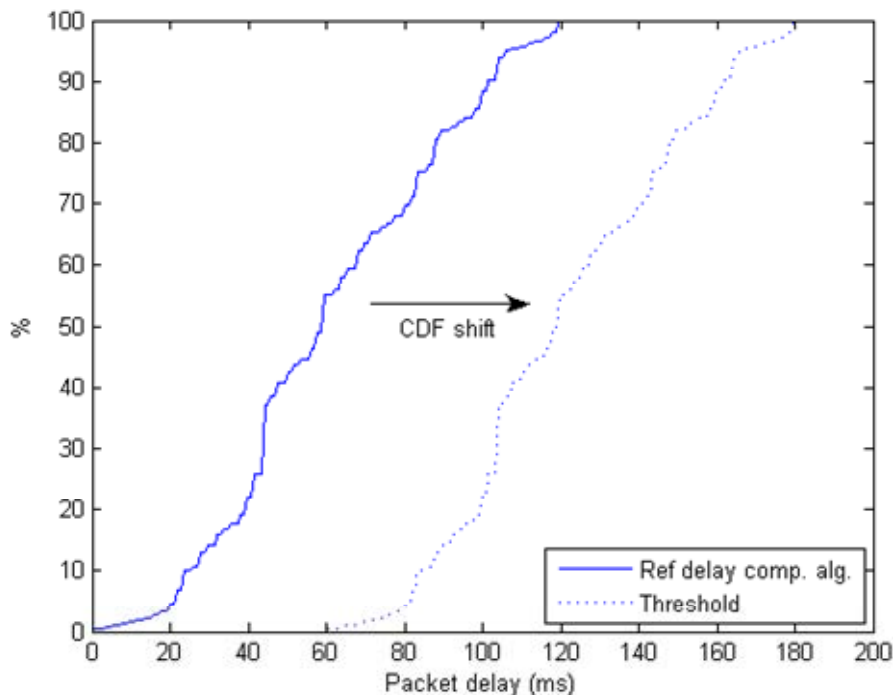


Figure 8.2: Example showing the relation between the reference delay algorithm and the CDF threshold - the delay and error profile 4 in table 8.1 has been used

The JBM algorithm under test shall achieve lower or same delay than that set by the CDF threshold for at least 90 % of the speech frames. The values for the CDF shall be collected for the full length of each delay and error profile. The delay measure in the criteria is measured as the time each speech frame spends in the JBM; i.e. the difference between the decoder consumption time and the arrival time of the speech frame to the JBM.

The parameter settings for the reference delay computation algorithm are:

- adaptation_lookback = 200;
- delay_delta_max = 20;
- target_loss= 0.5.

8.2.3.2.3 Jitter induced concealment operations

The jitter induced concealment operations include:

- JBM induced removal of a speech frame, i.e. buffer overflow or intentional frame dropping when reducing the buffer depth during adaptation.
- Deletion of a speech frame because it arrived at the JBM too late.
- Modification of the output timeline due to link loss.
- Jitter-induced insertion of a speech frame controlled by the JBM (e.g. buffer underflow).

Link losses handled as error concealment and not changing the output timeline shall not be counted in the jitter induced concealment operations.

$$\text{Jitter loss rate} = \text{JBM triggered concealed frames} / \text{Number of transmitted frames}$$

The jitter loss rate shall be calculated for active speech frames only.

NOTE: SID_FIRST and SID_UPDATE frames belong to the non-active speech period, hence concealment for losses of such frames should not be included in the statistics.

The jitter loss rate shall be below 1% for every channel measured over the full length of the respective channel. The value of 1 % was chosen because such a loss rate will usually not significantly reduce the speech quality.

8.2.3.3 Delay and error profiles

Six different delay and error profiles are used to check the tested JBM for compliance with the minimum performance requirements. The profiles span a large range of operating conditions in which the JBM shall provide sufficient performance for the MTSI service. All profiles are 7500 IP packets long.

Table 8.1: Delay and error profile overview - The channels are attached electronically

| Profile | Characteristics | Packet loss rate (%) | Filename |
|---------|--|----------------------|-------------------------|
| 1 | Low-amplitude, static jitter characteristics, 1 frame/packet | 0 | dly_error_profile_1.dat |
| 2 | Hi-amplitude, semi-static jitter characteristics, 1 frame/packet | 0.24 | dly_error_profile_2.dat |
| 3 | Low/high/low amplitude, changing jitter, 1 frame/packet | 0.51 | dly_error_profile_3.dat |
| 4 | Low/high/low/high, changing jitter, 1 frame/packet | 2.4 | dly_error_profile_4.dat |
| 5 | Moderate jitter with occasional delay spikes, 2 frames/packet (7 500 IP packets, 15 000 speech frames) | 5.9 | dly_error_profile_5.dat |
| 6 | Moderate jitter with severe delay spikes, 1 frame/packet | 0.1 | dly_error_profile_6.dat |

The attached profiles in the zip-archive "delay_and_error_profiles.zip" are formatted as raw text files with one delay entry per line. The delay entries are written in milliseconds and packet losses are entered as "-1". Note that when testing for compliance, the starting point in the delay and error profile shall be randomized.

8.2.3.4 Speech material for JBM minimum performance evaluation

The files described in table 8.2 and attached to the present document in the zip-archive "JBM_evaluation_files.zip" shall be used for evaluation of a JBM against the minimum performance requirements. The data is stored as RTP packets, formatted according to "RTP dump" format [41]. The input to these files is AMR or AMR-WB encoded frames, encapsulated into RTP packets using the octet-aligned mode of the AMR RTP payload format [28].

Table 8.2: Input files for JBM performance evaluation - The files are attached electronically

| Codec | Frames per RTP packet | Filename |
|---------------------|-----------------------|-------------------------|
| AMR (12.2 kbps) | 1 | test_amr122_fpp1.rtp |
| AMR (12.2 kbps) | 2 | test_amr122_fpp2.rtp |
| AMR-WB (12.65 kbps) | 1 | test_amrwb1265_fpp1.rtp |
| AMR-WB (12.65 kbps) | 2 | test_amrwb1265_fpp2.rtp |

8.3 Video

Video receivers should implement an adaptive video de-jitter buffer. The overall design of the buffer should aim to minimize delay, maintain synchronization with speech, and minimize dropping of late packets. The exact implementation is left to the implementer.

8.4 Text

Conversational quality of real-time text is experienced as being good, even with up to one second end-to-end text delay. Strict jitter buffer management is therefore not needed for text. Basic jitter buffer management for text is described in section 5 of RFC 4103 [31] where a calculation is described for the time allowed before an extra delayed text packet may be regarded to be lost.

9 Packet-loss handling

9.1 General

This clause specifies some methods to handle conditions with packet losses. Packet losses in general will also trigger adaptation, which is specified in clause 10.

The 'a=bw-info' attribute defined in clause 19 allows for negotiating how much additional bandwidth (if any) may be used for application layer redundancy in the session. When application layer redundancy is used, the media bandwidth negotiated for the session may need to be increased, i.e. by increasing the value used for the b=AS bandwidth modifier. The b=AS bandwidth modifier is however only a single value, which also applies only to the receiving direction. When an MTSI client in terminal sends the SDP, it is therefore not possible for the networks and the other client to know if the intention is to use the entire media bandwidth all the time (both with and without redundancy); or if the intention is to use the b=AS bandwidth only when redundancy is needed and to use a lower bandwidth when redundancy is not needed. It is also not possible to know what the MTSI client in terminal can do in the sending direction. The 'a=bw-info' attribute (see clause 19) offers an improved negotiation mechanism to better know what the MTSI client in terminal can do and what it intends to do. This is further discussed in TR 26.924 [144].

Improved error robustness can be enabled by packet-loss handling procedures of the codec or codec mode, via the adaptation procedures described in clause 10, or other mechanisms. Annex X specifies the CHEM feature which enables the 3GPP system to exploit error robustness to avoid, delay, or reduce the need to handoff a terminal due to degradation in the media quality. Annex Y provides example PLR threshold values that can be used for different codec configurations.

9.2 Speech

9.2.1 General

This clause provides a recommendation for a simple application layer redundancy scheme that is useful in order to handle operational conditions with severe packet loss rates. Simple application layer redundancy is generated by encapsulating one or more previously transmitted speech frames into the same RTP packet as the current previously not transmitted frame(s). An RTP packet may thus contain zero, one or several redundant speech frames and zero, one or several non-redundant speech frames.

When transmitting redundancy, the MTSI client should switch to a lower codec mode, if possible. An MTSI client using AMR or AMR-WB shall utilize the codec mode rates within the negotiated codec mode set with the negotiated adaptation steps and limitations as defined by mode-change-neighbor and mode-change-period. It is recommended to not send redundant speech frames before the targeted codec mode is reached. Table 9.1 defines the recommended codec modes for different redundancy level combinations.

When application layer redundancy is used for AMR or AMR-WB encoded speech media, the transmitting application may use up to 300 % redundancy, i.e. a speech frame transported in one RTP packet may be repeated in 3 other RTP packets.

Table 9.1: Recommended codec modes and redundancy level combinations when redundancy is supported

| Redundancy level | No redundancy | 100 % redundancy |
|--|---------------|------------------|
| Narrow-band speech | AMR 12.2 | AMR 5.9 |
| Wide-band speech (when wide-band is supported) | AMR12.65 | AMR 6.60 |

9.2.2 Transmitting redundant frames

When transmitting redundant frames, the redundant frames should be encapsulated together with non-redundant media data as shown in figure 9.1. The frames shall be consecutive with the oldest frame placed first in the packet and the most recent frame placed last in the packet. The RTP Timestamp shall represent the sampling time of the first sample in the oldest frame transmitted in the packet.

NOTE 1: When switching from no redundancy to using redundancy, the RTP Timestamp may be the same for consecutive RTP packets.

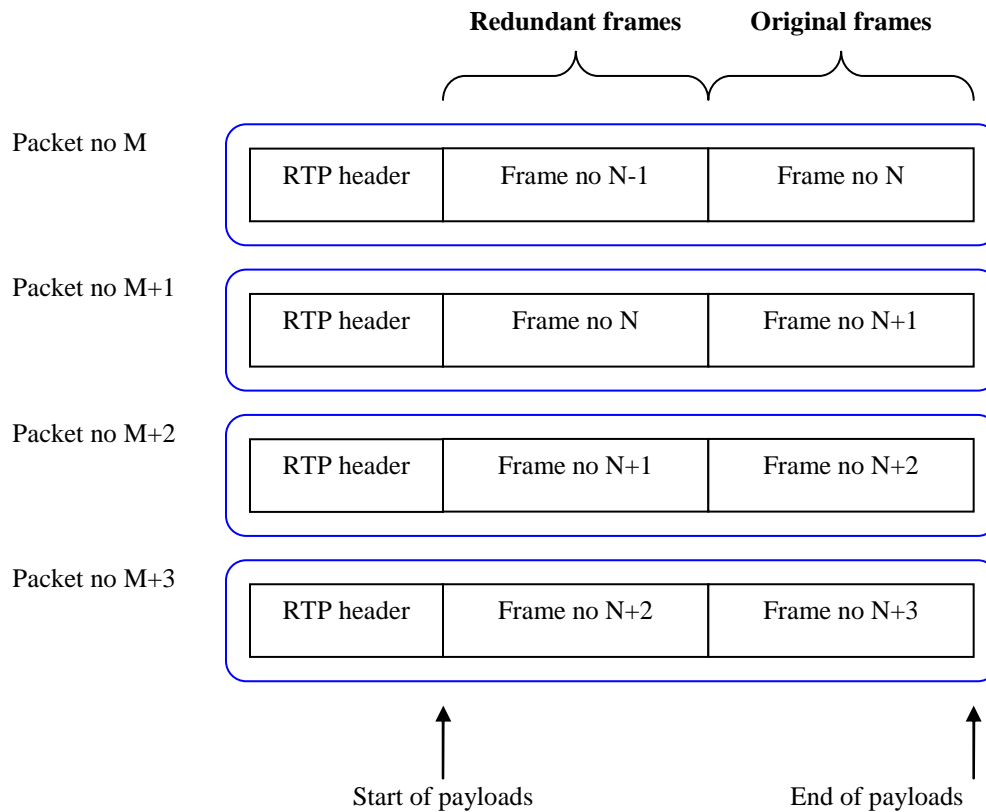


Figure 9.1: Redundant and non-redundant frames in the case of 100 % redundancy, when the original packing is 1 frame per packet

Figure 9.1 shows only one non-redundant frame encapsulated together with one redundant frame. It is allowed to encapsulate several non-redundant frames with one or several redundant frames. The following combinations of non-redundant frames and redundant frames can be used.

Table 9.2: Example frame encapsulation with different redundancy levels and when maxptime is 240

| Original encapsulation (without redundancy) | Encapsulation with 100 % redundancy | Encapsulation with 200 % redundancy | Encapsulation with 300 % redundancy |
|---|---|---|--|
| 1 frame per packet | ≤ 1 non-redundant frame and ≤ 1 redundant frame | ≤ 1 non-redundant frame and ≤ 2 redundant frames | ≤ 1 non-redundant frame and ≤ 3 redundant frames |
| 2 frames per packet | ≤ 2 non-redundant frames and ≤ 2 redundant frames | ≤ 2 non-redundant frames and ≤ 4 redundant frames | ≤ 2 non-redundant frames and ≤ 6 redundant frames |
| 3 frames per packet | ≤ 3 non-redundant frames and ≤ 3 redundant frames | ≤ 3 non-redundant frames and ≤ 6 redundant frames | ≤ 3 non-redundant frames and ≤ 9 redundant frames |
| 4 frames per packet | ≤ 4 non-redundant frames and ≤ 4 redundant frames | ≤ 4 non-redundant frames and ≤ 8 redundant frames | Not allowed since maxptime does not allow more than 12 frames per RTP packet in this example |

With a maxptime value of 240, it is possible to encapsulate up to 12 frames per packet. It is therefore not allowed to use 300 % when the original encapsulation is 4 frames per packet, as shown in table 9.2. If the receiver's maxptime value is lower than 240 then even more combinations of original encapsulation and redundancy level will be prohibited.

The sender shall also ensure that the Maximum Transfer Unit (MTU) is not exceeded when sending the IP/UDP/RTP packet.

Figure 9.2 shows an example where the frame aggregation is 2 frames per packet and when 100 % redundancy added.

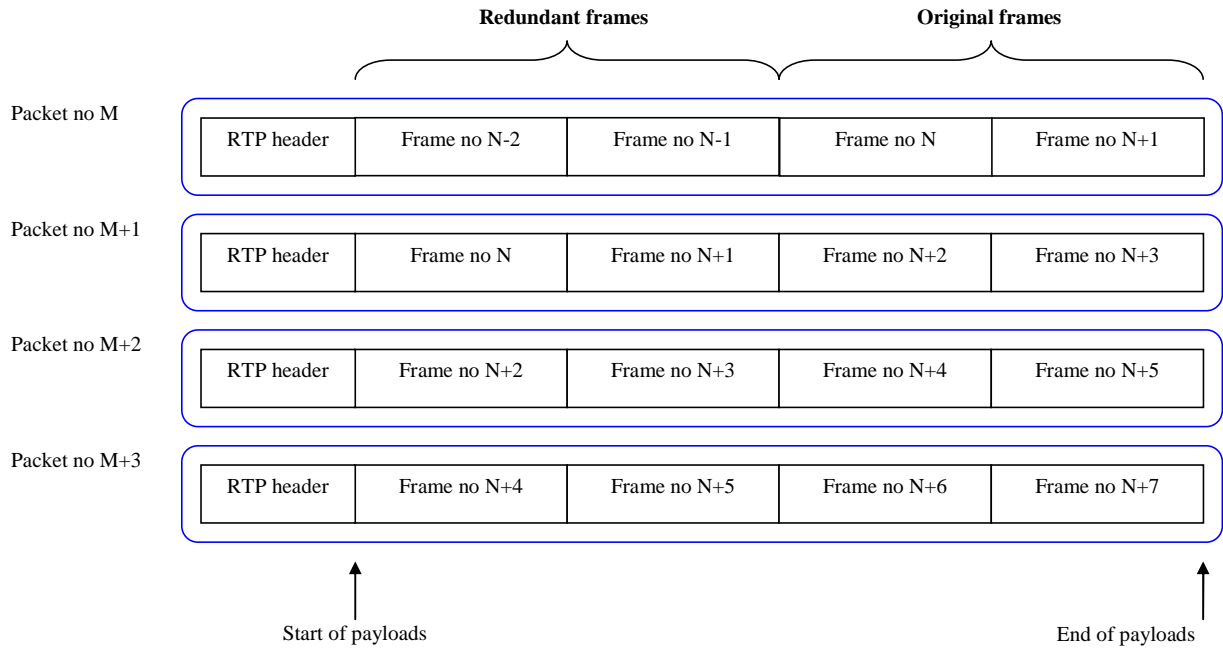


Figure 9.2: Redundant and non-redundant frames in the case of 100 % redundancy, when the original packing is 2 frames per packet

A redundant frame may be replaced by a NO_DATA frame. If the transmitter wants to encapsulate non-consecutive frames into one RTP packet, then NO_DATA frames shall be inserted for the frames that are not transmitted in order to create frames that are consecutive within the packet. This method is used when sending redundancy with an offset, see figure 9.3.

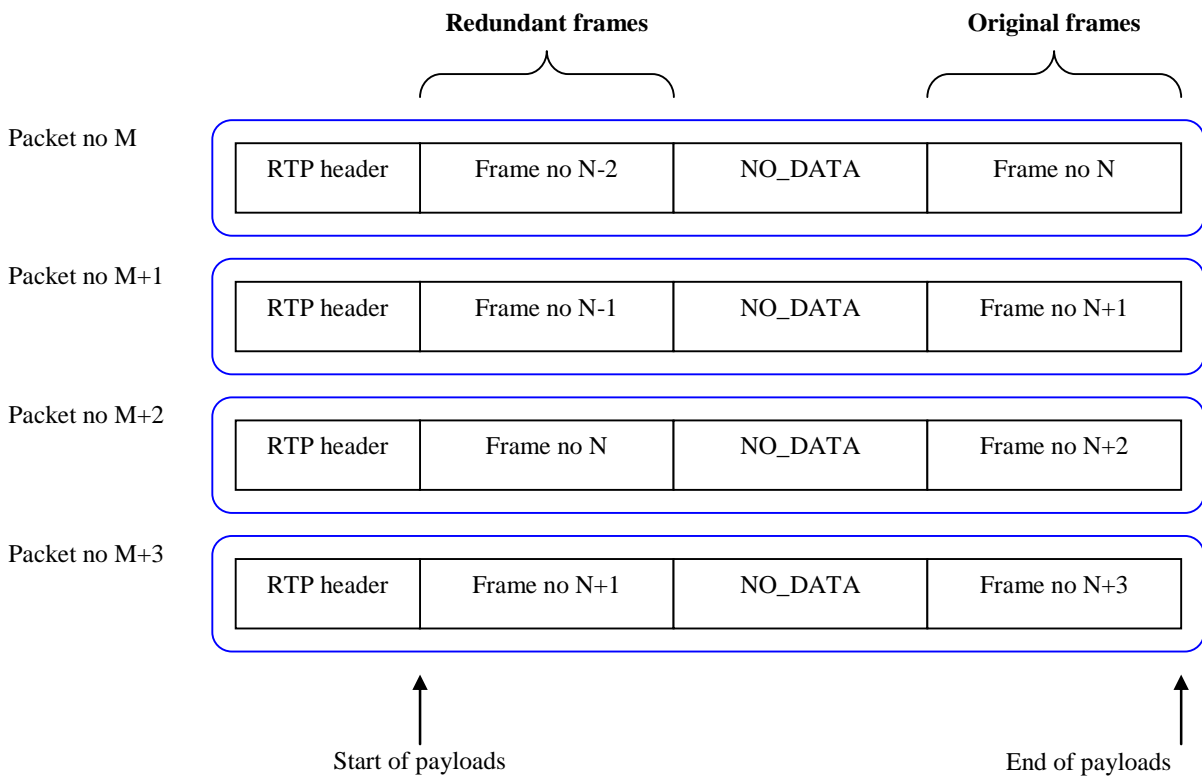


Figure 9.3: Redundant and non-redundant frames in the case of 100 % redundancy, when the original packing is 1 frame per packet and when the redundancy is transmitted with an offset of 20 ms

Note that with this scheme, the receiver may receive a frame 3 times: first the non-redundant encoding; then as a NO_DATA frame; and finally the redundant frame. Other combinations of redundancy and offset may result in receiving even more copies of a frame. The proper receiver behaviour is described in the AMR/AMR-WB payload format [28] and in the EVS payload format [125], respectively.

For any combinations of frame aggregation, redundancy and redundancy offset, the transmitter shall not exceed the frame encapsulation limit indicated by the receiver's maxptime value when constructing the RTP packet.

When source controlled rate operation is used, it is allowed to send redundant media data without any non-redundant media, if no non-redundant media is available.

NOTE 2: When going from active speech to DTX, there may be no non-redundant frames in the end of the talk spurt while there still are redundant frames that need to be transmitted.

In the end of a talk spurt, when there are no more non-redundant frames to transmit, it is allowed to drop the redundant frames that are in the queue for transmission.

NOTE 3: This ensures that it is possible to use redundancy without increasing the packet rate. The quality degradation by having less redundancy for the last frames should be negligible since these last frames typically contain only background noise.

9.2.3 Receiving redundant frames

In order to receive and decode redundant media properly, the receiving application shall sort the received frames based on the RTP Timestamp and shall remove duplicated frames. If multiple versions of a frame are received, i.e. encoded with different bitrates, then the frame encoded with the highest bitrate should be used for decoding.

9.3 Video

9.3.1 General

MTSI clients can use the following mechanisms to recover from packet losses:

- AVPF Generic NACK and Picture Loss Indication (PLI) feedback messages
- RTP Retransmission
- Forward Error Correction (FEC)

These mechanisms offer different performance tradeoff according to channel conditions: end-to-end delay, bandwidth, rate and profile of packet losses.

AVPF NACK messages are used by MTSI clients to indicate non-received RTP packets for video (see clause 7.3.3). An MTSI client transmitting video can use this information, as well as the AVPF Picture Loss Indication (PLI), to at its earliest opportunity take appropriate action to recover video from errors for the MTSI client that sent the NACK or PLI message. Recovery from error action is defined as sending a recovery picture that is equivalent to a good frame in clause 16.2.1, sending Gradual Decoder Refresh (GDR) that results in a good frame, or retransmitting missing packets. Requirements and recommendations for packet loss handling are described below.

Recommendation regarding usage under various channel conditions of error recovery mechanisms available for MTSI clients are provided in clause 9.3.4.

9.3.2 Receiver behaviour

When NACK and PLI have been negotiated without retransmission support for the session then an MTSI client in terminal receiving media:

- shall immediately queue a NACK message for RTCP scheduling upon detection of first error after decoding a good frame.
- should repeat queuing NACK messages for RTCP scheduling after an RWT duration if recovery picture does not arrive.

- shall queue a PLI message for RTCP scheduling if a recovery picture does not arrive in two RWT duration, and shall then stop sending NACK messages that relate to the same data as that PLI.
- shall repeat queuing PLI messages for RTCP scheduling after an RWT duration if the initially requested recovery picture does not arrive.

Receiver may report more losses or repeat messages if it deems necessary. As a minimum requirement on the receiver side, it shall support the capability of picture level error detection or tracking in order to stop reporting of prior losses from the recovery point. If FEC is supported, NACK messages should signal irrecoverable lost packets after recovery by FEC instead of the detected lost packets before recovery. The receiver should wait "repair-window" duration before issuing a NACK message for the missing packets.

When retransmission is supported, NACK messages correspond to requests for missing packets to be retransmitted instead of indication of error position for recovery without retransmission. The receiver should queue NACK messages for RTCP scheduling as necessary. The importance of lost packets along with possibility of timely arrival of requested packets should be considered before requesting retransmission.

Annex P.2 gives further description of receiver behaviour for error correction.

9.3.3 Sender behaviour

When NACK and PLI have been negotiated without retransmission support for the session then an MTSI client in terminal sending media:

- shall send a recovery picture or Gradual Decoder Refresh (GDR) upon receiving NACK message if loss indicated by the message corresponds to error in a reference picture within 500 ms. If a recovery picture corresponding to the NACK message was sent prior to reception of the NACK message by less than RWT duration, the sender does not have to respond to this particular NACK message.
- shall send an Instantaneous Decoder Refresh (IDR) or GDR picture upon receiving PLI message within 500 ms.
- should not respond to incoming NACK or PLI messages within RWT duration of the same message type indicating the same loss from the reception of the initial feedback message triggered by the onset of the loss.

IDR picture is an intra picture where pictures following the IDR picture can be decoded without referring (inter prediction) to pictures decoded prior to IDR picture. This corresponds to IDR pictures in H.264 and HEVC. GDR is performing intra refresh by distributing intra picture data over N pictures. At the end of N pictures from the start of GDR all macroblock regions are intra coded (refreshed) generating a good frame. Similar to IDR case, if intra picture or GDR is used as a recovery mechanism, the pictures following the intra picture or the GDR pictures should not reference pictures decoded prior to these pictures.

When retransmission is supported, sender should retransmit packets that it deems beneficial for timely recovery. Only source packets should be retransmitted. The minimum time the sender should keep an original RTP packet in its buffers available for retransmission, i.e. "rtx-time" value, should be RTT and the maximum time the sender should keep an original RTP packet should be 400 ms.

When FEC is supported, the amount of time that spans the source and the corresponding repair packets, i.e. "repair-window", should not be less than the minimum required duration for the bitmask used for generation of the repair packets. The maximum time that spans the source and corresponding repair packets should not be more than 300 ms.

An MTSI client in terminal sending video shall obey the rate restrictions imposed by the video rate adaptation specified in clause 10.3.

Annex P.2 gives further description of sender behaviour for error correction. Annex P.3 gives further description of sender and receiver behaviour for RTP retransmission based recovery.

9.3.4 Recommendations for packet loss recovery mechanisms usage

FEC should be used to provide robustness against moderate packet loss rates at high delay scenario. FEC can especially handle random losses and short burst losses and be beneficial in environments with high packet loss rates and/or high delay (RTT). The use of FEC may not be appropriate when packet losses are caused by insufficient throughput (over radio access or due to congestions in network) since it introduces some bit rate overhead. In order to compensate for bit rate overhead, FEC should be used with efficient rate adaptation mechanisms to reduce the source bit rate according to

channel conditions and not increase the total RTP bitrate. When error cases cannot be recovered by FEC, other mechanisms are needed in combination with FEC.

- Retransmission in combination with FEC should be used for the low RTT case with relatively high packet loss since retransmission can efficiently handle the FEC failure case.
- Generic NACK based recovery in combination with FEC should be used for high RTT, relatively high packet loss conditions since generic NACK based recovery does not introduce additional delay.

Selective retransmission should be used under low delay (RTT) and low failure (loss) rate conditions. Retransmission needs to ensure that retransmitted packets arrive in time to meet delay requirements of the end-to-end system. Higher packet loss rates may cause loss of retransmitted packets, hence leading to larger end-to-end delay.

Generic NACK and PLI based error correction mechanism should be used in combination with FEC or selective retransmission or under low packet loss rates with high RTT conditions. Generic NACK message can be used for indication of packets to be retransmitted as well as informing the sender of loss of particular RTP packets for sender to take necessary actions to recover from errors.

NOTE: Under unknown and varying conditions the MTSI client should dynamically select & adapt the proper mechanisms.

Additional information on the usage of these mechanisms is provided in [142].

9.4 Text

Redundant transmission provided by the RTP payload format as described in RFC 4103 [31] shall be supported. The transmitting application may use up to 200 % redundancy, i.e. a T140block transported in one RTP packet may be repeated once or twice in subsequent RTP packets. 200 % redundancy shall be used when the conditions along the call path are not known to be free of loss. However, the result of media negotiation shall be followed, and transmission without redundancy used if one of the parties does not show capability for redundancy.

The sampling time shall be 300 ms as a minimum (in order to keep the bandwidth down) and should not be longer than 500 ms. New text after an idle period shall be sent as soon as possible. The first packet after an idle-period shall have the M-bit set.

The procedure described in section 5 of RFC 4103 [31], or a procedure with equivalent or better performance, shall be used for packet-loss handling in the receiving MTSI client in terminal.

10 Adaptation

10.1 General

Adaptive mechanisms are used to optimize the session quality given the current transport characteristics. The mechanisms provided in MTSI are bit-rate, packet-rate and error resilience adaptation. These mechanisms can be used in different ways; however, they should only be used when the result of the adaptation is assumed to increase the session quality even if e.g. the source bit-rate is reduced.

Adaptive mechanisms that act upon measured or signalled changes in the transport channel characteristics may be used in a conservative manner. Examples of measured changes in transport characteristics are variations in Packet Loss Rate (PLR) and delay jitter. Examples of signalled changes in transport characteristics are ANBR (see clause 10.7) and ECN Congestion Experienced (ECN-CE) marking in IP packet headers. A conservative use of adaptation is characterized by a fast response to degrading conditions, and a slower, careful upwards adaptation intended to return the session media settings to the original default state of the session. The long-term goal of any adaptive mechanism is assumed to be a restoration of the session quality to the originally negotiated quality. The short-term goal is to maximize the session quality given the current transport characteristics, even if that means that the adapted state of the session will give a lower session quality compared to the session default state if transported on an undisturbed channel.

The 'a=bw-info' attribute defined in Clause 19 may be used to align the adaptation with the bearer setup and the end-to-end resource allocation. A few recommendations are described in sub-clause 10.6.

10.2 Speech

10.2.0 General

To reduce the risk for confusion in the media-sender, it is beneficial if the signaling from media-receiver back to media-sender for the media adaptation is the same regardless of which triggers are used in the adaptation-decision in the media-receiver. The ANBR described in clause 10.7 should, if supported by both the access network and the MTSI client in terminal, be used as one such trigger.

NOTE 1: The media-receiver is aware that other nodes in the media path may also influence the media adaptation. A media-receiver sending a specific CMR value X can expect that (after some time) no media is received with a mode higher than X, but modes lower than X may be received any time.

The adaptation for AMR, AMR-WB and EVS includes adapting the media bit-rate, the frame aggregation, the redundancy level and the redundancy offset. The domain of adaptation for EVS furthermore includes adapting audio bandwidth, partial redundancy, switching between EVS primary mode and EVS AMR-WB IO mode.

When the AMR codec or the AMR-WB codec is used, two signaling mechanisms are defined:

- CMR in the AMR/AMR-WB RTP payload, [28].
CMR in RTP can be used by the media-receiver to restrict the codec mode in the remote media-sender to an upper limit (maximum mode).
- RTCP-APP, see clause 10.2.1.
If the media-sender supports RTCP-APP, then the media-receiver can use it in the following way:
CMR in RTCP-APP can be used by the media-receiver to restrict the codec mode in the remote media-sender to an upper limit (maximum mode), in addition to CMR in RTP.
RTCP-APP can further be used by the media-receiver for the adaptation of frame aggregation, redundancy level and redundancy offset in the RTP packets to be sent by the remote media-sender.

When the EVS codec is used, the following signaling mechanism is defined:

- CMR in the EVS RTP payload, [125].
- RTCP-APP, see clause 10.2.1.

In response to received DL ANBR, a speech media receiver should trigger sending CMR requesting bitrate adaptation in the corresponding media sender RTP stream. If RTCP-APP is supported, then a speech media receiver should trigger sending CMR or RTCP-APP requesting bitrate adaptation in the corresponding media sender RTP stream based on the received DL ANBR.

When adapting frame aggregation and/or redundancy, the MTSI client must verify that the maximum packetization, defined by the maxptime SDP parameter, is not exceeded. The MTSI client must also verify that the IP packet sizes does not exceed the Maximum Transfer Unit (MTU).

The boundaries of the adaptation may be controlled by a set of parameters. These parameters may be configured into the MTSI client based on operator policy, for example using OMA-DM.

Table 10.1 defines a mandatory set of parameters that are used by the ECN triggered adaptation for AMR and AMR-WB. The default values for the parameters are also specified. Alternate values for these parameters may be configured into the MTSI client based on operator policy, for example using OMA-DM.

Table 10.1: Configuration parameters when ECN is used as a trigger

| Parameter | Description |
|--------------|---|
| ECN_min_rate | Lower boundary for the media bit-rate adaptation in response to ECN-CE marking. The media bit-rate shall not be reduced below this value as a reaction to the received ECN-CE. The ECN_min_rate should be selected to maintain an acceptable service quality while reducing the resource utilization. Default value: For AMR and AMR-WB, the default value shall be the rate of the recommended Initial Codec Mode, see Clause 7.5.2.1.6. |

| | |
|---------------------|--|
| ECN_congestion_wait | The waiting time after an ECN-CE marking for which an up-switch shall not be attempted. A negative value indicates an infinite waiting time, i.e. to prevent up-switch for the whole remaining session. Default value: 5 seconds |
|---------------------|--|

The configuration of adaptation parameters, and the actions taken during the adaptation, are specific to the particular triggers. For example, the adaptation may be configured to reduce the media bit-rate to AMR5.9 when ECN-CE is detected, while it may reduce the media bit-rate to AMR4.75 for bad radio conditions when high PLR is detected.

Multiple ECN-CE markings within one RTP-level round-trip time is considered as the same congestion event. Each time an MTSI client detects a congestion event it shall send an adaptation request to reduce the media bit-rate unless already operating at the ECN_min_rate or below. An MTSI client detecting a congestion event shall not send an adaptation request to increase the media bit-rate for a time period ECN_congestion_wait after the end of the congestion event.

Multiple adaptation trigger algorithms can be used in parallel, for example ECN-triggered adaptation, adaptation based on ANBR, and PLR-triggered adaptation. When multiple adaptation algorithms are used for the rate adaptation, the rate that the MTSI client is allowed to use should be no higher than any of the rates determined by each adaptation algorithm.

NOTE 2: For example, if the ECN-triggered adaptation indicates that AMR5.9 should be used and if the PLR-triggered adaptation indicates that AMR4.75 should be used then the rate that the MTSI client uses should be no higher than $\min(\text{AMR5.9}, \text{AMR4.75}) = \text{AMR4.75}$. An example adaptation scheme is described in Annex C.

When additional transport bandwidth information is provided using the 'a=bw-info' attribute defined in clause 19, the Minimum Desired Bandwidth should be aligned with the ECN_min_rate configuration parameter.

10.2.1 RTCP-APP with codec control requests

10.2.1.1 General

When signalling adaptation requests for speech in MTSI, an RTCP-APP packet should be used. This application-specific packet format supports three different adaptation requests when the AMR or AMR-WB codec is used; bit-rate requests, frame aggregation requests and redundancy requests. The requests for frame aggregation and redundancy are also used when the EVS codec is used. The codec mode request used for AMR-WB is also used when the EVS AMR-WB IO mode is used. The application specific format supports additionally five requests that are used for the EVS codec. The RTCP-APP packet is put in a compound RTCP packets according to the rules outlined in RFC 3550 [9] and RFC 4585 [40]. In order to keep the size of the RTCP packets as small as possible it is strongly recommended that the RTCP packets are transmitted as minimal compound RTCP packets, meaning that they contain only the items:

- SR or RR;
- SDES CNAME item;
- APP (when applicable).

The recommended RTCP mode is RTCP-AVPF early mode since it will enable transmission of RTCP reports when needed and still comply with RTCP bandwidth rules. The RTCP-APP packets should not be transmitted in each RTCP packet, but rather as a result in the transport characteristics which require end-point adaptation.

The signalling allows for a request that the other endpoint modifies the packet stream to better fit the characteristics of the current transport link. The request in the received RTCP-APP is valid until a new request is received. Note that the media sender can, if having good reasons, choose to not comply with the request received from the media receiver. One such reason could be knowledge of that the local conditions do not allow the requested format.

10.2.1.2 General Format of RTCP-APP packet with codec control requests

The RTCP-APP packet defined to be used for adaptation signalling for speech in MTSI is constructed as shown in figure 10.1.

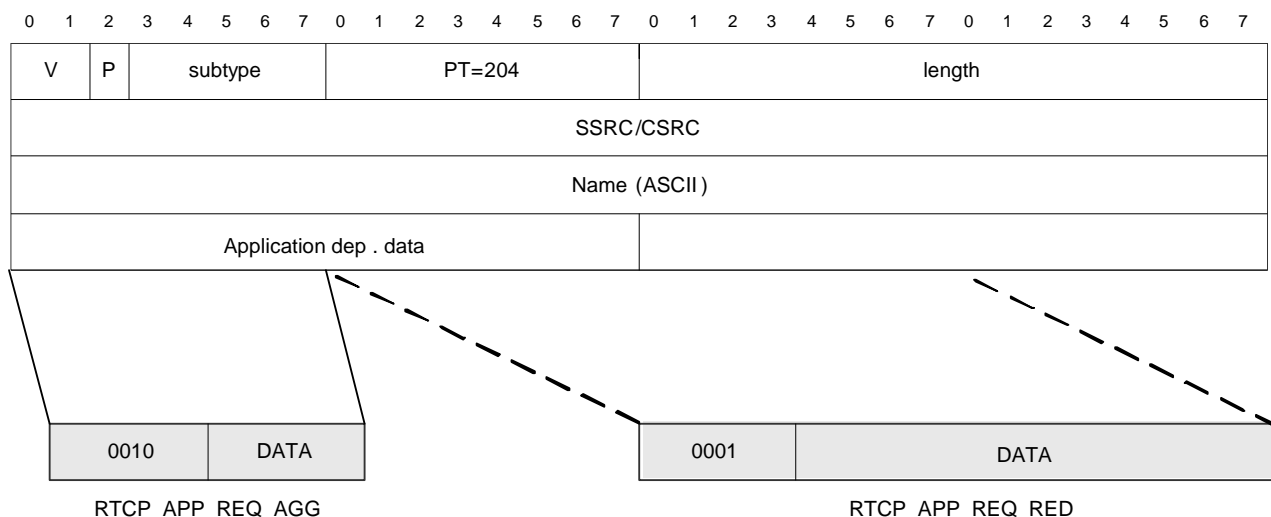


Figure 10.1: RTCP-APP formatting

The RTCP-APP specific fields are defined as follows:

- Subtype - the subtype value shall be set to "0".
- Name - the name shall be set to "3GM7", meaning 3GPP MTSI Release 7.

The application-dependent data field contains the requests listed below. The length of the application-dependent data shall be a multiple of 32 bits. The unused bytes shall be set to zero.

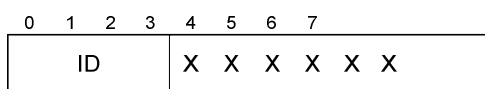


Figure 10.2: Basic syntax of the application-dependent data fields

The length of the messages is 1 or 2 bytes depending on request type.

The ID field identifies the request type. ID Code points [0000] ... [1000] are specified in the present document, whereas the other ID code points are reserved for future use.

The signalling for three different adaptation requests is defined below. For each request, the codecs that can use the request are also specified.

The requests that can be used in a session are negotiated with SDP, see clause 10.2.3.

10.2.1.2a Padding

PADDING: This message contains no request but is identical to a padding byte with all zeroes and is therefore used as padding.

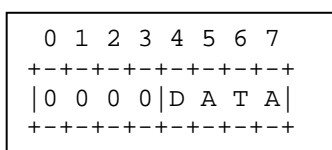


Figure 10.2a: Padding

Codecs: This request can be used for all codecs.

The DATA field is a 4-bit value field with all bits set to zero. When receiving a PADDING message, the whole octet shall be ignored, regardless of the bits in the DATA field.

An MTSI client uses this to pad the RTCP-APP to be 32 bit aligned when needed.

10.2.1.3 Redundancy Request

RTCP_APP_REQ_RED: Request for redundancy level and offset of redundant data.

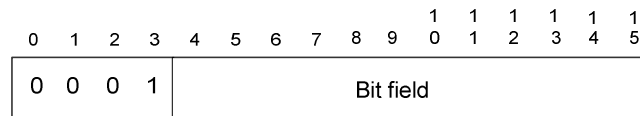


Figure 10.3: Redundancy request

Codecs: This request can be used for all codecs.

The Bit field is a 12 bit bitmask that signals a request on how non-redundant payloads chunks are to be repeated in subsequent packets.

The position of the bit set indicates which earlier non-redundant payload chunks is requested to be added as redundant payload chunks to the current packet.

- If the LSB (rightmost bit) is set equal to 1 it indicates that the last previous payload chunk is requested to be repeated as redundant payload in the current packet.
- If the MSB (leftmost bit) is set equal to 1 it indicates that the payload chunk that was transmitted 12 packets ago is requested to be repeated as redundant payload chunk in the current packet. Note that it is not guaranteed that the sender has access to such old payload chunks.

The maximum amount of redundancy is 300 %, i.e., at maximum three bits can be set in the Bit field.

See clause 10.2.1 for example use cases.

10.2.1.4 Frame Aggregation Request

RTCP_APP_REQ_AGG: Request for a change of frame aggregation.

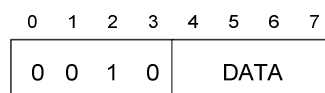


Figure 10.4: Frame aggregation request

Codecs: This request can be used for all codecs.

The DATA field is a 4 bit value field:

- 0000 - 1 frame / packet.
- 0001 - 2 frames / packet.
- 0010 - 3 frames / packet.
- 0011 - 4 frames / packet.

The values 0100...1111 are reserved for future use.

The maximum allowed frame aggregation is also limited by the maxptime parameter in the session SDP since the sender is not allowed to send more frames in an RTP packet than what the maxptime parameter defines.

The default aggregation is governed by the *ptime* parameter in the session SDP. It is allowed to send fewer frames in an RTP packet, for example if there are no more frames available at the end of a talk spurt. It is also allowed to send more frames in an RTP packet, but such behaviour is not recommended.

See clauses 7.4.2 and 12.3.2.1 for further information.

10.2.1.5 Codec Mode Request

RTCP_APP_CMR: Codec Mode Request

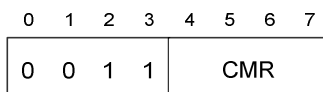


Figure 10.5: Codec mode request

Codecs: This request can only be used for the AMR codec, the AMR-WB codecs and for the EVS codec when operating in AMR-WB IO mode.

The definition of the CMR bits in the RTCP_APP_CMR message is identical to the definition of the CMR bits defined in [28]. The CMR indicates the maximum codec mode (highest bit-rate) that the receiver wants to receive. The sender may very well use a lower codec mode (lower bit-rate) when sending.

An MTSI client in terminal that requests mode adaptation should transmit the CMR in an RTCP_APP_CMR, unless specified otherwise in Clause 7.3.2.

When the MTSI MGW has an interworking session with a circuit-switched (CS) system using transcoding and requests mode adaptation, the MTSI MGW should transmit CMR in an RTCP_APP_CMR, unless specified otherwise in Clause 7.3.2, and should set the CMR in the AMR payload to 15 (no mode request present [28]).

When the MTSI MGW has an interworking session with a circuit-switched (CS) system using TFO/TrFO, then the MTSI media gateway should translate the CMR bits (in GERAN case) or the Iu/Nb rate control messages (in UTRAN case) from the CS client into the CMR bits in the AMR payload. If the MTSI media gateway prefers to receive a lower codec mode rate from the MTSI client in terminal than what the CMR from the CS side indicates, then the MTSI media gateway may replace the CMR from the CS side with the CMR that the MTSI media gateway prefers. The value 15 (no mode request present [28]) shall be used in the CMR bits in the AMR payload towards the PS side if on the CS side no mode request has been received and if the MTSI media gateway has no preference on the used codec mode. The RTCP_APP_CMR should not be used in the direction from the MTSI media gateway towards the MTSI client when TFO/TrFO is used.

If an MTSI client receives CMR bits both in the AMR payload and in an RTCP_APP_CMR message, the mode with the lowest bit rate of the two indicated modes should be used. A codec mode request received in a RTCP_APP_CMR is valid until the next received RTCP_APP_CMR.

10.2.1.6 Generation of RTP payloads for multiple codec control requests

Figure 10.6 below illustrates how the three requests are used by the transmitter. In this case, RTCP_APP_REQ_RED is equal to "000000000101".

- The speech encoder generates frames every 20 ms.
- The speech frames are buffered in the aggregation buffer until it is possible to generate a payload chunk with the number of frames requested by either *ptime* at session setup or by RTCP_APP_REQ_AGG during a session.
- The current payload chunk is used when constructing the current RTP packet.
- The history buffer contains previously transmitted payload chunks. The length of this buffer needs to be dimensioned to store the maximum number of payload chunks that are possible. This value is based on the *max-red* value, the *maxptime* values and from the minimum number of frames that the transmitter will encapsulate in the RTP packets. In this case, the buffer length is selected to 11 payload chunks since this corresponds to the worst case of *max-red*=220, *maxptime*=240 and one frame per payload chunk.

- After transmitting the current RTP packet, the content of the history buffer is shifted, the current payload chunk is shifted in to the history buffer as P(n-1) and the oldest payload chunk P(n-11) is shifted out.
- When constructing the (provisional) RTP payload, the selected preceding payload chunks are selected from the history buffer and added to the current payload chunk. In order to form a valid RTP payload, the transmitter needs to verify that the maxptime value is not exceeded. If the provisional RTP payload is longer than what maxptime allows, then the oldest speech frames shall be removed until the length (in time) of the payload no longer violates the maxptime value. NO_DATA frames in the beginning or at the end of the payload does not need to be transmitted and are therefore removed. The RTP Time Stamp needs to be incremented when a NO_DATA frames are removed from the beginning of the payload. A (provisional) RTP packet containing only NO_DATA frames does not need to be transmitted.

Note also that the transmitter is not allowed to send frames that are older than the max-red value that the transmitter has indicated in the SDP.

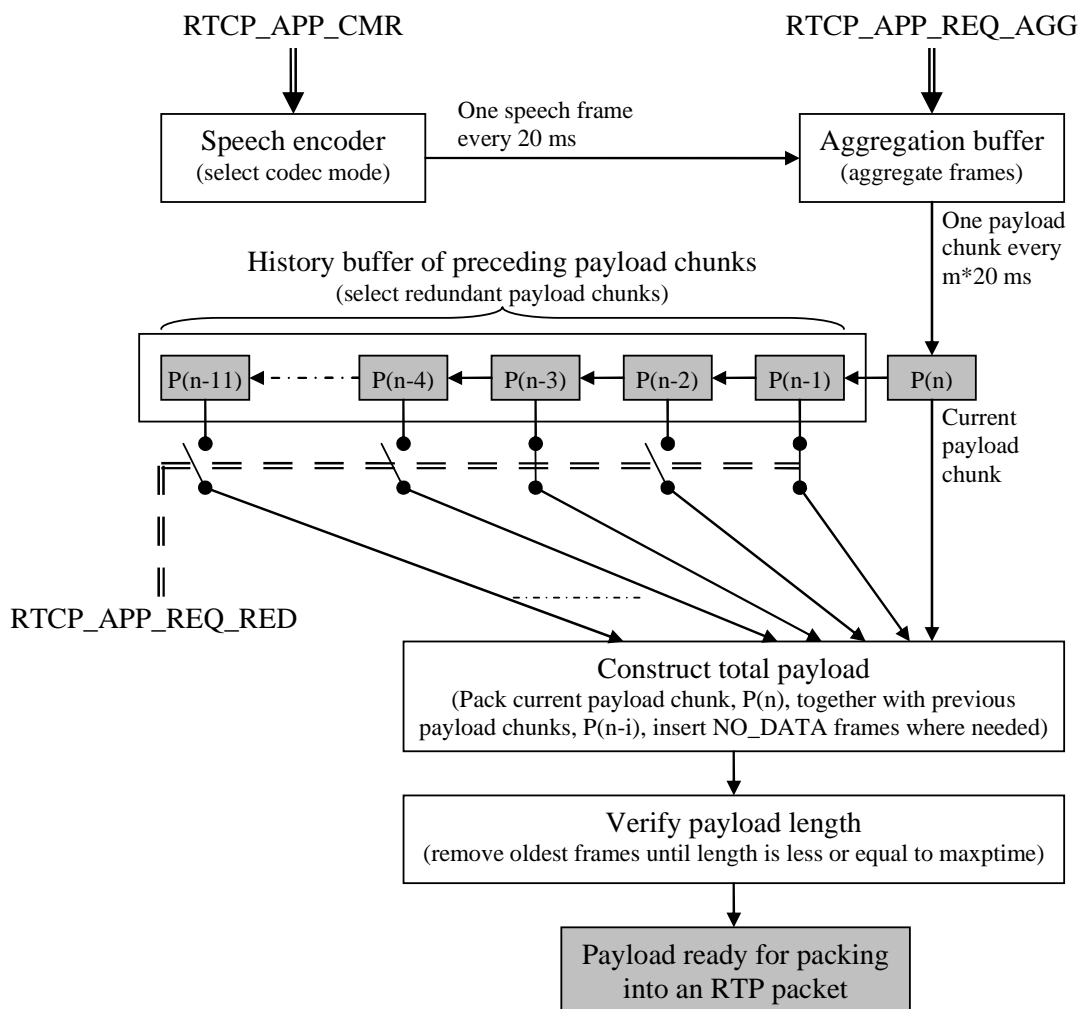
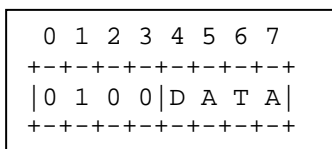


Figure 10.6: Visualization of how the different adaptation requests affect the encoding and the payload packetization

It should be noted that RTCP_APP_REQ_AGG and RTCP_APP_REQ_RED are independent. Furthermore, it should also be noted that different redundant payload chunks may contain different number of speech frames.

10.2.1.7 EVS Primary Rate Request

RTCP_APP_REQ_EPRR: EVS Primary Rate Request

**Figure 10.6a: EVS primary rate request**

Codecs: This request can be used for the EVS codecs when operating in EVS Primary mode.

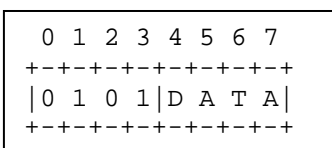
The DATA field a 4-bit field and is encoded as described in the table below. The rate request indicates the maximum codec mode (highest bit-rate) that the receiver wants to receive. The sender may use a lower codec mode (lower bit-rate) when sending. The rate request shall comply with the media type parameters that are negotiated in the session.

Table 10.1a Encoding of the DATA field in the EVS Primary Rate Request.

| Index | EVS Primary rate request |
|-------|--------------------------|
| 0000 | 5.9 |
| 0001 | 7.2 |
| 0010 | 8 |
| 0011 | 9.6 |
| 0100 | 13.2 |
| 0101 | 16.4 |
| 0110 | 24.4 |
| 0111 | 32 |
| 1000 | 48 |
| 1001 | 64 |
| 1010 | 96 |
| 1011 | 128 |
| 1100 | Not used |
| 1101 | Not used |
| 1110 | Not used |
| 1111 | Not used |

10.2.1.8 EVS Bandwidth Request

RTCP_APP_REQ_EBWR: EVS Bandwidth Request

**Figure 10.6b: EVS bandwidth request**

Codecs: This request can be used for the EVS codecs when operating in Primary mode.

The DATA field is a 4-bit field b0...b3, corresponding to bit 4 to bit 7 in the octet:

- b0 set to '1' = request for narrowband.
- b1 set to '1' = request for wideband.
- b2 set to '1' = request for super-wideband.
- b3 set to '1' = request for fullband.

Each bit in the DATA field indicates a bandwidth that the receiver wants to receive. One or several of these four bits can be set to '1'. For example, a request for '1110' indicates that the receiver wants to receive narrowband, wideband or super-wideband speech but not fullband speech. The bandwidth request shall comply with the media type parameters that are negotiated in the session.

- b5 = '0': AMR-WB 8.85 not allowed
b5 = '1': AMR-WB 8.85 allowed,
- b6 = '0': AMR-WB 12.65 not allowed
b6 = '1': AMR-WB 12.65 allowed,
- b7 = '0': AMR-WB 14.25 not allowed
b7 = '1': AMR-WB 14.25 allowed,
- b8 = '0': AMR-WB 15.85 not allowed
b8 = '1': AMR-WB 15.85 allowed,
- b9 = '0': AMR-WB 18.25 not allowed
b9 = '1': AMR-WB 18.25 allowed,
- b10 = '0': AMR-WB 19.85 not allowed
b10 = '1': AMR-WB 19.85 allowed,
- b11 = '0': AMR-WB 23.05 not allowed
b11 = '1': AMR-WB 23.05 allowed,
- b12 = '0': AMR-WB 23.85 not allowed
b12 = '1': AMR-WB 23.85 allowed.
- flags:
 - b13 = '0': mode-change-period=1,
b13 = '1': mode-change-period=2,
 - b14 = '0': mode-change-neighbor=0,
b14 = '1': mode-change-neighbor=1.

An MTSI client sending this request shall set at least one of the mode-set bits to '1'. An MTSI client receiving a request with all zeroes shall ignore the request.

The mode-set indicated in the EVS Primary mode to EVS AMR-WB IO mode Switching Request can only allow codec modes that have been negotiated in SDP offer-answer. This request cannot be used to allow codec modes that have not been negotiated in SDP offer-answer.

An MTSI client sending this request should also send an RTCP_APP_CMN to indicate the codec mode that should be used after switching to EVS AMR-WB IO mode. An MTSI client receiving this request without a request for a codec mode should use the rules for Initial Codec Mode (ICM) defined in clause 7.5.2.1.6 to determine the codec mode that should be used after switching to EVS AMR-WB IO mode.

The last bit (b15) 'R' is reserved for future use. An MTSI client sending this request shall set it to '0'. An MTSI client receiving this request shall ignore this bit.

10.2.1.11 EVS AMR-WB IO mode to EVS Primary mode Switching Request

RTCP_APP_REQ_EI2P: EVS AMR-WB IO mode to EVS Primary mode Switching Request

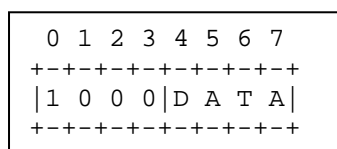


Figure 10.6f: EVS AMR-WB IO mode to EVS Primary mode Switching request

Codecs: This request can be used for the EVS codecs when operating in AMR-WB IO mode.

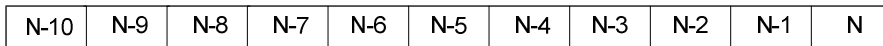
The DATA field is a 4-bit field which is reserved for future use. All four bits are set to '0'.

The bitrates and bandwidths that can be used after switching to EVS Primary mode are the same as negotiated at session setup or in a preceding session modification.

10.2.2 Example use cases

The following examples demonstrate how requests for redundancy and frame aggregation are realised in the RTP stream.

All examples assume that the speech codec generates frames numbered N-10...N in a continuous flow.



**Figure 10.7: Flow of parameter sets for encoded frames
Each increment corresponds to a time difference of 20 ms**

In the examples below, P-1...P denote the sequence numbers of the packets.

EXAMPLE 1:

An RTCP_APP_REQ_RED request with bit field 000000000000 (no redundancy) and RTCP_APP_REQ_AGG request with value = 0 (no frame aggregation) will yield packets as shown in figure 10.8.

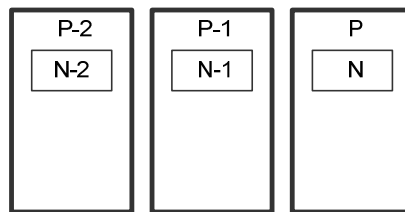


Figure 10.8: Default frame aggregation with one frame per packet

EXAMPLE 2:

An RTCP_APP_REQ_RED request with bit field 000000000001 (100% redundancy and no offset) and an RTCP_APP_REQ_AGG request with value = 0 (no frame aggregation) will yield packets as shown in figure 10.9.

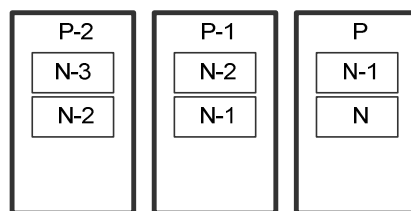


Figure 10.9: Payload packetization with 100 % redundancy and an offset of one packet

EXAMPLE 3:

An RTCP_APP_REQ_RED request with bit field 000000000010 (100% redundancy with offset 1 extra packet) and an RTCP_APP_REQ_AGG request with value = 0 (no frame aggregation) will yield packets as shown in figure 10.10.

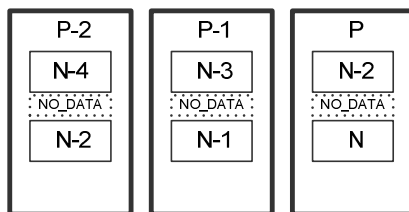


Figure 10.10: Payload packetization with 100 % redundancy and an extra offset of one packet

NO_DATA frames must be inserted to fill the gaps between two non-consecutive frames, e.g. between N-2 and N.

EXAMPLE 4:

An RTCP_APP_REQ_RED request with bit field 000000000000 (no redundancy) and RTCP_APP_REQ_AGG request with value = 1 (frame aggregation 2 frames/packet) will yield packets as shown in figure 10.11.

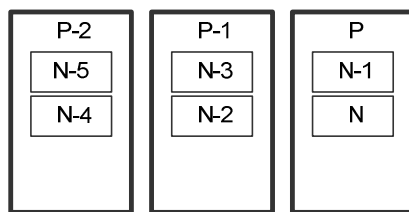


Figure 10.11: Payload packetization with 2 frames aggregated per packet

EXAMPLE 5:

An RTCP_APP_REQ_RED request with bit field 000000000001 (100% redundancy) and an RTCP_APP_REQ_AGG request with value = 1 (frame aggregation 2 frames/packet) will yield packets as shown in figure 10.12.

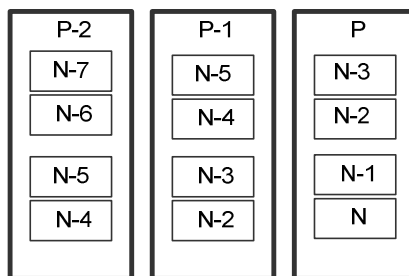


Figure 10.12: Payload packetization with 100 % redundancy and 2 frames aggregated per packet

EXAMPLE 6:

An RTCP_APP_REQ_RED request with bit field 000000000010 (100% redundancy with offset 1 extra packet) and an RTCP_APP_REQ_AGG request with value = 1 (frame aggregation 2 frames/packet) will yield packets as shown in figure 10.13.

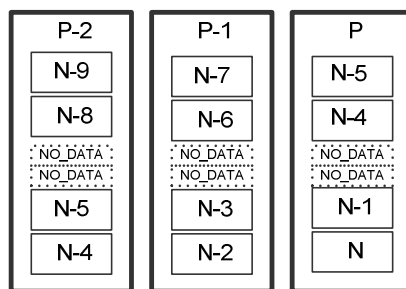


Figure 10.13: Payload packetization with 100 % redundancy, one extra offset and 2 frames aggregated per packet

10.2.3 SDP negotiation for RTCP-APP

RTCP-APP request messages that can be used are negotiated with SDP using the '3gpp_mtsi_app_adapt' attribute. The syntax for the 3GPP MTSI RTCP-APP adaptation attribute is:

```
a=3gpp_mtsi_app_adapt:<reqNames>
```

where:

<reqNames> is a comma-separated list identifying the different request messages (see below).

The ABNF for the RTCP-APP adaptation messages negotiation attribute is the following:

```
adaptation attribute = "a" "=" "3gpp_mtsi_app_adapt" ":" reqName *("," reqName)
```

```
reqName = "RedReq" / "FrameAggReq" / "AmrCmr" / "EvsRateReq" / "EvsBandwidthReq" /
"EvsParRedReq" / "EvsIoModeReq" / "EvsPrimaryModeReq"
```

The name denotes the RTCP APP packet types the SDP sender supports to receive. The meaning of the values is as follows:

RedReq: Redundancy Request, clause 10.2.1.3

FrameAggReq: Frame Aggregation Request, clause 10.2.1.4

AmrCmr: Codec Mode Request for AMR and AMR-WB, clause 10.2.1.5

EvsRateReq: EVS Primary Rate Request, clause 10.2.1.7

EvsBandwidthReq: EVS Bandwidth Request, clause 10.2.1.8

EvsParRedReq: EVS Partial Redundancy Request, clause 10.2.1.9

EvsIoModeReq: EVS Primary mode to EVS AMR-WB IO mode Switching Request, clause 10.2.1.10

EvsPrimaryModeReq: EVS AMR-WB IO mode to EVS Primary mode Switching Request, clause 10.2.1.11

An MTSI client supporting the reception of any RTCP APP packets defined in the present specification shall indicate the supported RTCP APP packet types in an initial SDP offer or answer it sends using the SDP

"a=3gpp_mtsi_app_adapt" attribute. If the answerer receives an "a=3gpp_mtsi_app_adapt" attribute in the SDP offer, it may send the indicated RTCP APP packet types towards the offerer. The answerer shall indicate its capabilities with the "a=3gpp_mtsi_app_adapt" attribute irrespective if an "a=3gpp_mtsi_app_adapt" attribute was received and the capabilities within. If the offerer receives an "a=3gpp_mtsi_app_adapt" attribute in the SDP answer, it may send the indicated RTCP APP packet types towards the answerer.

An MTSI client supporting only AMR and AMR-WB therefore may for instance include the following in the SDP offer:

```
a=3gpp_mtsi_app_adapt: RedReq,FrameAggReq,AmrCmr
```

An MTSI client supporting only AMR, AMR-WB and EVS may for instance include the following in the SDP offer:

```
a=3gpp_mtsi_app_adapt:  
RedReq,FrameAggReq,AmrCmr,EvsRateReq,EvsBandwidthReq,EvsParRedReq,EvsIoModeReq,EvsPrimaryM  
odeReq
```

The attribute shall only be used on media level.

When interworking with pre-Rel-12 clients or non-MTSI clients, it may happen that they support the RTCP-APP signalling but not the SDP negotiation for AMR and AMR-WB. An MTSI client failing to negotiate RTCP-APP as described may still try to use the RTCP-APP signalling when requesting adaptation, but the MTSI client shall then also monitor the received media in order to determine if some or all of the adaptation requests included in the RTCP-APP were partially or fully followed or not followed at all. If none of the adaptation requests is followed, not even partially, then this is an indication that the remote client does not support the RTCP-APP signalling. The MTSI client should then try to use other means for triggering the adaptation, for example CMR in the AMR/AMR-WB payload or RTCP Sender Reports/Receiver Reports.

10.3 Video

10.3.1 General

MTSI clients receiving RTCP Receiver Reports (RR) indicating nonzero packet loss shall support adjusting their outgoing bitrate accordingly (see RFC 3550 [9]). Note that for IMS networks, which normally have nonzero packet loss and fairly long round-trip delay, the amount of bitrate reduction specified in RFC 3448 [56] is generally too restrictive for video and may, if used as specified, result in very low video bitrates already at (for IMS) moderate packet loss rates.

A video sender shall support adapting its video output rate based on RTCP reports and TMMBR messages. This adaptation shall be used as described in clauses 10.3.2 to 10.3.6 unless the video sender is explicitly notified that no rate adaptation shall be performed, e.g. by setting the minimum quality bitrate equal to the negotiated bitrate. This adaptation should be performed while maintaining a balance between spatial quality and temporal resolution, which matches the bitrate and image size. Some examples are given in Annex B. For the handling of packet loss signaled through AVPF NACK and PLL, or for rate adaptation with RTCP reports and TMMBR messages, the video sender shall be able to dynamically adapt to the reported conditions, in particular to facilitate the operation of quality-recovery techniques pertinent to the situations. Quality-recovery techniques include, but may not be limited to, adapted intra frame periods, adaptation of random intra macroblock refresh ratios, FEC, and adaptation of the bit rates.

The rate adaptation can be controlled by using the video adaptation parameters defined in clause 17.2. By using the MIN_QUALITY/BIT_RATE/ABSOLUTE or the MIN_QUALITY/BIT_RATE/RELATIVE parameters it is possible to set the minimum bitrate for the adaptation.

10.3.2 Signaling mechanisms

The use of TMMBR and TMMBN depends on the outcome of the SDP offer/answer negotiation, see Clause 6.2.3.2.

If TMMBR and TMMBN are allowed to be used in the session and if the receiving MTSI client in terminal is made aware of a reduction in downlink bandwidth allocation through an explicit indication of the available bandwidth allocation from the network (e.g. due to QoS renegotiation or handoff to another radio access technology), or from measurements such as increased delay at the media receiver, it shall notify the media sender of the new current maximum bitrate using TMMBR. In this context the TMMBR message is used to quickly signal to the other party a reduction in available transport bitrate. If rate adaptation is allowed, the media sending MTSI client shall, after receiving TMMBR, adjust the sent media rate to the requested rate or lower and shall respond by sending TMMBN, as described in CCM [43]. When determining the encoder bitrate the MTSI client needs to compensate for the IP/UDP/RTP overhead since the bitrate indicated in the TMMBR message includes this overhead. To determine TMMBR and TMMBN content, both media sending and media receiving MTSI clients in terminals shall use their best estimates of packet measured overhead size when measured overhead values are not available. If the TMMBR message was sent due to an explicit indication of available bandwidth allocation, the MTSI client in terminal that sent the TMMBR message shall, after receiving the TMMBN, send a SIP UPDATE to the other party to establish the new rate as specified in clause 6.2.7.

It is the media sender's responsibility to estimate if, and by how much, queue build-up has occurred due to use of a sending rate that was higher than the available throughput, before being able to reduce the sending rate. It is therefore

also the media sender's responsibility to recover the buffering delay by sending with a rate that is lower than what the media receiver has requested in the TMMBR message for some period of time.

If TMMBR and TMMBN are not allowed to be used in the session and if the MTSI client in terminal is made aware of a reduction in downlink bandwidth allocation (e.g. due to QoS renegotiation or handoff to another radio access technology) it shall send a SIP UPDATE to the other party to establish the new rate as specified in clause 6.2.7.

If the receiving MTSI client in terminal is made aware of an increase in downlink bandwidth allocation (determined via separate negotiation) through an explicit indication from the network (e.g. due to QoS renegotiation or handoff to another radio access technology) then, if this has not yet occurred, it shall send a SIP UPDATE to the other party to establish the new rate as specified in clause 6.2.7.

When an MTSI client in terminal receives available bandwidth information from ANBR (see 10.7), it shall not be considered an explicit indication of available bandwidth allocation that requires sending SIP UPDATE as described above. The conditions under which it is allowed to send SIP UPDATE based on ANBR are described in clause 6.2.5.1.

The media sender information in the RTCP Sender Reports (RTCP SR) contains information about how many packets and how much data the media sender has sent. A media receiving MTSI client in terminal may use this information to detect the difference between the sent bitrate (from the remote media sending client) and the received bitrate (in the local media receiving client).

The report blocks in the RTCP Receiver Reports (RTCP RR) or in the RTCP Sender Reports (RTCP SR), contain information about the highest received sequence number, the packet loss rate, the cumulative number of packet losses and interarrival jitter as experienced by the media receiver. A media sending MTSI client in terminal may use this information to detect the difference between the sent bitrate (from the local media sending client) and the received bitrate (in the remote media receiving client) and also to estimate the queue build-up that can happen when congestion occurs somewhere in the path.

To enable proper video rate adaptation, RTCP Reports must be sent frequently enough (e.g. at least twice per second) to allow MTSI clients to detect network congestion. An MTSI client in terminal shall set the RR and RS bandwidth modifiers in the SDP offer/answer to reserve an RTCP bandwidth that is no smaller than the bandwidth reserved by setting the RTCP bandwidth modifiers as follows (see Annex A.6):

- 0 bps for the RS field (at media level);
- 5000 bps for the RR field (at media level).

NOTE 1: In a point-to-point session the MTSI clients in terminals will be reserved, on average, 2500 bps of RTCP bandwidth in each direction when the RTCP bandwidth modifiers are negotiated as described above.

Furthermore, unless there is a clear need to set the RTCP bandwidth higher than specified above, the RTCP bandwidth modifiers in the SDP offer should be set as specified above.

NOTE 2: RFC 3550 recommends that the RTCP bandwidth default be 5% of the media bandwidth. However, this default may be excessive in various scenarios, including 3GPP access networks, and should therefore be carefully evaluated when setting the RR and RS values differently than recommended in this clause.

Another way to estimate the transmitted bitrate is to analyse the size of the packets and the RTP time stamps.

10.3.3 Adaptation triggers

An MTSI client in terminal sending or receiving media needs to know the currently allowed bitrate (*currently_allowed_bitrate*). The currently allowed bitrate is the minimum of the bitrate negotiated in SDP offer/answer and the bitrate allowed after the latest preceding adaptation (e.g. last previous TMMBR message) that increased or decreased the allowed bitrate for the encoder. When no bitrate reduction trigger is received, the value from SDP offer/answer shall be used. The currently allowed bitrate may therefore vary over time.

An MTSI client in terminal sending media shall use at least one adaptation trigger that is based on the reception report blocks in the received RTCP Receiver Reports or in the RTCP Sender Reports. An MTSI client in terminal sending media should also use ANBR as an adaptation trigger.

NOTE: When interworking with non-MTSI clients then it may happen that the remote client only sends RTCP Receiver Reports (or Sender Reports) but does not use any adaptation triggers in its receiver. This may happen even if the remote client supports and uses TMMBR because it is possible that the remote client uses TMMBR only to signal bitrate changes due to handoff to another access and not for dynamic rate adaptation.

An MTSI client in terminal receiving media shall use at least one adaptation trigger that is not ECN. Examples of adaptation triggers are: ANBR, measurements of packet loss rate; measurements of jitter; difference between sending bitrate (e.g. from RTCP SR) and measured received bitrate; differences between sending packet rate (from RTCP SR) and received packet rate; and play-out delay margin (from packet arrival time until their scheduled play-out time).

An MTSI client in terminal sending or receiving media:

- Should use one or more triggers that is capable to detect a needed reduction in throughput of 10% or more. If a trigger requires reception of an RTCP Sender or Receiver Report, the change should be detected within 3 frame durations of reception of the Sender or Receiver Report. If all triggers do not require Sender or Receiver Report reception, the change should be detected within 8 frame durations of the reduction in throughput.
- Shall use one or more triggers that is capable to detect a needed reduction in throughput of 25% or more. If a trigger requires reception of an RTCP Sender or Receiver Report, the change shall be detected within 6 frame durations of reception of the Sender or Receiver Report. If all triggers do not require Sender or Receiver Report reception, the change shall be detected within 15 frame durations of the reduction in throughput.

If an MTSI client in terminal receiving media uses DL ANBR to detect a needed reduction in throughput of 10% or more, it should send RTCP TMMBR to request the highest bitrate that is lower than ANBR and all other adaptation triggers.

An MTSI client in terminal sending media should take UL ANBR into account and adapt the sent bitrate to the highest bitrate that is still lower than or equal to the minimum of the adaptation triggers, and should then send RTCP TMMBN based on this bitrate.

An MTSI client in terminal receiving media shall use at least one method to estimate if and by how much the bitrate can be increased (*rate_increase_step*). A method for how an MTSI client in terminal can estimate when and by how much the bitrate can be increased is described in Annex C.2.5.

10.3.4 Sender behavior, downswitching

10.3.4.1 Downswitching divided into phases

The downswitching of the encoder bitrate in response to received adaptation requests or performance metrics is divided into two phases:

- First a rate reduction phase, where the bitrate is reduced from the target bitrate currently used by the sender, which is too high for the current operating conditions, to the bitrate that is suitable for the current operating conditions.
- Then a delay recovery phase, where the delay of any buffered data is recovered.

These phases are described in more detail below.

Annex C.2 gives a further description of the downswitching procedure.

10.3.4.2 Rate reduction phase

An MTSI client in terminal sending media should be able to immediately change the sending bitrate to the bitrate requested in a received TMMBR message.

Due to differences in client implementations (video encoder, cameras, etc), a sending MTSI client in terminal may or may not be able to immediately change the sending bitrate to the bitrate requested in a TMMBR message. The capability to immediately change the bitrate may also depend on whether the bitrate adaptation requires changing the frame rate and/or the video resolution.

When a reduction of the bitrate is requested with TMMBR and the MTSI client in terminal cannot immediately adapt to the requested bitrate then this will introduce excessive bits (*excess_bits*) since the sending bitrate will be higher than the available bitrate. These excessive bits will cause buffering, packet delays and sometimes packet losses. In this case, the MTSI client in terminal shall calculate the amount of excessive bits that are created until the bitrate has been reduced to the requested bitrate. In this case, the sending MTSI client in terminal:

- should adapt the encoding bitrate such that:

$$excess_bits \leq 0.5 * excess_bits_wc \quad (10.3.4.2-1)$$

- shall adapt the encoding bitrate such that:

$$excess_bits \leq excess_bits_wc \quad (10.3.4.2-2)$$

where:

$$excess_bits_wc = adapt_time_wc * (prev_rate - new_rate) \quad (10.3.4.2-3)$$

and: *adapt_time_wc* is the adaptation time required by the Worst-Case Adaptation Algorithm, see Annex C.2.4, in this case 1 second;

prev_rate is the bit-rate used before the adaptation starts;

new_rate is the bit-rate requested in the TMMBR message.

The *excess_bits* is calculated over the *measurement_window*, which is from the time when the TMMBR message is received until encoder has adapted down to the *new_rate*, see also Annex C.2.4. The bitrate used by the encoder is expected to vary from frame to frame. The bitrate should therefore be averaged using a sliding window over at least the last 5 frame durations before comparing it to the *new_rate*.

An MTSI client in terminal reducing the bitrate:

- should have adapted down to *new_rate* $1.5 * adapt_time_wc$ after the TMMBR message was received,
- shall have adapted down to *new_rate* $2 * adapt_time_wc$ after the TMMBR message was received.

The above procedure applies only when a bitrate reduction is requested with a TMMBR message. When the bitrate is increased, after the congestion has been cleared, then the above procedure does not apply.

Annex C.2.4 gives a further description of the above requirements and recommendations and how the encoder should behave during the rate reducing phase.

10.3.4.3 Delay recovery phase

After adapting down to the requested bit-rate the sending MTSI client in terminal shall use a delay recovery phase where the bit-rate is (on average) lower than the requested bit-rate until the buffering delay caused by the excessive bits (*excess_bits*) described in clause 10.3.4.2 have been recovered, see also Annex C.2.4 and C.2.6.

10.3.5 Sender behavior, up-switching

An MTSI client in terminal sending media with a bitrate lower than currently allowed bitrate should try to increase the bitrate up to the currently allowed bitrate. The bitrate of the encoded media is increased slowly until the currently allowed bitrate is reached while monitoring that the quality is maintained, i.e. no packet losses and no delay should be introduced because of the up-switch.

An MTSI client in terminal sending media with a bitrate according to the currently allowed bitrate and receiving a TMMBR request for increasing the bitrate:

- should ramp up the bitrate to the currently allowed bitrate within 0.5 seconds,
- shall ramp up the bitrate to the currently allowed bitrate within 1 second.

If during the up-switch procedure the MTSI client receives a TMMBR message for reducing the bitrate then the up-switch shall be aborted and the down-switch is started as described in clause 10.3.4.

10.3.6 Receiver behavior, down-switching

An MTSI client in terminal receiving media and detecting that the throughput is reduced shall behave as follows:

- When detecting that the throughput is reduced by more than 10% then it should send a TMMBR message requesting a bitrate that is at least 10% lower than the currently allowed bitrate,
- When detecting that the throughput is reduced by more than 25% then it shall send a TMMBR message requesting a bitrate that is at least 25% lower than the currently allowed bitrate.

TMMBR messages for down-switch are urgent feedback messages and shall be sent as soon as possibly. AVPF early mode or immediate mode, [40] shall be used whenever possible.

10.3.7 Receiver behavior, up-switch

An MTSI client in terminal receiving media and detecting that the bitrate can be increase shall behave as follows:

- If the bitrate can be increased by at least 5% then the MTSI client in terminal should send a TMMBR message requesting a bitrate that is:

$$requested_bitrate = \min((currently_allowed_bitrate + rate_increase_step), negotiated_bitrate) \quad (10.3.7-1)$$

- If the bitrate can be increased by at least 15% then the MTSI client in terminal shall send a TMMBR message requesting a bitrate that is:

$$requested_bitrate = \begin{cases} \geq \min((currently_allowed_bitrate + 0.8 * rate_increase_step), negotiated_bitrate) \\ \leq \min((currently_allowed_bitrate + rate_increase_step), negotiated_bitrate) \end{cases} \quad (10.3.7-2)$$

TMMBR messages for up-switch shall be sent with the normal compound RTCP packets following the normal RTCP transmission rules defined for the RTP/AVP profile, [9]. This is to not unnecessarily prevent possible subsequent urgent feedback messages, e.g. for down-switch, to be sent using AVPF early mode or immediate mode.

10.3.8 ECN triggered adaptation

ECN triggered adaptation may be used in addition to other adaptation triggers. However, when ECN is used an MTSI client in terminal receiving media shall also use at least one other adaptation trigger, see clause 10.3.3. When ECN is used, an MTSI client in terminal sending media shall also monitor the received RTCP SR/RR. If ANBR described in clause 10.7 is used, the bitrate value used in that bitrate recommendation shall be seen as independent from ECN and thus not taking ECN-CE markings into account. It is therefore possible that ECN-CE and ANBR with a decreased bitrate value both report on the same detected restriction.

NOTE 1: When ECN is negotiated, some networks in the path may allow ECN signalling to pass through even though the network does not actively use ECN to indicate congestion. For example, in a session between an LTE UE and a HSPA UE, the LTE access may allow and use ECN, but the backbone and the HSPA access may allow ECN to be used without marking packets with ECN-CE if congestion occurs in the backbone or in the HSPA access side.

Table 10.2 defines a mandatory set of parameters that are used by the ECN triggered adaptation. The default values for the parameters are also specified. Alternate values for these parameters may be configured into the MTSI client based on operator policy, for example using OMA-DM.

Table 10.2: Configuration parameters when ECN is used as a trigger

| Parameter | Description |
|-----------------------|--|
| ECN_min_rate_relative | Lower boundary (proportion of the bit rate negotiated for the video stream) for the media bit-rate adaptation in response to ECN-CE marking. The media bit-rate shall not be reduced below this value as a reaction to the received ECN-CE. The ECN_min_rate should be selected to maintain an acceptable service quality while reducing the resource utilization. Default value: Same as INITIAL_CODEC_RATE for video if defined, otherwise 50% |
| ECN_min_rate_absolute | Lower boundary (kbps) for the media bit-rate adaptation in response to ECN-CE marking. The media bit-rate shall not be reduced below this value as a reaction to the received ECN-CE. The ECN_min_rate should be selected to maintain an acceptable service quality while reducing the resource utilization. Default value: 48 kbps |
| ECN_congestion_wait | The waiting time after an ECN-CE marking for which an up-switch shall not be attempted. A negative value indicates an infinite waiting time, i.e. to prevent up-switch for the whole remaining session. Default value: 5 seconds |

The ECN_min_rate parameter is set to the larger of the ECN_min_rate_relative and ECN_min_rate_absolute values. Since the ECN_min_rate_relative parameter is relative to the outcome of the offer-answer negotiation this means that the ECN_min_rate value may be different for different sessions. The ECN_min_rate_absolute parameter is used to prevent too low bit rates for video, which would result in too low quality.

The configuration of adaptation parameters, and the actions taken during the adaptation, are specific to the particular triggers. For example, the adaptation may be configured to reduce the media bit-rate to ECN_min_rate when ECN-CE is detected, while it may reduce the media bit-rate even further for bad radio conditions when high PLR is detected.

Multiple ECN-CE markings within one RTP-level round-trip time is considered as the same congestion event. Each time an MTSI client detects a congestion event it shall send an adaptation request to reduce the media bit-rate unless already operating at the ECN_min_rate or below. An MTSI client detecting a congestion event shall not send an adaptation request to increase the media bit-rate for a time period ECN_congestion_wait after the end of the congestion event.

Multiple adaptation algorithms can be used in parallel, for example ECN-triggered adaptation, ANBR, and Packet Loss Rate-triggered adaptation. When multiple adaptation trigger algorithms are used for the rate adaptation, the rate that the MTSI client is allowed to use should be no higher than any of the rates determined by each adaptation algorithm.

NOTE 2: For example, if the ECN-triggered adaptation indicates that 100kbps should be used and if the PLR-triggered adaptation indicates that 75kbps should be used then the rate that the MTSI client uses should be no higher than $\min(100, 75) = 75$ kbps.

10.4 Text

Rate adaptation (downgrade of used bandwidth) of text shall follow the recommendation in clause 9 of RFC 4103 [31]. RTCP reports are used as indicator of loss rate over the channel.

When the transmission interval has been increased in order to handle a congestion situation, return to normal interval shall be done when RTCP reports low loss.

10.5 Explicit Congestion Notification

When the (e)NodeB experiences congestion it may set the ECN bits in the IP header to '11' to indicate "Congestion Experienced" for packets that have been marked with ECN Capable Transport (ECT), [83], [84].

Adaptation requests should be sent in response to ECN congestion events. Clauses 10.2 and 10.3 describe adaptation for speech and video when ECN-CE is detected.

10.6 Using the a=bw-info attribute for adaptation

This sub-clause outlines a few generic recommendations for how the bandwidth properties signalled with the 'a=bw-info' may be used for the adaptation. Media specific usage may override these guidelines.

During the session, when adaptation is needed:

- The bit rate range from the Minimum Desired Bandwidth up to the Maximum Desired Bandwidth (if different) defines the most commonly used adaptation range. The primary means for the adaptation in this range is adapting the source encoding while adapting the RTP packetization should remain unchanged. This includes adapting the frame range for video.
 - When adapting in this range, downwards rate adaptation should be fast while upwards rate adaptation should be relatively slow. This is because when these bandwidth properties are different then it is likely that an MBR>GBR bearer is used and the performance defined with the QoS parameters is only guaranteed when the bit rate does not exceed the GBR [90].
- The bit rate range above the Maximum Desired Bandwidth up to the Maximum Supported Bandwidth (if different) is mainly intended for sending application layer redundancy, in case additional bandwidth is needed for this purpose.
 - It is preferable to first try to overcome the degraded operating conditions by reducing the bit rate, at least down to the Minimum Desired Bandwidth or even down to the Minimum Supported Bandwidth, before adding application layer redundancy.
- The bit rate range below the Minimum Desired Bandwidth down to the Minimum Supported Bandwidth (if different) is mainly intended to be used to keep the session alive during severely degraded operating conditions. For video, this can be achieved by e.g. reducing the resolution or the frame rate. For speech, this can be achieved by using frame aggregation or using a codec mode below the Minimum Desired Bandwidth. The end-to-end delay may be significantly increased and/or the quality may be significantly reduced. It may still be preferable to keep the media alive compared to e.g. video freezing, even if the quality is degraded.
- The Minimum Supported Bandwidth may be set larger than zero to limit the adaptation, e.g. to fulfil certain service requirements on end-to-end delay or minimum quality level. If the throughput is so low that not even the Minimum Supported Bandwidth can be fulfilled then there is likely no reason to continue using that media type in the session.

10.7 Access network bitrate recommendation

10.7.1 General

Support and use of access network bitrate recommendations (ANBR) as described in this clause are optional for MTSI clients in terminal. Clause 10.7 does not apply to an MTSI client in terminal that does not support the ANBR message.

Some access networks may provide the MTSI client in terminal with ANBR messages, separately per local access bearer and separately for the local uplink and downlink. It is expected that an ANBR message is sent to the MTSI client in terminal whenever the access network finds it reasonable to inform about a change in the recommended bitrate, such that the MTSI client in terminal is generally provided with up-to-date recommended bitrate information.

In general, a single access bearer can carry multiple RTP streams, in which case ANBR applies to the sum of the individual RTP stream bitrates on that bearer.

Access networks supporting ANBR may also support a corresponding ANBR Query (ANBRQ) message, which allows the MTSI client in terminal to query the network for updated ANBR information. ANBRQ shall only be used to query for an ANBR update when media bitrate is to be increased, not for media bitrate decrease.

The ANBR and ANBRQ messages, as used in this clause, are *conceptual* messages that allows generalization of the description between different accesses, e.g. LTE (see 10.7.4) and NR (see 10.7.5) and wireless LAN. There shall be a defined mapping between the conceptual ANBR/ANBRQ and actual messages for each access where ANBR/ANBRQ signaling is to be used. The format of such access-specific ANBR/ANBRQ messages may differ between different types of access networks, and there may not even exist a one-to-one mapping of messages. The recommended bitrate value in ANBR/ANBRQ is here defined to include IP and higher layer overhead, including bitrate used for RTCP signaling (as opposed to e.g. b=AS line in SDP, which does not include RTCP). Other definitions can be used by the individual access network mappings (for LTE and NR, the recommended physical layer bitrate is signalled by the access network, see clauses 10.7.4 and 10.7.5), e.g., including overhead below IP layer that is added by the access network, and the UE shall then perform appropriate value translation, e.g. adjusting for use of ROHC and removing the lower layer overhead.

While the sizes of all other protocol overheads are static or change slightly during an MTSI session, the size of ROHC header is highly dynamic, and hence there is no deterministic and standardized way to map the recommended physical layer bitrate into the IP layer bitrate. The queried physical layer bitrate should be set considering all the L2 and above headers.

NOTE 1: The UE may determine the corresponding IP layer bitrate based on the long-term average of the IP packet sizes, L2 header sizes, and ROHC header sizes, but the translation methodologies and the estimation error levels required to implement accurate media bitrate adaptation have not been specified.

NOTE 2: The eNodeB may determine the corresponding IP layer bitrate based on the long-term average of the IP packet sizes, L2 header sizes, and ROHC header sizes, but the translation methodologies and the estimation error levels required to implement accurate media bitrate adaptation have not been specified.

NOTE 3: The recommended/queried bitrate as signalled over the LTE and NR access is defined to be in kbps at the physical layer. The uplink/downlink bitrate at the physical layer is $r_{UL/DL} = \frac{\sum_k L_k}{T}$, where L_k is the bit-length of the k -th successfully transmitted/received TB by the UE within the window T . In TS 36.321 and 38.321, a window length of 2000 ms is applied.

10.7.2 Relation to session signaling bitrate information

ANBR is only a recommendation and does not change any bitrate restrictions established by session signaling, described in clause 6.2.5.1. Such unaffected session signaling bitrate information includes:

- SDP "b=" lines, including "b=AS", "b=RS", and "b=RR".
- SDP "a=bw-info" lines, if present.
- Codec-specific min/max bitrates, based on codec configuration and/or packetization parameters.
- MBR and GBR QoS parameters.

An ANBR value with a bitrate above a maximum bitrate limit established by any of the above shall therefore instead be considered as a bitrate recommendation for the lowest of the above listed maximum bitrates (except for GBR that is not considered a maximum bitrate). When using a codec configuration with discrete bitrate steps, and if the ANBR value does not exactly match such discrete codec bitrate, it shall be considered as a bitrate recommendation for the next lower, signaled codec bitrate.

If a GBR bearer is used ($GBR > 0$), an ANBR value below GBR may be ignored, but the MTSI client in terminal should then be prepared to handle a higher than target packet loss and / or delay for the affected bearer. An ANBR value of 0 should be taken as a recommendation to temporarily stop sending RTP media on the affected bearer, without re-negotiating the session, with the exception for RTP media related to the first "m=audio" line in the SDP that shall never temporarily stop sending based on ANBR value 0. Corresponding RTCP transmission shall, when enabled, continue even when RTP media is stopped due to ANBR value 0.

10.7.3 Use with dynamic bitrate adaptation

10.7.3.1 General

An MTSI client in terminal shall use the ANBR message as adaptation trigger, taking other available triggers into account. The same principle shall apply for both speech and video, adapting to the lowest bitrate resulting from any of

the possibly multiple, available triggers. Clauses 10.2 and 10.3 describe adaptation details for speech and video. Use of ANBR in combination with ECN signaling is described in clause 10.3.8.

A received ANBR message for a certain access bearer and media direction shall be considered valid for use as input to adaptation trigger evaluation until either another ANBR message is received for the same access bearer and media direction, until that access bearer is closed, or until the SIP session is either re-negotiated or closed.

10.7.3.2 Adaptation of sent media

This relates to adaptation of the media in RTP streams that the MTSI client in terminal sends in the uplink direction, controlling the local media encoder bitrate.

The below figures with signaling diagrams describe ANBR usage for a single media, regardless if that is voice or video. The "Request max" message in those figures is a generalized application level message that corresponds to RTCP TMMBR for video and CMR or RTCP-APP for voice. Similarly, the "Notify max" message is a generalized application level message that corresponds to RTCP TMMBN for video. "Notify max" has no counterpart for voice and is therefore not used for voice.

An MTSI client in terminal that receives both ANBR with bitrate R_0 and a "Request max" message with bitrate R_1 for its media sending direction shall use $\min(R_0, R_1)$ as the combined bitrate for those two adaptation triggers.

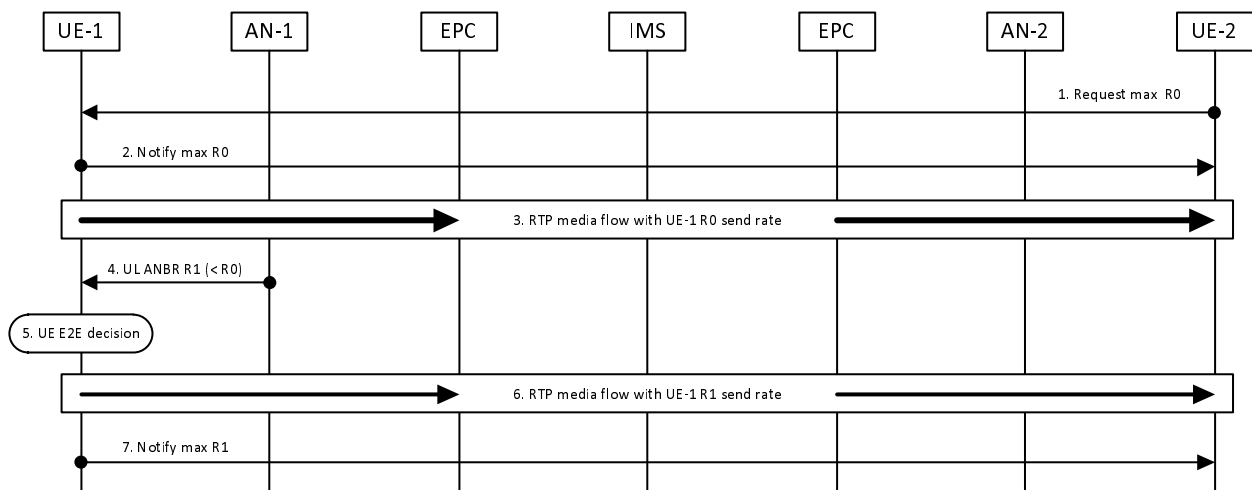


Figure 10.7-1 Uplink bitrate decrease based on ANBR

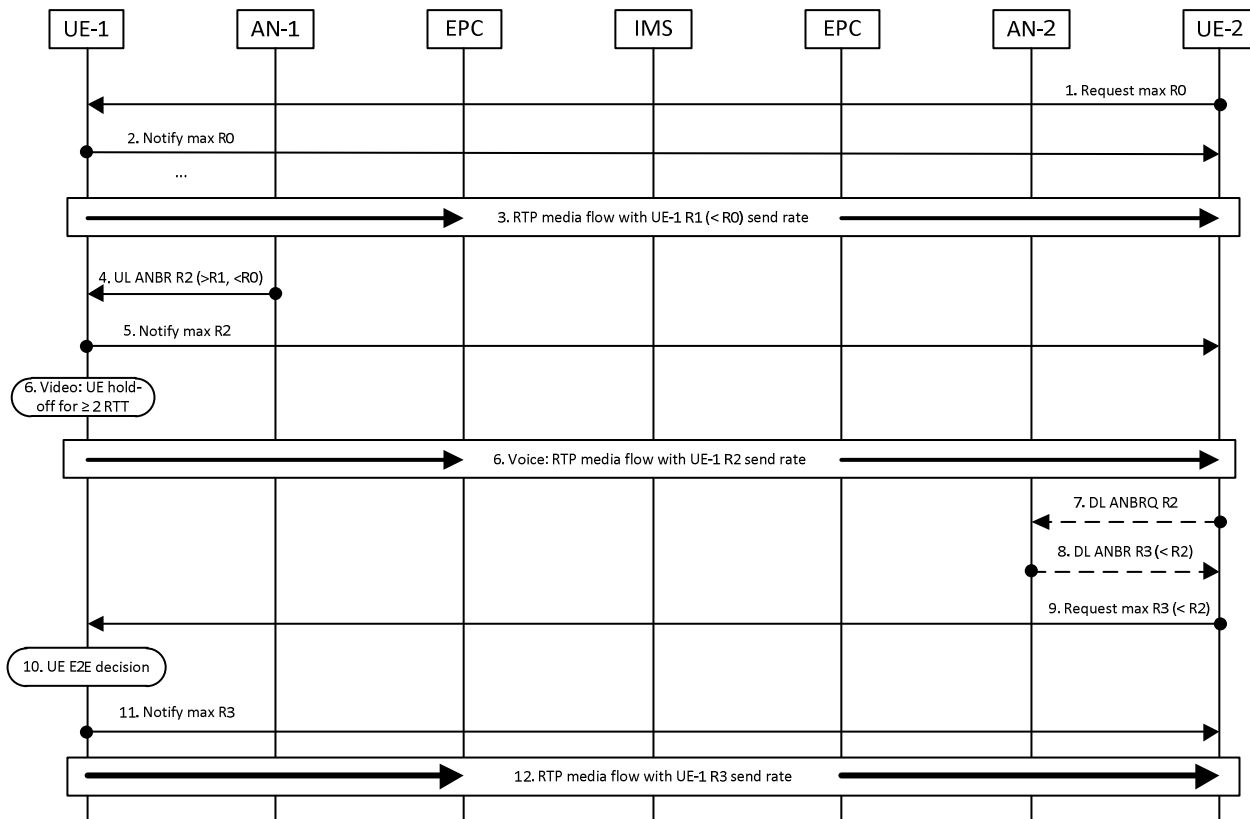


Figure 10.7-2 Uplink bitrate increase based on ANBR

When an MTSI client in terminal receives an ANBR message for the local uplink that triggers an adaptation decision (step 4 of Figures 10.7-1 and 10.7-2):

1. For both video and voice, an adaptation resulting in a reduction of the media sender bitrate shall be initiated immediately without further signaling (step 6 of Figure 10.7-1).
2. For the case of video and if TMMBR / TMMBN are supported in the session:
 - a) If the adaptation decision means that the MTSI client in terminal adapts bitrate below the most recently received TMMBR message (if any, step 4 of Figures 10.7-1 or 10.7-2), the media sender itself owns the uplink bitrate restriction and a corresponding TMMBN message shall be sent, notifying the remote media receiver of this changed local uplink restriction (step 7 of Figure 10.7-1 or step 5 of Figure 10.7-2). Such TMMBN message has the media sender's own SSRC included in the bounding set [43]. Note that only the media sender or receiving TMMBR with a lower bitrate can then remove such own restriction, which means that TMMBR messages with a higher bitrate received from the remote media receiver will be ineffective and ignored. The media sender can repeat this procedure and either increase or decrease the used bitrate based on subsequent local uplink ANBR, sending corresponding TMMBN also for those changes, as long as the MTSI client in terminal does not receive a TMMBR from the remote MTSI client with a bitrate lower than the most recently sent TMMBN.
 - b) An adaptation resulting in an increase of the media sender bitrate in uplink (Figure 10.7-2) shall delay the media bitrate increase (step 6 of Figure 10.7-2) to allow sufficient time for the remote media receiver to receive and react to the TMMBN in bullet 2.a above, as described by section 3.5.4 of CCM [43] (steps 7-9 of Figure 10.7-2). After such delay, the media sender bitrate is increased (step 12 of Figure 10.7-2). The bitrate increase shall take all available adaptation triggers into account, which can cause the bitrate increase to be separated into several steps (see clause C.2.5).
 - c) It is recommended that the bitrate in a TMMBN from bullet 2.b) that is pre-announcing an increase in the media sender bitrate in uplink is set to correspond to the received ANBR message and not to the next bitrate step in a step-wise increase (see clause 10.3.5), to avoid unnecessary TMMBN, ANBRQ, and ANBR signaling (see also bullet 4) below).

- For the case of voice, an adaptation resulting in an increase of the media sender bitrate in uplink shall result in an increase of media sender bitrate at the earliest opportunity (step 6 of Figure 10.7-2), taking other adaptation triggers into account, such as CMR.

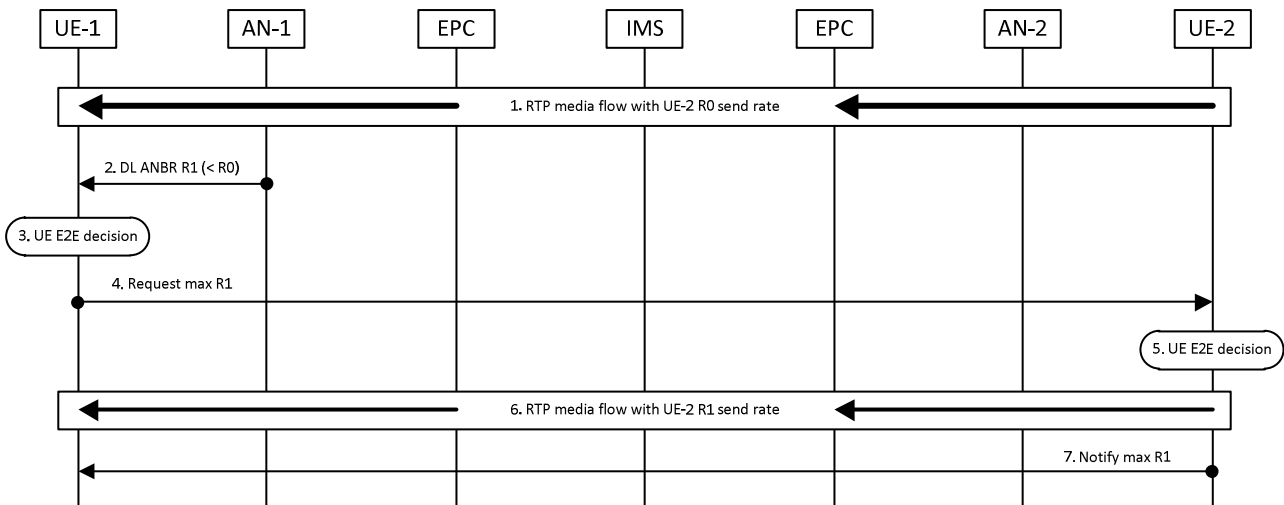


Figure 10.7-3 Downlink bitrate decrease based on ANBR through application signaling

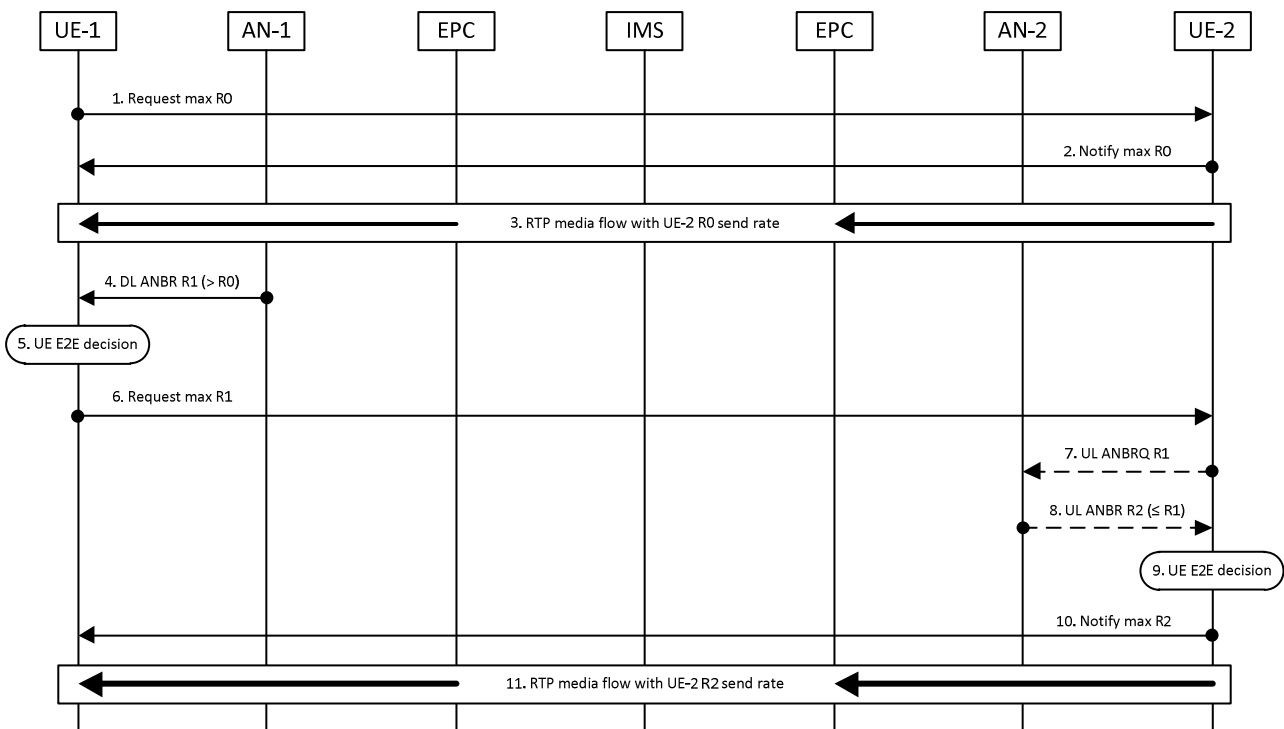


Figure 10.7-4 Downlink bitrate increase based on ANBR through application signaling

When an MTSI client in terminal receives application signaling for bitrate adaptation of media, such as CMR (for speech) or TMMBR (for video), that triggers an adaptation decision (step 4 of Figure 10.7-3 or step 6 of Figure 10.7-4):

- For video and if TMMBR is supported in the session, when receiving a TMMBR that would result in an increase of the media sender bitrate in uplink direction (step 6 of Figure 10.7-4), the media sender shall take all available adaptation triggers for the local uplink into account, e.g. any bitrate limit from the most recently received ANBR message. If the media sender has reason to believe that the most recently received ANBR for its uplink no longer applies, it may send an ANBRQ message for its uplink (step 7 of Figure 10.7-4), if supported, to trigger

receiving an ANBR message with recent information (step 8 of Figure 10.7-4) before deciding on what bitrate value to send in a TMMBN (step 10 of Figure 10.7-4) and to use for media in the uplink direction (step 11 of Figure 10.7-4).

5. For voice, when receiving a CMR that would result in an increase of the media sender bitrate in uplink direction (step 6 of Figure 10.7-4), the media sender shall take all available adaptation triggers for the local uplink into account, e.g. any bitrate limit from the most recently received ANBR message. If the media sender has reason to believe that the most recently received ANBR for its uplink no longer applies, it may send an ANBRQ message for its uplink (step 7 of Figure 10.7-4), if supported, to trigger receiving an ANBR message with recent information (step 8 of Figure 10.7-4) before deciding on what voice mode to use in the uplink direction (step 11 of Figure 10.7-4).

10.7.3.3 Adaptation of received media

This relates to adaptation of the media in RTP streams that the MTSI client in terminal receives in the downlink direction, which can require sending application-level messages to adapt the remote media encoder bitrate.

When an MTSI client in terminal receives an ANBR message for the local downlink that triggers an adaptation decision (step 2 of Figure 10.7-3 or step 4 of Figure 10.7-4):

1. For the case of video and if TMMBR / TMMBN are supported in the session:

- a) A corresponding TMMBR message requesting the remote media sender to change its rate to match the local downlink restriction shall be sent, as described in clause 10.3.2 (step 4 of Figure 10.7-3 or step 6 of Figure 10.7-4).

NOTE: Adaptation is not triggered if the most recently received TMMBN from the remote media sender indicated a lower bitrate than would be included in a TMMBR message, because the remote media sender is then owning the bitrate limit itself (similar to bullet 2.a of the local uplink in clause 10.7.3.2 above).

- b) It is recommended that the bitrate in a TMMBR from bullet 1.a) that is increasing the media sender bitrate is set to correspond to the most recently received ANBR message, to avoid unnecessary TMMBR, ANBRQ, and ANBR signaling caused by a possible step-wise increase (see also bullet 2.c in 10.7.3.2 above).

2. For the case of voice, adaptation signaling to match the local downlink restriction shall be initiated towards the remote media sender, as described in clause 10.2 (step 4 of Figure 10.7-3 or step 6 of Figure 10.7-4).

When an MTSI client in terminal receives application signaling for bitrate adaptation related to received media, such as TMMBN (for video):

3. A media receiver receiving a TMMBN with increased bitrate and where the remote media sender owns the restriction (see bullet 2.a of 10.7.3.2 and step 5 of Figure 10.7-2) shall re-evaluate its downlink adaptation triggers and, if an adaptation decision arrives at a lower bitrate value than in the received TMMBN (step 5 of Figure 10.7-2), send a TMMBR with that lower bitrate, as described by section 3.5.4 of CCM [43]. When deciding whether or not to send TMMBR, the media receiver shall take all available adaptation triggers into account, e.g. bitrate limit from the most recently received downlink ANBR message. If the media receiver has reason to believe the most recently received ANBR for its downlink no longer applies, it may send an ANBRQ message for its downlink (step 7 of Figure 10.7-2), if supported, to trigger receiving an ANBR message with recent information (step 8 of Figure 10.7-2).

10.7.4 Message mapping for LTE access

When using LTE access, ANBR is mapped to a MAC level message named "Recommended bit rate MAC Control Element" sent by the eNodeB and applicable to a specific dedicated bearer, as described by [85] and [157]. Similarly, when using LTE access, ANBRQ is mapped to a MAC level message named "Recommended bit rate query MAC Control Element" sent to the eNodeB and applicable to a specific, existing dedicated bearer, as described by [85] and [157]. An MTSI client in terminal using LTE access may support ANBR and ANBRQ signaling.

10.7.5 Message mapping for NR access

When using NR access, ANBR is mapped to a MAC level message named "Recommended bit rate MAC Control Element" sent by the gNB and applicable to a specific logical channel which is mapped to the single media flow (e.g.,

audio or video) to which the recommended bit rate applies. Similarly, when using NR access, ANBRQ is mapped to a MAC level message named "Recommended bit rate query MAC Control Element" sent to the gNB and applicable to a specific, existing logical channel which is mapped to the single media flow to which the recommended bit rate applies. An MTSI client in terminal using NR access may support ANBR and ANBRQ signaling.

11 Front-end handling

11.1 General

Terminals used for MTSI shall conform to the minimum performance requirements on the acoustic characteristics of 3G terminals specified in TS 26.131 [35]. The codec modes and source control rate operation (DTX) settings shall be as specified in TS 26.132 [36].

Furthermore, the test point (Point-of-Interconnect (POI)) specified in [35] shall be a reference terminal capable of receiving digital speech data at the send side and producing a digital output of the received signal (see figure 11.1). During the testing, the radio conditions should be error free and the jitter and packet loss in the IP transport shall be kept to a minimum.

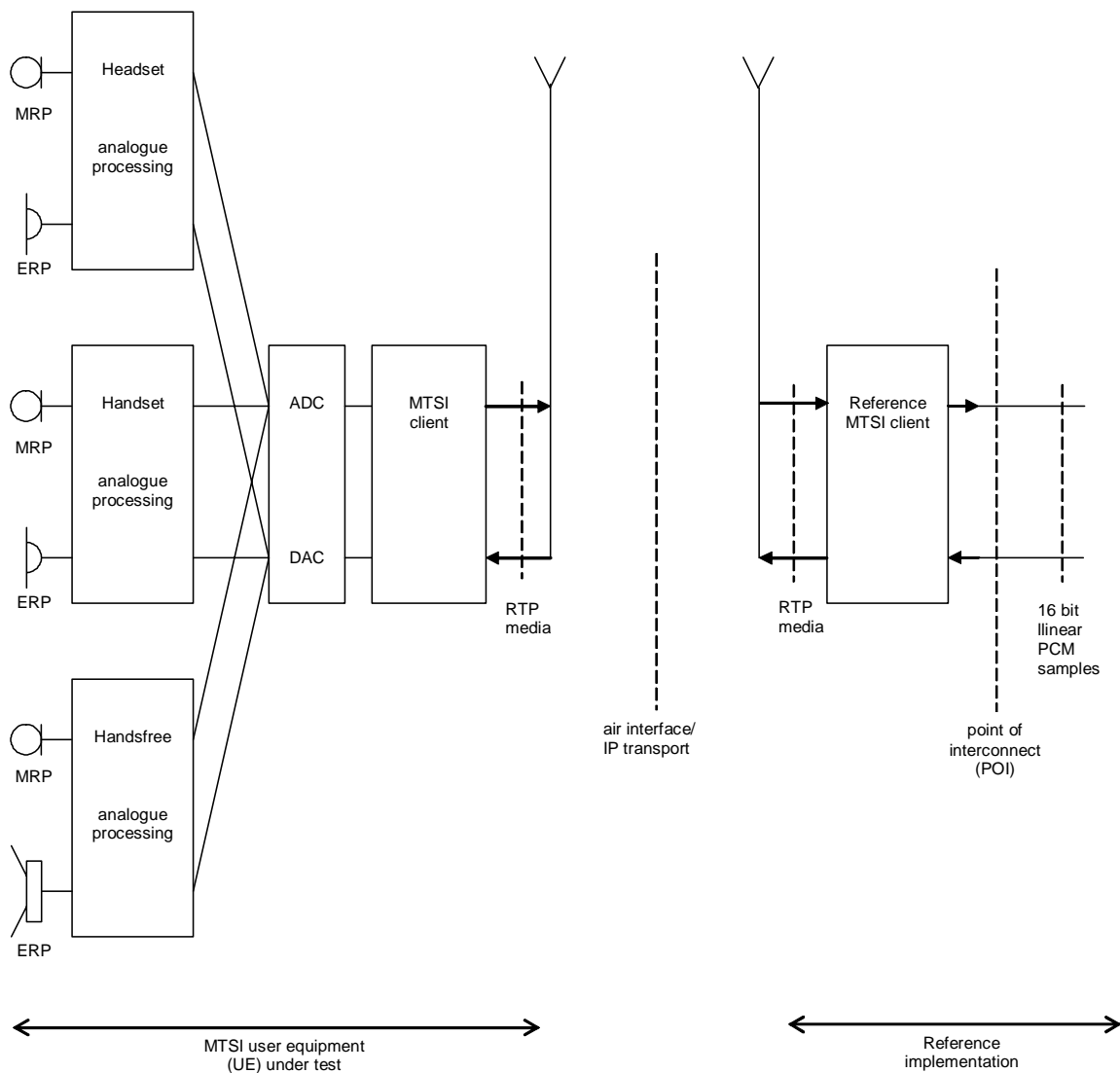


Figure 11.1: Interface for testing acoustic properties of a terminal used for MTSI

12 Inter-working

12.1 General

In order to support inter-working between different networks it is good if common codecs for the connection can be found. Requirements for different networks are described in this clause. In some cases functionality is also needed in the network to make the inter-working possible (e.g. MGCF and MGW).

NOTE: The term MTSI MGW (or MTSI Media gateway) is used in a broad sense, as it is outside the scope of the current specification to make the distinction whether certain functionality should be implemented in the MGW or in the MGCF.

12.2 3G-324M

12.2.1 General

Inter-working functions are required between IMS and CS. There are separate functions, in e.g. a MGCF, for control-plane inter-working (see TS 29.163 [65]) and, in e.g. a IM-MGW, for user-plane inter-working. Control-plane inter-working includes for instance SIP ↔ BICC and SIP ↔ H.245 protocol translations, whereas user-plane inter-working requires transport protocol translations and possibly transcoding.

12.2.2 Codec usage

12.2.2.1 General

An interoperable set of speech, video and real-time text codecs is specified for 3G-324M and MTSI. Both video codec level and maximum bitrate can be specified as part of the call setup negotiation (see clause 12.2.5). Thus, it may be possible that the MTSI client in terminal and a CS UE agree on a common codec end-to-end without the need for MGW transcoding.

If a common codec is not found and the MTSI MGW does not support transcoding between any of the supported codecs, then the controlling MGCF may drop the unsupported media component. If the speech part cannot be supported, then the connection should not be set up.

12.2.2.2 Text

A channel for real-time text is specified in ITU-T H.324. Presentation and coding is specified according to ITU-T Recommendation T.140, which is also used for MTSI clients (see clause 7.4.4). Inter-working is a matter of establishing the text transport channels and moving the text contents between the two transport levels.

12.2.3 Payload format

See clause 7.4 of the present document.

12.2.4 MTSI media gateway trans-packetization

12.2.4.1 General

The MTSI MGW shall offer conversion between H.223 as used in 3G-324M on the CS side and RTP as used in IMS. This clause contains a list inter-working functionalities that should be included.

12.2.4.2 Speech de-jitter buffer

The MTSI MGW should use a speech de-jitter buffer in the direction IMS to CS with sufficient performance to meet the 10 milliseconds maximum jitter requirement in clause 6.7.2 of ITU-T Recommendation H.324. H.324 specifies that

transmission of each speech AL-SDU at the H.223 multiplex shall commence no later than 10 milliseconds after a whole multiple of the speech frame interval, measured from transmission of the first speech frame.

12.2.4.3 Video bitrate equalization

Temporary video rate variations can occur on the IMS side for example due to congestion. The video rate on the CS side, in contrast, is under full control of the CS side UE and the MGCF.

During session setup, the MGCF shall negotiate a video bitrate on the IMS side that allows all video bits to be conveyed to/from the CS link.

A buffer shall be maintained at the IM-MGW in the direction from the IMS to the CS side. The size of the buffer should be kept small enough to allow for a low end-to-end delay, yet large enough to conceal most network jitter on the IMS side. Temporary uneven traffic on the IMS side, beyond the handling capability of the buffer, should be handled as follows: if the buffer overflows, RTP packets should be dropped and the resulting loss and observed jitter should be reported by the means of an RTCP RR at the earliest possible sending time. The drop strategy may preferably be implemented media aware (i.e. favouring dropping predicted information over non-predicted information and similar techniques), or may be drop-head. If the buffer runs empty, the CS side should insert appropriate flag stuffing.

A buffer shall be maintained in the direction from the CS to the IMS side. The size of the buffer should be kept small enough to allow for a low end-to-end delay, but large enough to conceal most network jitter on the CS side. If the buffer overflows, then video bits must be dropped, preferably in a media-aware fashion, i.e. at GOB/slice/picture boundaries. IM-MGWs may also take into account the type of media data, i.e. coded with or without prediction. When the buffer runs empty, no activity is required on the IMS side.

If the CS video call is changed to a speech-only call [46], the video component on the IMS side shall be dropped.

12.2.4.4 Data loss detection

If RTP packet loss is detected on input to the MTSI MGW at the IMS side, including losses caused by buffer-full condition as described above, corresponding H.223 AL-SDU sequence number increments should be made on the CS side to enable loss detection and proper concealment in the receiving CS UE.

If packet loss is detected on the CS side, e.g. through H.223 AL-SDU sequence numbers, those losses should be indicated towards the IMS side through corresponding RTP packet sequence number increments. The deliberate increments made for this reason will be visible in the RTCP RR from the MTSI client and the MTSI MGW should take that into account when acting on RTCP RR from the MTSI client, as the CS side losses are not related to the IMS network conditions.

12.2.4.5 Data integrity indication

This is mainly relevant in the direction from CS to IMS. The H.223 AL-SDUs include a CRC that forms an unreliable indication of data corruption. On the IMS side, no generic protocol mechanisms are available to convey this CRC and/or the result of a CRC check. The MTSI MGW shall discard any AL-SDUs which fail a CRC check and are not of a payload type that supports the indication of possible bit errors in the RTP payload header or data. If such payload type is in use, the MTSI MGW may forward corrupted packets, but in this case shall indicate the possible corruption by the means available in the payload header or data. One example is setting the Q bit of RFC 3267 [28] to 0 for AMR speech data that was carried in an H.223 AL-SDU with CRC indicating errors. Another example is setting the F bit of RFC 6184 [25] for H.264 (AVC) NAL units or the F bit of [120] for H.265 (HEVC) NAL units that may contain bit errors.

The H.223 AL-SDU CRC is not fully fail-safe and it is therefore recommended that a MTSI client is designed to be robust and make concealment of corrupt media data, similar to the CS UE.

12.2.4.6 Packet size considerations

12.2.4.6.0 General

The same packet size and alignment requirements and considerations as defined in clause 7.5.2 of the present document and in TS 26.111 [45] apply to the MTSI MGW and controlling MGCF, as it in that sense acts both as a MTSI client towards the IMS and as a CS UE towards the CS side. Maximum available buffer size for packetization of media data

may differ between IMS and CS UE. To avoid non-favourable segmentation of data (especially video) by the MTSI MGW, the controlling MGCF should indicate the SDP 'a' attribute "3gpp_MaxRecvSDUSize" to the MTSI client in terminal. This attribute indicates the maximum SDU size of the application data (excluding RTP/UDP/IP headers) that can be transmitted to the receiver without segmentation. The specific maximum SDU size limit is determined by the MGCF from the H.245 bearer capability exchange between the CS UE and the MGCF. For example, the MTSI MGW determines this through the maximumA12SDUSize and maximumA13SDUSize fields of the H223Capability member in H.245 TerminalCapabilitySet message.

12.2.4.6.1 The Maximum Receive SDU Size attribute "3gpp_MaxRecvSDUSize"

The ABNF for the maximum receive SDU size attribute is described as follows:

```
Max-recv-SDU-size-def = "a" "=" "3gpp_MaxRecvSDUSize" ":" size-value CRLF
size-value              = 1*5DIGIT; 0 to 65535 in octets
```

The value "size-value" indicates the maximum SDU size of application data, excluding RTP/UDP/IP headers, that can be transmitted to the other end point without segmentation.

The parameter "3gpp_MaxRecvSDUSize" should be included in the SDP at the session level and/or at the media level. Its usage is governed by the following rules:

1. At the session level, the "3gpp_MaxRecvSDUSize" attribute shall apply to the combination of the data from all the media streams in the session.
2. At the media level, the "3gpp_MaxRecvSDUSize" attribute indicates to the MTSI client in terminal that this particular media stream in the session has a specific maximum SDU size limit beyond which received SDUs will be segmented before delivery to the CS UE.
3. If the "3gpp_MaxRecvSDUSize" attribute is included at the session and media levels, then the particular media streams have specific maximum SDU size limits for their own data while the session has an overall maximum SDU size limit for all the media data in the session.

The MGCF includes the "3gpp_MaxRecvSDUSize" attribute in the SDP offer or answer sent to the MTSI client in terminal after the MGCF determines the bearer capability of the CS UE (see Annex E of [65]). Upon reception of the SDP offer or answer that includes the "3gpp_MaxRecvSDUSize" attribute, the MTSI client in terminal need not include this attribute in its subsequent exchange of messages with the MTSI MGW.

There are no offer/answer implications on the "3gpp_MaxRecvSDUSize" attribute. The "3gpp_MaxRecvSDUSize" attribute in the SDP from the MTSI MGW is only an indication to the MTSI client in terminal of the maximum SDU size that avoids segmentation for the specified media streams and/or session.

NOTE: Default operation in the absence of the "3gpp_MaxRecvSDUSize" attribute in SDP is to not have any SDU size limits for any of the media streams or session.

12.2.4.7 Setting RTP timestamps

In general, no explicit timestamps exist at the CS side. Even without transcoding functionality, the MTSI MGW may have to inspect and be able to interpret media data to set correct RTP timestamps.

12.2.4.8 Protocol termination

The MTSI MGW shall terminate the H.223 protocol at the CS side. Similarly, the MTSI MGW shall terminate RTP and RTCP at the IMS side.

12.2.4.9 Media synchronization

The IM-MGW and controlling MGCF should forward and translate the timing information between the IMS side (RTP timestamps, RTCP sender reports) and the CS side (H.245 message H223SkewIndication) to allow for media synchronization in the MTSI client in terminal and the CS UE. The MTSI MGW shall account for its own contribution to the skew in both directions. Note that transmission timing of H223SkewIndication and RTCP SR must be decoupled. H223SkewIndication has no timing restrictions, but is typically sent only once in the beginning of the session. RTCP SR timing is strictly regulated in RFC 3550 [9], RFC 4585[40], and clause 7.3. To decouple send timings, the time shift

information conveyed in H223SkewIndication and RTCP SR must be kept as part of the MTSI MGW/MGCF session state. H223SkewIndication should be sent at least once, and may be sent again when RTCP SR indicates a synchronization change. A synchronization change of less than 50 ms (value to be confirmed) should be considered insignificant and need not be signalled.

NOTE: This procedure is not supported in the present Release in a decomposed MGCF and IM-MGW, as H.245 is treated on the MGCF and RTCP is sent at the IM-MGW, and no means are defined to forward information from the H223SkewIndication over the Mn interface.

12.2.5 Session control

The MGCF shall offer translation between H.245 and SIP/SDP signalling according to TS 29.163 [65] to allow for end-to-end capability negotiation.

12.3 GERAN/UTRAN CS inter-working

This clause defines requirements only for the PS side of the MGW, i.e. for the PS session in-between the MTSI client in a terminal and the MGW. The CS side of the MGW, i.e. in-between the MGW and the CS terminal, is out of scope of this clause.

This clause applies for MTSI MGWs supporting inter-working between a CS terminal using CS GERAN/UTRAN access or an MTSI client in terminal performing SRVCC to CS and:

- an MTSI client in terminal using 3GPP access; or
- an MTSI client in terminal using fixed access; or
- a non-MTSI client.

The requirements and recommendations for these three cases are harmonized to enable using the same procedures regardless of the type of PS client and what access it uses, as long as it uses IP based access.

The target for this clause is to enable tandem-free operation when the same codec (AMR or AMR-WB) is used by both end-points.

An MTSI MGW may also support the other codecs listed in clause 18.2.2 for inter-working between an MTSI client in terminal using fixed access and a CS terminal using GERAN or UTRAN access. This means that tandem coding will be used and then the PS side and the CS side operate independently of each other. This further means that the requirements and recommendations for the PS side of the MGW are the same as for an MTSI client in terminal using fixed access, as described in clause 18, unless it is explicitly defined below.

12.3.0 3G-324M

If 3G-324M is supported in the GERAN/UTRAN CS, then the inter-working can be made as specified in clause 12.2.

12.3.1 Codecs for MTSI media gateways

12.3.1.1 Speech interworking between 3GPP PS access and CS GERAN/UTRAN

This clause applies to MTSI MGWs used for interworking between an MTSI client in terminal using 3GPP access and a CS GERAN/UTRAN UE.

MTSI media gateways supporting speech communication between an MTSI client in terminal using 3GPP access and terminals operating in the CS domain in GERAN and UTRAN should support Tandem-Free Operation (TFO) for AMR or AMR-WB according to TS 28.062 [37], and Transcoder-Free Operation (TrFO), see TS 23.153 [38].

MTSI media gateways supporting speech communication and supporting TFO and/or TrFO shall support:

- AMR speech codec modes 12.2, 7.4, 5.9 and 4.75 [11], [12], [13], [14] and source-controlled rate operation [15].

MTSI media gateways should also support the other AMR codec types and configurations as defined in Clause 5.4 in [16].

In the receiving direction, from the MTSI client in the terminal, the MTSI media gateway shall be capable of restricting codec mode changes to be aligned to every other frame border and shall be capable of restricting codec mode changes to neighbouring codec modes within the negotiated codec mode set.

NOTE 1: This means that the MTSI client in a terminal will apply and accept mode changes according to UMTS AMR2 [16]. An example of an SDP offer for how the MTSI MGW can restrict AMR mode changes in the MTSI client in a terminal is shown in Table A.2.1. An example of an SDP answer from the MTSI MGW for restricting the mode changes in the MTSI client in a terminal is shown in Table A.3.4a.

MTSI media gateways supporting wideband speech communication at 16 kHz sampling frequency and supporting TFO and/or TrFO for wideband speech shall support:

- AMR wideband codec 12.65, 8.85 and 6.60 [17], [18], [19], [20] and source controlled rate operation [21].

MTSI media gateways supporting wideband speech communication at 16 kHz sampling frequency should also support the other AMR-WB codec types and configurations as defined in [16].

In the receiving direction, from the MTSI client in the terminal, the MTSI media gateway shall be capable of restricting codec mode changes to be aligned to every other frame border and shall be capable of restricting codec mode changes to neighbouring codec modes within the negotiated codec mode set.

NOTE 2: This means that the MTSI client in a terminal will apply and accept mode changes according to UMTS AMR-WB [16]. An example of an SDP offer for how the MTSI MGW can restrict AMR and AMR-WB mode changes in the MTSI client in a terminal is shown in Table A.2.4. An example of an SDP answer from the MTSI MGW for restricting the mode changes in the MTSI client in a terminal is shown in Table A.3.4.

MTSI MGWs supporting wideband speech communication shall also support narrowband speech communications. When offering both wideband speech and narrowband speech communication, wideband shall be listed as the first payload type in the m line of the SDP offer (RFC 4566 [8]).

Requirements applicable to MTSI media gateways for DTMF events are described in Annex G.

12.3.1.1a Speech inter-working between fixed access and CS GERAN/UTRAN

This clause applies to MTSI MGWs used for interworking between an MTSI client in terminal using fixed access and a CS GERAN/UTRAN UE.

Media codecs for MTSI MGWs for speech inter-working between fixed access and CS GERAN/UTRAN are specified in TS 181 005 [98] in clause 6.2 for narrow-band codecs and in clause 6.3 for wide-band codecs.

MTSI MGWs for speech inter-working between fixed access and CS GERAN/UTRAN supporting AMR and AMR-WB shall follow clause 12.3.1.1 for the AMR and AMR-WB codecs. Tandem-free inter-working should be used whenever possible.

For the other codecs, the MTSI MGW shall follow the recommendations and requirements defined in clause 18 for the respective codec. For these codecs, tandem-free inter-working is not possible when interworking with CS GERAN/UTRAN.

Requirements applicable to MTSI media gateways for DTMF events are described in Annex G.

12.3.1.2 Text

The CTM coding format defined in TS 26.226 [52] is used for real time text in CS calls. In order to arrange inter-working, a transcoding function between CTM and RFC 4103 is required in the MTSI media gateway. A buffer shall be used for rate adaptation between receiving text from a real-time text transmitter according to the present document and transmitting to a CTM receiver. A gateway buffer of 2K characters is considered sufficient according to clause 13.2.4 in EG 202 320 [51].

12.3.2 RTP payload formats for MTSI media gateways

12.3.2.1 Speech

For RTP payload formats, see clause 18.4.3.

MTSI media gateways supporting AMR or AMR-WB shall support the bandwidth-efficient payload format and should support the octet-aligned payload format. When offering both payload formats, the bandwidth-efficient payload format shall be listed before the octet-aligned payload format in the preference order defined in the SDP.

The MTSI media gateway should use the SDP parameters defined in table 12.1 for the session.

For all access technologies and for normal operating conditions, the MTSI media gateway should encapsulate the number of non-redundant speech frames in the RTP packets that corresponds to the ptime value received in SDP from the other MTSI client, or if no ptime value has been received then according to "Recommended encapsulation" defined in table 12.1. The MTSI media gateway may encapsulate more non-redundant speech frames in the RTP packet but shall not encapsulate more than 4 non-redundant speech frames in the RTP packets. The MTSI media gateway may encapsulate any number of redundant speech frames in an RTP packet but the length of an RTP packet, measured in ms, shall never exceed the maxptime value.

Table 12.1: Recommended encapsulation parameters

| Access technology | Recommended encapsulation (if no ptime and no RTCP_APP_REQ_AGG has been received) | ptime | maxptime when redundancy is not supported | maxptime when redundancy is supported |
|-----------------------|---|------------------|---|---------------------------------------|
| Default | 1 non-redundant speech frame per RTP packet Max 4 or 12 speech frames in total depending on whether redundancy is supported but not more than a received maxptime value requires | 20 | 80 | 240 |
| HSPA E-UTRAN NR | 1 non-redundant speech frame per RTP packet Max 4 or 12 speech frames in total depending on whether redundancy is supported but not more than a received maxptime value requires | 20 | 80 | 240 |
| EGPRS | 2 non-redundant speech frames per RTP packet but not more than a received maxptime value requires Max 4 or 12 speech frames in total depending on whether redundancy is supported but not more than a received maxptime value requires | 40 | 80 | 240 |
| GIP | 1 to 4 non-redundant speech frames per RTP packet but not more than a received maxptime value requires Max 12 speech frames in total but not more than a received maxptime value requires | 20, 40, 60 or 80 | N/A | 240 |

When the access technology is not known to the MTSI media gateway, the default encapsulation parameters defined in Table 12.1 shall be used.

The SDP offer shall include an RTP payload type where octet-align=0 is defined or where the octet-align parameter is not specified and should include another RTP payload type with octet-align=1. MTSI media gateways offering wide-

band speech shall offer these parameters and parameter settings also for the RTP payload types used for wide-band speech.

MTSI media gateways should support the RTCP-APP signalling defined in clause 10.2.1. The Codec Mode Request (RTCP_APP_CM) is only relevant when AMR or AMR-WB is used but the Redundancy Request and the Frame Aggregation Request can be used for all codecs. When RTCP-APP is not supported or cannot be used in the session then adaptation can also be based on RTCP Receiver Reports/Sender Reports.

MTSI media gateways should support redundancy according to clause 9.

NOTE: Support of transmitting redundancy may be especially useful in the case an MTSI media gateway is aware of the used access technology and knows that the Generic Access technology is used.

12.3.2.2 Text

Both CTM according to TS 26.226 [52] and RFC 4103 make use of ITU-T Recommendation T.140 presentation and character coding. Therefore inter-working is a matter of payload packetization and CTM modulation/demodulation.

12.3.3 Explicit Congestion Notification

An MTSI MGW can be used to enable ECN between the MTSI client in terminal and the MTSI MGW when inter-working with CS GERAN/UTRAN.

If ECN is supported in the MTSI MGW, then the MTSI MGW shall also:

- support ECN as described in this specification for the MTSI client in terminal, except that the MTSI MGW does not determine whether ECN can be used based on the Radio Access Technology that is used towards the MTSI client in terminal;
- support RTP/AVPF and SDPCapNeg if the MTSI MGW supports RTCP AVPF ECN feedback messages;
- be capable of enabling end-to-end rate adaptation between the MTSI client in terminal and the CS terminal by performing the following:
 - negotiate the use of ECN with the MTSI client in terminal, if it can be confirmed that the network used towards the MTSI client in terminal properly handles ECN-marked packets;
 - inter-work adaptation requests between the MTSI client in terminal and the CS GERAN/UTRAN;

12.3.4 Codec switching procedures with SRVCC

An MTSI client in terminal (hereinafter "local client") using 3GPP PS access may be handed over to CS access. By that SRVCC procedure, the end-point of the IP connection moves from the local client to a CS MGW in the CS network, as described in TS 23.216 (SRVCC) [133].

In order to achieve this handover, the MSC server, controlling the CS MGW, sends a SIP INVITE message:

- either to the remote client (in case of SRVCC handover without SRVCC enhancement);
- or to the ATCF (in case of SRVCC handover with ATCF enhancement),

to change the communication end from the MTSI client in terminal to the CS MGW as described in TS 23.237 [134].

If EVS is used between local and remote client before SRVCC and if AMR-WB is used after SRVCC by the local CS UE, an MTSI MGW (e.g. MSC/CS-MGW or ATCF/ATGW) can send the RTCP_APP_EP2I request message, (see clause 10.1.2.10), or a CMR in the RTP payload requesting an EVS AMR-WB IO mode, to the remote client to request that it switches from the EVS Primary mode to the EVS AMR-WB IO mode. The mode-set used in CS shall be included in the RTCP_APP_EP2I request message. Furthermore, the RTCP_APP_EP2I request message also supports signalling to restrict the timing and destination of codec mode changes. An SDP offer/answer negotiation between the MTSI MGW and the remote client can also be performed to align the mode-sets and to optimize the resource usage and also to request switching to the EVS AMR-WB IO mode.

Correspondingly, the RTCP_APP_EI2P request message can be used to switch from the EVS AMR-WB IO mode to the EVS Primary mode, e.g. in case an SRVCC handover to a CS access and a switch to the EVS AMR-WB IO mode is followed by a reverse SRVCC to perform handover back to the PS access. An SDP offer/answer negotiation can also be performed to restore the session, e.g. bitrates, bandwidths and other configuration parameters, to what was used before SRVCC.

NOTE: The DTX operation of EVS Primary and AMR-WB IO may be configured in sending direction with either a fixed SID update interval (from 3 to 100 frames) or an adaptive SID update interval – more details can be found in clauses 4.4.3 and 5.6.1.1 of TS 26.445 [125]. The DTX operation of AMR-WB is defined with a fixed interval of 8 frames for SID updates. Implementers of MTSI MGWs are advised to take into account the SID flexibility of EVS (with respect to AMR-WB) for the interworking between AMR-WB and EVS AMR-WB IO.

12.4 PSTN

12.4.1 3G-324M

If 3G-324M is supported in the PSTN, then the inter-working can be made as specified in clause 12.2.

12.4.2 Text

PSTN text telephony inter-working with PS environments is described in ITU-T Recommendation H.248.2 [50] and further elaborated in EG 202 320 [51].

Text telephony modem tones are sensitive to packet loss, jitter and echo canceller behaviour. Therefore, conversion of modem based transmission of real-time text is best done at the border of the PSTN. If PSTN text telephone tones need to be carried audio coded in a PS network, considerations must be taken to carry them reliably as for example specified in ITU-T Recommendations V.151 [54] and V.152 [55].

When inter-working with PSTN text telephones, it must be considered that in PSTN most text telephone communication methods do not allow simultaneous speech and text transmission. An MTSI client in terminal indicating text capability shall not automatically initiate text connection efforts on the PSTN circuit. Instead, either a requirement for text support should be required from the MTSI client in terminal, active transmission of text from the MTSI client in terminal, or active transmission of text telephone tones from the PSTN terminal. See clause 13 of EG 202 320 [51].

Note that the primary goal of real-time text support in MTSI is not to offer a replica of PSTN text telephony functionality. On the contrary, real-time text in MTSI is aiming at being a generally useful mainstream feature, complementing the general usability of the Multimedia Telephony Service for IMS.

12.5 GIP inter-working

12.5.1 Text

RFC 4103 [31] and T.140 are specified as default real-time text codec in SIP telephony devices in RFC 4504 [53]. When GIP implements this codec, the media stream contents are identical for the two environments. Packetization will also in many cases be equal, while consideration must be taken to cope with different levels of redundancy and possible use of different media security and integrity measures.

12.5.2 Speech

See Clause 12.7.

12.6 Void

12.6.1 Void

12.6.2 Void

12.7 Inter-working with other IMS and non-IMS IP networks

12.7.1 General

IMS and MTSI services are required to support inter-working with similar services operating on other IP networks, both IMS based and non-IMS based, [2]. It is an operator option to provide transcoding when the end-to-end codec negotiation fails to agree on a codec to be used for the session. The requirements herein apply to MTSI MGWs when such transcoding is provided.

These requirements were designed for sessions carried with IP end-to-end, possibly inter-connected through one or more other IP networks.

A main objective is to harmonize the requirements for this inter-working case with the requirements for GERAN/UTRAN CS inter-working defined in Clause 12.3. There is however one major difference as the MGW requirements in Clause 12.3 apply only to the PS side of the MTSI MGW, i.e. between the MTSI MGW and the MTSI client in the terminal, while here there are requirements for the MTSI MGW both towards the MTSI client in the terminal and towards the remote network.

Most requirements included here apply only to the PS access towards the remote network but there are also requirements that target both the local MTSI client in terminal and the remote network or even only the local MTSI client.

12.7.2 Speech

12.7.2.1 General

This clause defines how speech media should be handled in MTSI MGWs in inter-working scenarios between an MTSI client in terminal using 3GPP access and a non-3GPP IP network and between an MTSI client in terminal using fixed access and a non-3GPP IP network. This clause therefore defines requirements for what the MTSI MGW needs to support and how it should behave during session setup and session modification. A few SDP examples are included in Annex A.10.

12.7.2.2 Speech codecs and formats

12.7.2.2.1 MTSI MGW for interworking between MTSI client in terminal using 3GPP access and other IMS or non-IMS IP networks

This clause applies to MTSI MGWs used for interworking between an MTSI client in terminal using 3GPP access and a client using another IMS or non-IMS IP network.

MTSI MGWs offering speech communication between an MTSI client in a terminal and a client in another IP network through a Network-to-Network Interface (NNI) using AMR shall support:

- AMR speech codec modes 12.2, 7.4, 5.9 and 4.75 [11], [12], [13], [14] and source-controlled rate operation [15], both towards the local MTSI client in terminal and towards the remote network;
- G.711, both A-law and μ -law PCM, [77], towards the remote network.

and should support:

- linear 16 bit PCM (L16) at 8 kHz sampling frequency, towards the remote network.

When such MTSI MGWs also offer wideband speech communication using AMR-WB they shall support:

- AMR wideband codec 12.65, 8.85 and 6.60 [17], [18], [19], [20] and source controlled rate operation [21] , both towards the local MTSI client in terminal and towards the remote network;

and should support:

- G.722 (SB-ADPCM) at 64 kbps, [78], towards the remote network; and
- linear 16 bit PCM (L16) at 16 kHz sampling frequency, towards the remote network.

NOTE: A TrGW decomposed from an IBCF can also be media-unaware and forward any media transparently without changing the encoding. Transcoding support is optional at the Ix interface.

12.7.2.2.2 MTSI MGW for interworking between MTSI client in terminal using fixed access and other IMS or non-IMS IP networks

This clause applies to MTSI MGWs used for interworking between an MTSI client in terminal using fixed access and a client using another IMS or non-IMS IP network.

Media codecs for MTSI MGWs for speech inter-working between fixed access and IP clients in other IMS or non-IMS IP networks are specified in TS 181 005 [98] in clause 6.2 for narrow-band codecs and in clause 6.3 for wide-band codecs. In addition, the MTSI MGW should support linear 16 bit PCM (L16) at 8 kHz sampling frequency for narrow-band speech. An MTSI MGW supporting wideband speech should also support linear 16 bit PCM (L16) at 16 kHz sampling frequency.

MTSI MGWs for speech inter-working between access and CS GERAN/UTRAN supporting AMR and AMR-WB shall follow clause 12.7.2.2.2 for the AMR and AMR-WB codecs. Tandem-free inter-working should be used whenever possible.

For the other codecs, the MTSI MGW shall follow the recommendations and requirements defined in clause 18 for the respective codec.

12.7.2.2.3 Common procedures

If the remote network supports AMR for narrowband speech and/or AMR-WB for wideband speech, then transcoding shall be avoided whenever possible. In this case, the MTSI MGW should not be included in the RTP path unless it is required for non transcoding related purposes. If the MTSI MGW is included in the RTP path then it shall support forwarding the RTP payload regardless of codec mode and packetization.

NOTE: An example of where transcoding may be required when AMR and/or AMR-WB are supported by the remote network is when the remote terminal is limited to modes that are not supported by the local MTSI client in terminal due to operator configuration.

If the MTSI MGW is performing transcoding of AMR or AMR-WB then it shall be capable of restricting mode changes, both mode change period and mode changes to neighboring mode, if this is required by the remote network.

Requirements applicable to MTSI MGW for DTMF events are described in Annex G.

12.7.2.3 Codec preference order for session negotiation

It is important to optimize the quality-bandwidth compromise, even though the NNI uses a fixed IP network. For this reason, the following preference order should be used by MTSI MGWs unless another preference order is defined in bilateral agreements between the operators or configured otherwise by the operator:

- The best option is if a codec can be used end-to-end. For example, using AMR or AMR-WB end-to-end is preferable over transcoding through G.711 or G.722 respectively.
- The second best solution is to use G.711 or G.722 as inter-connection codecs, for narrow-band and wide-band speech respectively, since these codecs offer a good quality while keeping a reasonable bit rate.
- The linear 16 bit PCM format should only be used as the last resort, when none of the above solutions are possible.

If a wide-band speech session is possible, then fall-back to narrow-band speech should be avoided whenever possible, unless another preference order is indicated in the SDP.

NOTE: There may be circumstances, for example bit rate constraints, when a fall-back to narrow-band speech is acceptable since the alternative would be a session setup failure.

12.7.2.4 RTP profiles

MTSI MGWs offering speech communication over the NNI shall support the RTP/AVP profile and should support the RTP/AVPF profile, [40]. If the RTP/AVPF profile is supported then the SDP Capability Negotiation (SDPCapNeg) framework shall also be supported, [69].

An MTSI MGW supporting EVS should support the RTCP-APP signalling for speech adaptation defined in clause 10.2.1.

12.7.2.5 RTP payload formats

The payload format to be used for AMR and AMR-WB encoded media is defined in Clause 12.3.2.1. The payload format to be used for EVS encoded media is defined in [125]. The MTSI MGW shall support the following payload SDP parameters for AMR and AMR-WB: octet-align, mode-set, mode-change-period, mode-change-capability, mode-change-neighbor, maxptime, ptime, channels and max-red.

The payload format to be used for G.711 encoded media is defined in RFC 3551, [10], for both μ -law (PCMU) and A-law (PCMA).

The payload format to be used for G.722 encoded media is defined in RFC 3551, [10].

NOTE: The sampling frequency for G.722 is 16 kHz but is set to 8000 Hz in SDP since it was (erroneously) defined this way in the original version of the RTP A/V profile, see [10].

The payload format to be used for linear 16 bit PCM is the L16 format defined in RFC 3551, [10]. When this format is used for narrow-band speech then the rate (sampling frequency) indicated on the `a=rtpmap` line shall be 8000. When this format is used for wide-band speech then the rate (sampling frequency) indicated on the `a=rtpmap` line shall be 16000.

The payload formats to be used for the other codecs are listed in Clause 18.4.3.

12.7.2.6 Packetization

For the G.711, G.722 and linear 16 bit PCM formats, the frame length shall be 20 ms, i.e. 160 and 320 speech samples in each frame for narrow-band and wide-band speech respectively.

MTSI MGWs offering speech communication over the NNI shall support encapsulating up to 4 non-redundant speech frames into the RTP packets.

MTSI MGWs may support application layer redundancy. If redundancy is supported then the MTSI MGW should support encapsulating up to 8 redundant speech frames in the RTP packets. Thereby, an RTP packet may contain up to 12 frames, up to 4 non-redundant and up to 8 redundant frames.

An MTSI MGW setting up a speech session should align the ptime and maxptime between the networks so that the same packetization can be used end-to-end, even when transcoding is used.

The MGW should use the packetization schemes indicated by the ptime value in the SDP offer and answer. If no ptime value is present in the SDP then the MGW should encapsulate 1 frame per packet or the packetization used by the end-point clients.

The MTSI MGW should preserve the packetization used by the end-point clients to minimize the buffering times otherwise caused by jitter. For example, if one end-point adapts the packetization to use 2 frames per packet then the MTSI MGW should adapt the packetization to the other end-point to also use 2 frames per packet. This applies also when the MTSI MGW performs transcoding. The packet size can become quite large for some combinations of formats and packetization. If the packet size exceeds the Maximum Transfer Unit (MTU) of the network then the MTSI MGW should encapsulate fewer frames per packet.

NOTE: It is an implementation consideration to determine the MTU of the network. RFC 4821 [79] describes one method that can be used to discover the path MTU.

When the MTSI MGW does not perform any transcoding then it shall be transparent to the packetization schemes used by the end-point clients.

12.7.2.7 RTCP usage and adaptation

The RTP implementation shall include an RTCP implementation.

MTSI MGWs offering speech should support AVPF (RFC 4585 [40]) configured to operate in early mode. When allocating RTCP bandwidth, it is recommended to allocate RTCP bandwidth and set the values for the "b=RR:" and the "b=RS:" parameters such that a good compromise between the RTCP reporting needs for the application and bandwidth utilization is achieved, see also SDP examples in Annex A.10. When an MTSI MGW uses tandem-free inter-working between two PS networks then it should align the RTCP bandwidths such that RTCP packets can be sent with the same frequency in both networks. This is to allow for sending adaptation requests end-to-end without being forced to buffer the requests in the MTSI MGW. The value of "trr-int" should be set to zero or not transmitted at all (in which case the default "trr-int" value of zero will be assumed) when Reduced-Size RTCP (see clause 7.3.6) is not used.

For speech sessions, between the MTSI client in terminal and the MTSI MGW, it is beneficial to keep the size of RTCP packets as small as possible in order to reduce the potential disruption of RTCP onto the RTP stream in bandwidth-limited channels. RTCP packet sizes can be minimized by using Reduced-Size RTCP packets or using the parts of RTCP compound packets (according to RFC 3550 [9]) which are required by the application.

The MTSI MGW shall be capable of adapting the session to handle possible congestion. For AMR and AMR-WB encoded media, the MTSI MGW shall support the adaptation signalling method using RTCP APP packets as defined in clause 10.2, both towards the MTSI client in terminal and towards the remote network. As the IP client in the remote network may or may not support the RTCP APP signalling method, the MTSI MGW shall also be capable of using the inband CMR in the AMR payload. When receiving inband CMR in the payload from the remote network, the MTSI MGW does not need to move the adaptation signalling to RTCP APP packets before sending it to the MTSI client in terminal.

For PCM, G.722 and linear 16 bit PCM encoded media, the MTSI MGW shall support RFC 3550 for signalling the experienced quality using RTCP Sender Reports and Receiver Reports.

For a given RTP based media stream to/from the MTSI client in terminal, the MTSI MGW shall transmit RTCP packets from and receive RTCP packets to the same port number.

For a given RTP based media stream to/from the remote network, the MTSI MGW shall transmit RTCP packets from and receive RTCP packets on the same port number, not necessarily the same port number as used to/from the MTSI client in terminal.

This facilitates inter-working with fixed/broadband access. However, the MTSI MGW may, based on configuration or local policy, accept RTCP packets that are not received from the same remote port where RTCP packets are sent by either the MTSI client in terminal or the remote network.

12.7.2.8 RTP usage

For AMR and AMR-WB encoded media, the MTSI MGW shall follow the same requirements when inter-working with other IP network as when inter-working with GERAN/UTRAN CS, see clause 12.3.2.1.

For a given RTP based media stream to/from the MTSI client in terminal, the MTSI MGW shall transmit RTP packets from and receive RTP packets to the same port number.

For a given RTP based media stream to/from the remote network, the MTSI MGW shall transmit RTP packets from and receive RTP packets on the same port number, not necessarily the same port number as used to/from the MTSI client in terminal.

This facilitates inter-working with fixed/broadband access. However, the MTSI MGW may, based on configuration or local policy, accept RTP packets that are not received from the same remote port where RTP packets are sent by either the MTSI client in terminal or the remote network.

12.7.2.9 Session setup and session modification

The MTSI MGW shall be capable of dynamically adding and dropping speech media during the session.

The MTSI MGW may use the original SDP offer received from the MTSI client in terminal when creating an SDP offer that is to be sent outbound to the remote network.

If the MTSI MGW adds codecs to the SDP offer then it shall follow the recommendations of Clause 12.7.2.3 when creating the outbound SDP offer and when selecting which codec to include in the outbound SDP answer.

If the MTSI MGW generates an SDP offer based on the offer received from the MTSI client in terminal, it should maintain the ptime and maxptime values as indicated by the MTSI client in terminal. If the MTSI MGW generates an SDP offer without using the SDP offer from the MTSI client in terminal then it should define the ptime and maxptime values in accordance in Clause 12.7.2.6, i.e. the preferred values for ptime and maxptime are 20 and 80 respectively.

If the MTSI MGW does not support AVPF (nor SDPCapNeg) then it shall not include the corresponding lines in the SDP offer that is sent to the remote network.

12.7.2.10 Audio level alignment

In case of interworking, the audio levels should be aligned to ensure suitable audio levels to the end users. This is especially important when codecs with different overload points are used on each side of the MTSI MGW as this can result in an asymmetrical loudness between the end points.

NOTE 1: The overload point of a given codec refers to the adjustment factor between the digital levels in input/output of this codec and the resulting acoustic levels. In practice the overload point value corresponds to the analog Root Mean Square (RMS) level of a full-scale sinusoidal signal.

For MTSI client in terminal using fixed access, clause 18.8 applies to ensure proper audio alignment.

For communications requiring interworking with other IMS or non-IMS IP networks, terminals connected to these networks may use different codecs, which have different overload points. In this case, it is recommended that the MTSI MGW doing transcoding ensure proper audio level alignment. This alignment shall be performed such that the nominal level is preserved (0 dBm0 shall be maintained to 0 dBm0). As an example, a fixed CAT-IQ DECT terminal implementing G.722 with a 9 dBm0 overload point as recommended in ITU-T Recommendation G.722 [78] might need some audio level alignment in case of wideband voice interworking with a 3GPP terminal using AMR-WB with a 3.14 dBm0 overload point. The audio level alignment may use dynamic range control to prevent saturation or clipping.

NOTE 2: The definition of the dBm0 unit can be found in ITU-T P.10 [108].

12.7.3 Explicit Congestion Notification

An MTSI MGW can be used to enable ECN within the local network when the local ECN-capable MTSI client in terminal is in a network that properly handles ECN-marked packets, and either the remote network cannot be confirmed to properly handle ECN-marked packets or the remote terminal does not support or use ECN.

If ECN is supported in the MTSI MGW, then the MTSI MGW shall also:

- support RTP/AVPF and SDPCapNeg if the MTSI MGW supports RTCP AVPF ECN feedback messages;
- be capable of enabling end-to-end rate adaptation between the local MTSI client in terminal and the remote client by performing the following towards the local MTSI client in terminal:
 - negotiate the use of ECN;
 - support ECN as described in this specification for the MTSI client in terminal, except that the MTSI MGW does not determine whether ECN can be used based on the Radio Access Technology.

NOTE: The adaptation requests are transmitted between the local and the remote client without modification by the MTSI MGW.

An MTSI MGW can also be used to enable ECN end-to-end if the remote client uses ECN in a different way than what is described in this specification for the MTSI client in terminal, e.g. if the remote client only supports probing for the ECN initiation phase or it needs the RTCP AVPF ECN feedback messages.

12.7.4 Text

The codec and other considerations for real-time text described in the present document for MTSI clients in terminal using 3GPP access apply also to MTSI clients in terminal using fixed access. There are thus no inter-working considerations on the media level between these types of end-points.

12.7.5 Inter-working IPv4 and IPv6 networks

If different IP versions are used by the offerer and the answerer, information in the SDP offer or answer related to IP version and QoS negotiation should be modified appropriately by the MTSI MGW so that the offerer and the answerer agree with an identical or similar source bit-rates.

For video, $b=AS$ in IPv6 should be assumed to be a product of $b=AS$ in IPv4 and 1.04, rounded down to a nearest integer, when other information that can be used to re-compute $b=AS$ in IPv6 from $b=AS$ in IPv4 is not present. Likewise, $b=AS$ in IPv4 should be assumed to be a product of $b=AS$ in IPv6 and 0.96, rounded up to a nearest integer. These formulas meet the relationship of $b=AS$ values for 176×144 and 320×240 in Table N.x. Depending on service policy or codec configuration, other formulas can be used.

An MTSI MGW for interworking between IPv4 and IPv6 networks supporting the 'a=bw-info' attribute (see clause 19) shall re-compute the bandwidth properties signalled with this attribute if only bandwidths for either IPv4 or IPv6 are present. If bandwidth properties are provided with values for both IPv4 and IPv6 then the MTSI MGW should not re-compute the bandwidths.

12.8 MGW handling for NO_REQ interworking

The meaning of "none" and "NO_REQ" for EVS (as specified in TS 26.445 [125] is not equivalent to code-point "CMR=15" for AMR and AMR-WB (as specified according to TS 26.114 and RFC 4867 with its errata):

- For AMR-WB, CMR=15 overrides the previously received CMR value (corresponding to a speech mode or CMR=15). In other words, when an MTSI client receives CMR15 it is no longer restricted for its outbound packets by the previously received CMR, however it still complies with the negotiated mode-set.
- For EVS, the 'NO_REQ' and 'none' CMR code points mean that there is no request and this CMR value shall be ignored. In other words, when an MTSI client receives NO_REQ or 'none' for EVS it is still restricted for its outbound packets by the previously received CMR (if any) and in addition it still complies within the negotiated codec operation modes.

MGWs in the path, repacking between the RTP format according to RFC 4867 [28] and the EVS RTP format in TS 26.445 [125] shall translate between these code-points (in transcoder-free operation):

- When translating a single frame per packet from AMR-WB to EVS (AMR-WB IO): CMR=15 shall be replaced by the highest possible of EVS AMR-WB IO allowed in the session.
- When translating a single frame per packet from EVS (AMR-WB IO) to AMR-WB: NO_REQ and none shall be replaced by the previously sent CMR (or the highest possible of AMR-WB allowed in the session if no request has been sent since the beginning of the session).
- When translating more than one frame per packet (e.g. from 1 frame per packet to 2 frames per packets or vice versa), the MGW may have to "combine" or "repeat" CMRs following same translation as for the single frame per packet when applicable.

The above translation rules apply except when MGW wants to change the CMR. An example is when a MGW detects problems at an early stage in uplink which may require the MGW to send a CMR to limit bitrate at a lower value than the incoming CMR from the remote media receiver.

NOTE: When EVS AMR-WB IO is not used (transcoder-free operation is not possible), the speech path is split into two links (AMR-WB and EVS) and the adaptation on these two links are independent from each other. CMR translation between AMR-WB and EVS at the MGW is therefore not required.

13 Void

13a Media types, codecs and formats used for MSRP transport

13a.1 General

The IMS messaging service is described in TS 26.141 [59]. The description of IMS messaging in clauses 1-6 of TS 26.141 [59] is applicable for MSRP-transported media in MTSI. The MSRP transport itself is described in TS 24.247 [82].

All statements in TS 26.141 regarding IMS messaging are valid for MSRP transported media in MTSI including the status of the statement (shall, should, may).

Any differences between IMS messaging in TS 26.141 [59] and MSRP transported media in MTSI are described in clause 13a.2.

13a.2 Difference relative to 3GPP TS 26.141

13a.2.1 Video

For MSRP transported Media in MTSI, clause 5.2.2 of this specification applies.

14 Supplementary services

14.1 General

In this section media layer behaviour is specified for relevant supplementary services. The supplementary services included in MTSI are described in TS 24.173 [57]. The requirements on the codec support and the data transport are identical to those listed in clauses 5.2 and 7. These requirements are listed here due to the fact that there might be other media-influencing nodes in MTSI whose behaviour is not explicitly covered by other parts of the present document.

The recommended behaviour described in the following sections is valid for MTSI clients, i.e. all session IP end-points; terminals, MTSI media gateways and other 3GPP network nodes acting as IP endpoints in MTSI sessions.

14.2 Media formats and transport

Any implementation of a supplementary service which affects media or media handling, e.g. such as media creation, media rendering and media manipulation, shall meet the same requirements as a MTSI client in terminal regarding codec support and codec usage. Where applicable, speech codecs shall be supported according to clause 5.2.1, video according to clause 5.2.2 and text according to clause 5.2.3.

Similarly, the configuration and the transport of the media in any implementation of a supplementary service which affects media or media handling shall be done according to clause 7.

14.3 Media handling in hold procedures

Whenever a supplementary service includes a hold procedure according to RFC 3264 [58], e.g. when using the HOLD supplementary service, the media flow is changed in terms of the session flow attribute (e.g. changing the session attribute "sendrecv" into "sendonly" or "recvonly" or "inactive" and then back again). When this occurs, any involved media-originating or media-terminating node should take measures to ensure that the transitions between the different media flow states in the session occur with minimal impact on the media quality.

When a full-duplex session has put the media flow on hold (see section 8.4 in RFC 3264 [58]), the media flow has been changed into a unidirectional flow through changing the session attribute into either "sendonly" or "recvonly". When resuming the session, it is restored to full duplex by changing the flow attributes back into "sendrecv" from "sendonly" and "recvonly". In this case, the encoder and decoder states in the MTSI clients may not be aligned and a state mismatch could occur. This would result in media quality degradation. Therefore, the following actions are recommended whenever the media session is not being put on hold anymore and the session is restored to full duplex:

- for speech media, the speech decoders should be reset;
- for video media, the video encoders should start the updated session with a full infra refresh even if the previously allocated encoders are still active and no infra refresh is scheduled to be sent.

15 Network preference management object

15.1 General

The MTSI client in the terminal may use the OMA-DM solution specified in this clause for enhancing the SDP negotiation and resource reservation process. If a MTSI client in the terminal uses this feature, it is mandatory for the MTSI client in the terminal to implement the Management Object (MO) as described in this clause.

The 3GPP MTSINP (MTSI Network Preference) MO defined in this clause may be used to manage the QoS profile settings which express the network preference for the MTSI client in the terminal. The MO covers parameters that the MTSI client in the terminal could make use of in SDP negotiation and resource reservation process. If a MTSI client in the terminal supports the feature, the usage of the MO includes:

1. During SDP negotiation process, MTSI client in the terminal should start SDP negotiation based on the MO parameters.
2. During resource reservation process, MTSI client in the terminal should start QoS negotiation based on the MO parameters.

The following parameters in MTSI should be included in the Management Object (MO):

Speech codec (AMR, AMR-WB, EVS) and bearer QoS parameters

Video codec (H.264 (AVC), H.265 (HEVC)) and bearer QoS parameters

Real Time text bearer QoS parameters

Indication of the priority when there are more than one alternative for a media type is included. Version numbering is included for possible extending of MO.

The Management Object Identifier shall be: urn:oma:mo:ext-3gpp-mtsinp:1.0.

Protocol compatibility: The MO is compatible with OMA Device Management protocol specifications, version 1.2 and upwards, and is defined using the OMA DM Device Description Framework as described in the Enabler Release Definition OMA-ERELED_DM-V1_2[67].

15.2 Nodes Definition

The following nodes and leaf objects in figure 15.1 shall be contained under the 3GPP_MTSINP node if a MTSI client in the terminal support the feature described in this clause (information of DDF for this MO is given in Annex H):

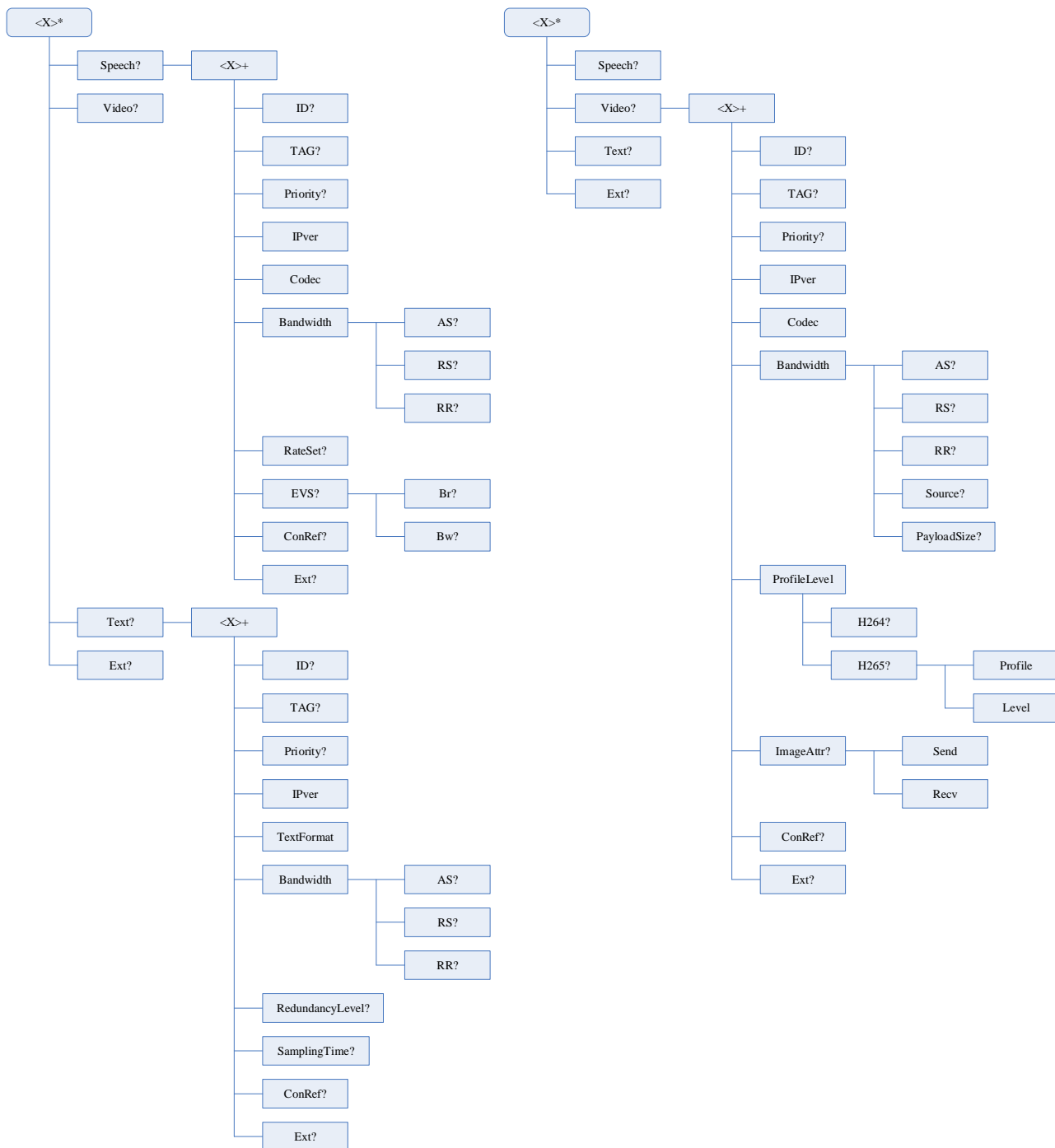


Figure 15.1: MTSI network preference management object tree

Node: /<X>

This interior node specifies the unique object id of a MTSI network preferences management object. The purpose of this interior node is to group together the parameters of a single object.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

The following interior nodes shall be contained if the MTSI client in the terminal supports the "MTSI network preferences Management Object".

/ $\langle X \rangle$ /Speech

The Speech node is the starting point of the speech codec definitions (if any speech codec are available)

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$

This interior node is used to allow a reference to a list of speech codec objects.

- Occurrence: OneOrMore
- Format: node
- Minimum Access Types: Get

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /ID

This leaf node represents the identification number of a set of parameters for speech session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /TAG

This leaf node represents the identification tag of a set of parameters for speech session. It is recommended to have at least a node, for example, ID, TAG, or implementation-specific ones, for the identification purpose such that each set of parameters can be distinguished and accessed.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /Priority

This leaf represents the priority of a set of parameters for speech session. Lower value means higher priority and the value is used in the terminal for client initiated QoS handling. The priority uses a 16 bit unsigned integer.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get
- Values: Zero or higher

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /IPver

This leaf represents the version of the Internet Protocol used in the session.

- Occurrence: One
- Format: chr

- Minimum Access Types: Get
- Values: "IPv4", "IPv6"

/<X>/Speech/<X>/Codec

This leaf gives the MIME subtype name of speech codec. This leaf is preferably pre-configured by the device.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get
- Values: MIME subtype name of speech codec, e.g., "AMR", "AMR-WB", "EVS".

The value "AMR" refers to the AMR speech codec as defined in 3GPP. The value "AMR-WB" refers to the AMR-WB speech codec as defined in 3GPP. The value "EVS" refers to the EVS speech codec as defined in 3GPP.

/<X>/Speech/<X>/Bandwidth

This interior node is used to allow a reference to a list of parameters related to speech bandwidth assignment.

- Occurrence: One
- Format: node
- Minimum Access Types: Get
- Values: positive integer

/<X>/Speech/<X>/Bandwidth/AS

This leaf gives the preferred speech codec bandwidth by the network for the bearer set-up, including RTP/UDP/IP headers. It provides the value for "b=AS" line for speech part used in the end-to-end SDP negotiation process, which represents the bit rate in kbits/sec.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/Bandwidth/RS

This leaf provides the value for "b=RS" line for speech part used in the end-to-end SDP negotiation process, which represents the bit rate in bits/sec.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/Bandwidth/RR

This leaf provides the value for "b=RR" line for speech part used in the end-to-end SDP negotiation process, which represents the bit rate in bits/sec.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/RateSet

This leaf node represents a list of bit rates used by speech codec. Depending on the codec, each value can be understood as either the highest rate or the average rate. The entries in the list may either be generic, i.e., usable for any codec, but can also be codec-specific. The default usage is the generic list where the bit rates in bits/sec are included, e.g., "5000, 6000, 7500, 12500". A codec-specific list may indicate the desired modes. For example, in the case of AMR, the list could be "0, 2, 4, 7".

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/EVS

This interior node is used to allow a reference to a list of parameters related to the configuration of EVS speech codec.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/EVS/Br

This leaf gives the value of br, a parameter representing the range or value of bit-rate for EVS speech codec defined in [125].

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/EVS/Bw

This leaf gives the value of bw, a parameter representing the range or value of bandwidth for EVS speech codec defined in [125].

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/ConRef

This node specifies a reference to QoS parameters Management Object. The interior node's leaf nodes specify the network preferred QoS parameters as defined in TS 24.008 and they should be used in the bearer request when client initiated QoS happen. Implementation specific MO may be referenced.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video

The Video node is the starting point of the video codec definitions (if any video codec are available)

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`

This interior node is used to allow a reference to a list of video codec objects.

- Occurrence: OneOrMore
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ID

This leaf node represents the identification number of a set of parameters for video session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/TAG

This leaf node represents the identification tag of a set of parameters for video session. It is recommended to have at least a node, for example, ID, TAG, or implementation-specific ones, for the identification purpose such that each set of parameters can be distinguished and accessed.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/Priority

This leaf represents the priority of a set of parameters for speech session. Lower value means higher priority and the value is used in the terminal for client initiated QoS handling. The priority uses a 16 bit unsigned integer.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get
- Values: Zero or higher

/`<X>`/Video/`<X>`/IPver

This leaf represents the version of the Internet Protocol used in the session.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get
- Values: "IPv4", "IPv6"

/`<X>`/Video/`<X>`/Codec

This leaf gives the MIME subtype name of video codec. This leaf is preferably pre-configured by the device.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get
- Values: MIME subtype name of video codec, e.g., "H264", "H265".

The values "H264" and "H265" refer to the H.264 (AVC) and H.265 (HEVC) codecs as defined by MPEG and ITU respectively. The usage of H.264 (AVC) and H.265 (HEVC) codecs (profiles, levels etc) is described in the document TS 26.114 Chapter 5.5.2.

/`<X>`/Video/`<X>`/Bandwidth

This interior node is used to allow a reference to a list of parameters related to video bandwidth assignment.

- Occurrence: One
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/Bandwidth/AS

This leaf gives the preferred video codec bandwidth by the network for the bearer set-up, including RTP/UDP/IP headers. It provides the value for "b=AS" line for video part used in the end-to-end SDP negotiation process, which represents the bit rate in kbits/sec.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/Bandwidth/RS

This leaf provides the value for "b=RS" line for video part used in the end-to-end SDP negotiation process, which represents the bit rate in bits/sec.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/Bandwidth/RR

This leaf provides the value for "b=RR" line for video part used in the end-to-end SDP negotiation process, which represents the bit rate in bits/sec.

- Occurrence: ZeroOrOne
- Format: int

- Minimum Access Types: Get

/`<X>`/Video/`<X>`/Bandwidth/Source

This leaf gives the preferred video encoding bandwidth in kbits/sec.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/Bandwidth/PayloadSize

This leaf gives the preferred payload size for video, excluding payload header, which represents the amount of encoded video data in bytes transported over a RTP packet.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ProfileLevel

This interior node is used to allow a reference to a list of parameters related to the profile and level of video codec.

- Occurrence: One
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ProfileLevel/H264

This leaf gives the profile-level-id of H.264 (AVC) video codec, which indicates the profile that the codec supports and the highest level supported for the signaled profile [24], [25].

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ProfileLevel/H265

This interior node is used to allow a reference to a list of parameters related to the profile and level of H.265 (HEVC) video codec.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ProfileLevel/H265/Profile

This leaf gives the value of profile-id, a parameter representing the profile of H.265 (HEVC) video codec defined in [119], [120].

- Occurrence: One
- Format: int
- Minimum Access Types: Get

/~~X~~/Video/~~X~~/ProfileLevel/H265/Level

This leaf gives the value of level-id, a parameter representing the level of H.265 (HEVC) video codec defined in [119], [120]. Level indicates the maximum computational complexity supported by the offerer in performing decoding for the given profile.

- Occurrence: One
- Format: int
- Minimum Access Types: Get

/~~X~~/Video/~~X~~/ImageAttr

This interior node is used to allow a reference to a list of parameters related to the image sizes supported or preferred, specified with the "imageattr" attribute. (see clause A.4)

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/~~X~~/Video/~~X~~/ImageAttr/Send

This leaf gives the supported image sizes for the send direction. The value is a string such as "176, 144, 224, 176, 272, 224, 320, 240" which means four image sizes, 176x144, 224x176, 272x224, and 320x240 are supported for the send direction. The maximum image size in this leaf shall not exceed the maximum size limited by the offered codec level.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get

/~~X~~/Video/~~X~~/ImageAttr/Recv

This leaf gives the supported image sizes and their preferences for the receive direction. The value is a string such as "176, 144, 0.5, 224, 176, 0.5, 272, 224, 0.6, 320, 240, 0.5" which means four image sizes, 176x144, 224x176, 272x224, and 320x240 are supported for the receive direction but 272x224 is preferred since it might fit the available space on the display of the receiver better than the other image sizes. The maximum image size in this leaf shall not exceed the maximum size limited by the offered codec level. The value representing the level of preference by the offerer, defined in [76], is between 0 and 1 inclusive and 0.5 by default.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get

/~~X~~/Video/~~X~~/ConRef

This node specifies a reference to QoS parameters Management Object. The interior node's leaf nodes specify the network preferred QoS parameters as defined in TS 24.008 and they should be used in the bearer request when client initiated QoS happen. Implementation specific MO may be referenced.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/~~X~~/Video/~~X~~/Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Text

The Text node is the starting point of the real time text codec definitions (if the real time text codec is available).

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Text/`<X>`

This interior node is used to allow a reference to the real time text codec objects.

- Occurrence: OneOrMore
- Format: node
- Minimum Access Types: Get

/`<X>`/Text/`<X>`/ID

This leaf node represents the identification number of a set of parameters for text session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Text/`<X>`/TAG

This leaf node represents the identification tag of a set of parameters for text session. It is recommended to have at least a node, for example, ID, TAG, or implementation-specific ones, for the identification purpose such that each set of parameters can be distinguished and accessed.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Text/`<X>`/Priority

This leaf represents the priority of a set of parameters for text session. Lower value means higher priority and the value is used in the terminal for client initiated QoS handling. The priority uses a 16 bit unsigned integer.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get
- Values: Zero or higher

/`<X>`/Text/`<X>`/IPver

This leaf represents the version of the Internet Protocol used in the session.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get
- Values: "IPv4", "IPv6"

/`<X>`/Text/`<X>`/TextFormat

This leaf node represents the MIME subtype name of text conversation protocol. The value "t140" refers to T.140 defined in ITU-T [26], [27].

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: MIME subtype name of the text conversation protocol, e.g., "t140"

/`<X>`/Text/`<X>`/Bandwidth

This interior node is used to allow a reference to a list of parameters related to text bandwidth assignment.

- Occurrence: One
- Format: node
- Minimum Access Types: Get

/`<X>`/Text/`<X>`/Bandwidth/AS

This leaf provides the value for "b=AS" line for text part used in the end-to-end SDP negotiation process, which represents the bit rate in kbits/sec.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/Bandwidth/RS

This leaf provides the value for "b=RS" line for text part used in the end-to-end SDP negotiation process, which represents the bit rate in bits/sec.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/Bandwidth/RR

This leaf provides the value for "b=RR" line for text part used in the end-to-end SDP negotiation process, which represents the bit rate in bits/sec.

- Occurrence: ZeroOrOne
- Format: int

- Minimum Access Types: Get

/ $\langle X \rangle$ /Text/ $\langle X \rangle$ /RedundancyLevel

This leaf node represents the level of redundancy when redundancy is used with T.140 text.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get
- Values: 0, 100, 200, 300

/ $\langle X \rangle$ /Text/ $\langle X \rangle$ /SamplingTime

This leaf node, defined in clause 9.4, represents the period for which text may be buffered before transmission. Buffering time, defined in [31], has an identical meaning as this node, i.e., the shortest period between text transmissions. Default value is 300 ms.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/ $\langle X \rangle$ /Text/ $\langle X \rangle$ /ConRef

This node specifies a reference to QoS parameters Management Object. The interior node's leaf nodes specify the network preferred QoS parameters as defined in TS 24.008 and they should be used in the bearer request when client initiated QoS happen. Implementation specific MO may be referenced.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/ $\langle X \rangle$ /Text/ $\langle X \rangle$ /Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/ $\langle X \rangle$ /Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

15.3 Example Configuration of 3GPP MTSINP MO

The examples below are configurations of 3GPP MTSINP MO for selected speech, text, and video sessions in Annex A. An example of SDP offer for speech session is shown in Table A.6.1, which includes two RTP payload types for AMR-NB. Parameter values in Table 15.1 may apply to both payload types and additional SDP parameters such as max-red may be included under the Ext node as vendor extensions. Depending on the implementation, two sets of session parameters may be defined for the two payload types respectively.

Table 15.1: Example configuration of MTSINP for speech session

| | | | |
|---------------|-----------|-----------|------|
| Speech | ID | 4 | |
| | TAG | Undefined | |
| | Priority | 2 | |
| | IPver | IPv4 | |
| | Codec | "AMR" | |
| | Bandwidth | AS | 30 |
| | | RS | 0 |
| | | RR | 2000 |
| | RateSet | Undefined | |
| ConRef | Undefined | | |

An example configuration of MTSINP for video session is shown in Table 15.3, which includes the RTP payload types for H.264. Although the "b=AS" value can also be computed with the Source and PayloadSize nodes, a different value with appropriate implementation margin can be directly assigned to the AS node. If the AS, Source, and PayloadSize nodes are defined together, the AS node value should be used for setting "b=AS". In Table 15.3, the "b=AS" values of 315, for H.264, are computed assuming IPv4 addressing. Note that the Priority node of H.264 is assigned values of 5, which shows that depending on service policy, parameters sets of lower priority may be preferred in the construction of SDP offer. If the ImageAttr node is to be defined, the maximum image size in either the Send or Recv node shall not exceed the maximum size limited by the offered codec level, which is 352x288 for Baseline profile at level 1.1.

Table 15.2: Example configuration of MTSINP for text session

| | | | |
|-------------|-----------------|-----------|-----|
| Text | ID | 3 | |
| | TAG | Undefined | |
| | Priority | 1 | |
| | IPver | IPv4 | |
| | TextFormat | "t140" | |
| | Bandwidth | AS | 2 |
| | | RS | 0 |
| | | RR | 500 |
| | RedundancyLevel | 200 | |
| ConRef | Undefined | | |
| Text | ID | 4 | |
| | TAG | Undefined | |
| | Priority | 2 | |
| | IPver | IPv4 | |
| | TextFormat | "t140" | |
| | Bandwidth | AS | 2 |
| | | RS | 0 |
| | | RR | 500 |
| | RedundancyLevel | 0 | |
| ConRef | Undefined | | |

An example of SDP offer for video session is shown in Table A.4.4b, which includes a RTP payload type for H.264. Although the "b=AS" value can also be computed with the Source and PayloadSize nodes, a different value with

appropriate implementation margin can be directly assigned to the AS node. If the AS, Source, and PayloadSize nodes are defined together, the AS node value should be used for setting "b=AS". In Table 15.3, the "b=AS" values of 315 and 57 kbps, for H.264 and H.263 respectively, are computed assuming IPv4 addressing. Note that the Priority nodes of H.264 and H.263 are assigned values of 5 and 3 respectively, which shows that depending on service policy, parameters sets of lower priority may be preferred in the construction of SDP offer. If the ImageAttr node is to be defined, as for H.264 in Table A.4.10a, the maximum image size in either the Send or Recv node shall not exceed the maximum size limited by the offered codec level, which is 352x288 for Baseline profile at level 1.1.

Table 15.3: Example configuration of MTSINP for video session

| | | | | |
|-----------|--------------|--|-------------|--|
| Video | ID | | 4 | |
| | TAG | | Undefined | |
| | Priority | | 5 | |
| | IPver | | IPv4 | |
| | Codec | | "H264" | |
| | Bandwidth | AS | | 315 |
| | | RS | | 0 |
| | | RR | | 2500 |
| | | Source | | 300 |
| | PayloadSize | | 1250 | |
| | ProfileLevel | H263 | Profile | Undefined |
| | | | Level | Undefined |
| | | MPEG4 | | Undefined |
| | ImageAttr | Send | | "176, 144, 224, 176, 272, 224, 320, 240" |
| Receive | | "176, 144, 0.5, 224, 176, 0.5, 272, 224, 0.6, 320, 240, 0.5" | | |
| ConRef | | Undefined | | |
| Video | ID | | 1 | |
| | TAG | | Undefined | |
| | Priority | | 3 | |
| | IPver | | IPv4 | |
| | Codec | | "H263-2000" | |
| | Bandwidth | AS | | 57 |
| | | RS | | 0 |
| | | RR | | 2500 |
| | | Source | | 48 |
| | PayloadSize | | 250 | |
| | ProfileLevel | H263 | Profile | 0 |
| | | | Level | 10 |
| | | MPEG4 | | Undefined |
| | H264 | | Undefined | |
| ImageAttr | Send | | Undefined | |
| | Receive | | Undefined | |
| ConRef | | Undefined | | |

16 Quality of Experience

16.1 General

The MTSI Quality of Experience (QoE) metrics feature is optional for an MTSI client in a terminal and shall not disturb the MTSI service. Non-terminal MTSI clients (such as gateways) should not implement MTSI QoE reporting. An MTSI

client that supports the QoE metrics feature shall support OMA-DM. The OMA-DM configuration server can configure the activation/deactivation and gathering of QoE metrics in the MTSI client (see clause 16.3). Configuration can also be done using the QMC functionality (see clause 16.5). An MTSI client supporting the QoE metrics feature shall perform the quality measurements in accordance to the measurement definitions, aggregate them into client QoE metrics and report the metrics. The MTSI client may send QoE metrics reports during the session and at the end of the session. The way how the QoE metrics are processed and made available is out of the scope of this specification.

16.2 Metrics Definition

An MTSI client supporting the QoE metrics feature shall support the reporting of the metrics in this clause. The metrics are valid for speech, video and text media, and are calculated for each measurement resolution interval "Measure-Resolution" (sub-clause 16.3.2). They are reported to the server according to the measurement reporting interval "Sending-Rate" (sub-clause 16.3.2) and after the end of the session (sub-clause 16.4).

16.2.1 Corruption duration metric

Corruption duration, M , is the time period from the NPT time of the last good frame (since the NPT time for the first corrupted frame cannot always be determined) before the corruption, to the NPT time of the first subsequent good frame. A corrupted frame may either be an entirely lost frame, or a media frame that has quality degradation and the decoded frame is not the same as in error-free decoding.

A good frame is a completely received frame:

- where all parts of the image are guaranteed to contain the correct content; or
- that is a refresh frame, that is, does not reference any previously decoded frames; or
- which only references previously decoded good frames

Completely received means that all the bits are received and no bit error has occurred.

Corruption duration, M , in milliseconds can be calculated as below:

- a) M can be derived by the client using the codec layer, in which case the codec layer signals the decoding of a good frame to the client. A good frame could also be derived by error tracking methods, but decoding quality evaluation methods shall not be used.
- b) Alternatively, the corruption is considered as ended after N milliseconds with consecutively completely received frames, or when a refresh frame has been completely received, whichever comes first..

The optional configuration parameter N can be set to define the average characteristics of the codec. If N has not been configured it shall default to the length of one measurement interval for video media, and to one frame duration for non-video media.

The syntax for the metrics "Corruption_Duration" is as defined in sub-clause 16.3.2.

The N parameter is specified in milliseconds and is used with the "Corruption_Duration" parameter in the "3GPP-QoE-Metrics" definition. The value of N may be set by the server. The syntax for N to be included in the "att-measure-spec" (sub-clause 16.3.2) is as follows:

- $N = "N" "=" 1 * \text{DIGIT}$

All the occurred corruption durations within each resolution period are summed and stored in the vector *TotalCorruptionDuration*. The unit of this metrics is expressed in milliseconds. Within each resolution period the number of individual corruption events are summed up and stored in the vector *NumberOfCorruptionEvents*. These two vectors are reported by the MTSI client as part of the reception report (sub-clause 16.4).

The parameter *CorruptionAlternative* indicates how the metric has been calculated, and shall be sent by the client via reception reporting (sub-clause 16.3.2) as "a", or "b".

16.2.2 Successive loss of RTP packets

The metric "Successive_Loss" indicates the number of RTP packets lost in succession per media channel.

The syntax for the metrics "Successive_Loss" is as defined in sub-clause 16.3.2.

All the number of successively lost RTP packets are summed up within each measurement resolution period of the stream and stored in the vector *TotalNumberOfSuccessivePacketLoss*. The unit of this metric is expressed as an integer equal to or larger than 0. The number of individual successive packet loss events within each measurement resolution period are summed up and stored in the vector *NumberOfSuccessiveLossEvents*. The number of received packets are also summed up within each measurement resolution period and stored in the vector *NumberOfReceivedPackets*. These three vectors are reported by the MTSI client as part of the QoE report (sub-clause 16.4)

16.2.3 Frame rate

Frame rate indicates the playback frame rate. The playback frame rate is equal to the number of frames displayed during the measurement resolution period divided by the time duration, in seconds, of the measurement resolution period.

The syntax for the metric "Frame_Rate" is defined in sub-clause 16.3.2.

For the Metrics-Name "Frame_Rate", the value field indicates the frame rate value. This metric is expressed in frames per second, and can be a fractional value. The frame rates for each resolution period are stored in the vector *FrameRate* and reported by the MTSI client as part of the QoE report (sub-clause 16.4).

16.2.4 Jitter duration

Jitter happens when the absolute difference between the actual playback time and the expected playback time is larger than *JitterThreshold* milliseconds. The expected time of a frame is equal to the actual playback time of the last played frame plus the difference between the NPT time of the frame and the NPT time of the last played frame.

The syntax for the metric "Jitter_Duration" is defined in sub-clause 16.3.2.

The optional configuration parameter JT can be set to control the amount of allowed jitter. If the parameter has not been set, it defaults to 100 ms. The JT parameter is specified in ms and is used with the "Jitter_Duration" parameter in the "3GPP-QoE-Metrics" definition. The value of JT may be set by the server. The syntax for JT to be included in the "att-measure-spec" (sub-clause 16.3.2) is as follows:

- JT = "JT" "=" 1 *DIGIT

All the jitter durations are summed up within each measurement resolution period and stored in the vector *TotalJitterDuration*. The unit of this metric is expressed in seconds, and can be a fractional value. The number of individual events within the measurement resolution period are summed up and stored in the vector *NumberOfJitterEvents*. These two vectors are reported by the MTSI client as part of the QoE report (sub-clause 16.4).

Note that the jitter duration metric is not measuring the incoming RTP or frame jitter, but instead the actual visible media playback jitter. Thus with a large enough jitter buffer the incoming RTP or frame jitter might be substantial, without any measurable jitter duration being reported (even if the *JitterThreshold* would have been set to zero).

16.2.5 Sync loss duration

Sync loss happens when the absolute difference between value A and value B is larger than *SyncThreshold* milliseconds. Value A represents the difference between the playback time of the last played frame of the video stream and the playback time of the last played frame of the speech/audio stream. Value B represents the difference between the expected playback time of the last played frame of the video stream and the expected playback time of the last played frame of the speech/audio stream.

The syntax for the metric "SyncLoss_Duration" is defined in sub-clause 16.3.2.

The optional configuration parameter ST can be set to control the amount of allowed sync mismatch. If the parameter has not been set, it defaults to 100 ms. The ST parameter is specified in ms and is used with the "SyncLoss_Duration" parameter in the "3GPP-QoE-Metrics" definition. The value of ST may be set by the server. The syntax for ST to be included in the "att-measure-spec" (sub-clause 16.3.2) is as follows:

- ST = "ST" "=" 1 *DIGIT

All the sync loss durations are summed up within each measurement resolution period and stored in the vector *TotalSyncLossDuration*. The unit of this metric is expressed in seconds, and can be a fractional value. The number of

individual events within the measurement resolution period are summed up and stored in the vector *NumberOfSyncLossEvents*. These two vectors are reported by the MTSI client as part of the QoE report (sub-clause 16.4).

16.2.6 Round-trip time

The round-trip time (RTT) consists of the RTP-level round-trip time, plus the additional two-way delay (RTP level->loudspeaker->microphone->RTP level) due to buffering and other processing in each client.

The syntax for the metric "Round_Trip_Time" is defined in sub-clause 16.3.2.

The last RTCP round-trip time value estimated during each measurement resolution period shall be stored in the vector *NetworkRTT*. The unit of this metrics is expressed in milliseconds.

The two-way additional internal client delay valid at the end of each measurement resolution period shall be stored in the vector *InternalRTT*. The unit of this metrics is expressed in milliseconds.

The two vectors are reported by the MTSI client as part of the QoE report (sub-clause 16.4).

Note that if the RTP and the RTCP packets for a media are not sent in the same RTP stream the estimated media round-trip time might be unreliable.

16.2.7 Average codec bitrate

The average codec bitrate is the bitrate used for coding "active" media information during the measurement resolution period.

For speech media "active" information is defined by frames containing speech, i.e. silence frames (SID-frames) and DTX periods are excluded from the calculation. If application-layer redundancy is used, any redundant frames should also be excluded. If partial redundancy is used within frames (e.g. EVS channel-aware mode), only the non-redundant bits in these frame should be counted as having "active" information.

The exact method for how the client identifies the active frames or bits is not specified here, and it is recognized that an implementation might not be able to fully exclude all non-active frames or bits from the calculation.

Thus for speech media the average codec bitrate can be calculated as the number of "active" speech bits received for "active" frames, divided by the total time, in seconds, covered by these frames. The total time covered is calculated as the number of "active" frames times the length of each speech frame.

For non-speech media the average codec bitrate is the total number of RTP payload bits received, divided by the length of the measurement resolution period.

The syntax for the metric "Average_Codec_Bitrate" is defined in sub-clause 16.3.2.

The average codec bitrate value for each measurement resolution period shall be stored in the vector *AverageCodecBitrate*. The unit of this metrics is expressed in kbit/s and can be a fractional value. The vector is reported by the MTSI client as part of the QoE report (sub-clause 16.4).

16.2.8 Codec information

The codec information metrics contain details of the media codec settings used in the receiving direction during the measurement resolution period. If the codec information is changed during the measurement resolution period, the codec information valid when each measurement resolution period ends shall be reported. The unit of this metric is a string value. No "white space" characters are allowed in the string values, and shall be removed if necessary.

For speech media the semi-colon-separated codec information contains the used speech codec type (represented as in an SDP rtpmap offer) and the used codec settings for bandwidth and redundancy (represented as in an SDP fmpo offer). For instance "EVS/16000/1;bw=swb;", or "EVS/16000/1;ch-aw-recv=3".

For video media, the codec information contains the video codec type, represented as in an SDP offer, for instance "H265/90000". Furthermore, the semi-colon-separated video profile, level (and tier if applicable) used shall be reported, represented as in an SDP offer. For instance for H.265, "profile-id=1;level-id=120;tier-flag=1", or for H.264, "profile-

level-id=42e00a". The actually used image size (not the maximum allowed) shall also be reported, represented as "WxH", for example "320x240".

For real-time text media, the codec information contains the text encoding, represented as in an SDP offer, for instance "t140/1000/1".

The syntax for the metric "Codec_Info", "Codec_ProfileLevel" and "Codec_ImageSize" are defined in sub-clause 16.3.2.

The codec info, profile/level/tier and codec image size value for each measurement resolution period shall be stored in the vectors *CodecInfo*, *CodecProfileLevel* and *CodecImageSize* respectively. If the metric values in these vectors for a measurement resolution period are unchanged from the previous values in the respective vector, it is allowed to put the value "=" in the vector to indicate this. The *CodecInfo*, *CodecProfileLevel* and *CodecImageSize* vectors are reported by the MTSI client as part of the QoE report (sub-clause 16.4).

16.2.9 Call setup time

The call setup time is measured on SIP level for originating calls (see ITU-T G-1028 [181] Table 3). It is defined as the time between the transmitted INVITE, and the reception of either "200 OK" or "180 RINGING" (whichever comes first).

The syntax for the metric "Call_Setup_Time" is defined in sub-clause 16.3.2.

The measured call setup time shall be stored in the variable *CallSetupTime*. The unit of this metrics is expressed in milliseconds. The variable is reported by the MTSI client as part of the QoE report (sub-clause 16.4).

NOTE: The reason for also using "180 RINGING" as an end-of-call-setup criteria is to compensate for the physical answering delay caused by the called user. Thus the metric measures the fastest possible call setup time, i.e. as if the called user had answered directly.

16.3 Metric Configuration

An MTSI client supporting the QoE metrics feature shall support the OMA-DM solution specified in this clause for configuration of QoE metrics and their activation. The MTSI client shall also support the QMC functionality specified in clause 16.5 for configuration of QoE metrics.

The QoE configuration shall only be evaluated by the client at the start of a QoE measurement and reporting session ("QoE session") associated with a MTSI session. This includes evaluation of any filtering criteria such as by geographical area. Client evaluation of all measurement and reporting criterias for an ongoing QoE session shall be unaffected by any QoE configuration changes received during that session – i.e., any changes to the QoE configuration shall only affect QoE sessions started after these configuration changes have been received.

If an MTSI client uses the OMA-DM configuration feature, it is mandatory for the MTSI client to implement the Management Object (MO) as described in this clause.

The 3GPP MTSIQOE (MTSI QoE metrics) MO defined in this clause may be used to configure the QoE metrics and reporting settings.

The metrics specified in the MO may be derived by the MTSI client. Version numbering is included for possible extension of the MO.

The Management Object Identifier shall be: urn:oma:mo:ext-3gpp-mtsique:1.0.

Protocol compatibility: The MO is compatible with OMA Device Management protocol specifications, version 1.2 and upwards, and is defined using the OMA DM Device Description Framework as described in the Enabler Release Definition OMA-ERELD_DM-V1_2 [67].

16.3.1 QoE metrics reporting management object

The following nodes and leaf objects in figure 16.1 shall be contained under the 3GPP_MTSIQOE node if an MTSI client supports the feature described in this clause (information of DDF for this MO is given in Annex I):

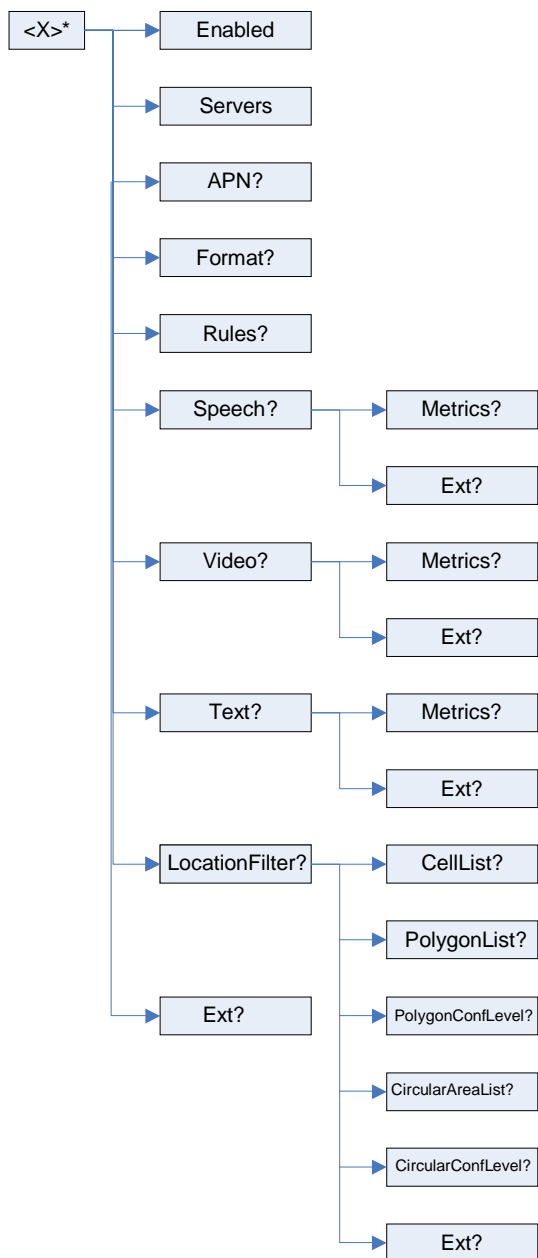


Figure 16.1: MTSI QoE metrics management object tree

Node: /<X>

This interior node specifies the unique object id of a MTSI QoE metrics management object. The purpose of this interior node is to group together the parameters of a single object.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

The following interior nodes shall be contained if the MTSI client supports the "MTSI QoE metrics Management Object".

/<X>/Enabled

This leaf indicates if QoE reporting is requested by the provider.

- Occurrence: One
- Format: bool
- Minimum Access Types: Get

/<X>/Servers

This leaf contains a space-separated list of URL of servers to which the QoE reports can be transmitted. It is URI addresses, e.g. <http://qoeserver.operator.com>. In case of multiple servers, the MTSI client randomly selects one of the servers from the list, with uniform distribution.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get
- Values: URI of the servers to receive the QoE report.

/<X>/APN

This leaf contains the Access Point Name that should be used for establishing the PDP context or EPS bearer on which the QoE metric reports will be transmitted. This may be used to ensure that no costs are charged for QoE metrics reporting.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: the Access Point Name

/<X>/Format

This leaf specifies the format of the report and if compression (Gzip XML) [71] is used.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: "XML", "GZIPXML".

/<X>/Rules

This leaf provides in textual format the rules used to decide whether metrics are to be reported to the QoE metrics report server. The syntax and semantics of this leaf are defined in clause 16.3.3.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: See clause 16.3.3.

/<X>/Ext

The Ext node is an interior node where the vendor specific information can be placed (vendor includes application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized subtrees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Speech

The Speech node is the starting point of the speech media level QoE metrics definitions.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Speech/Metrics

This leaf provides in textual format the QoE metrics that need to be reported, the measurement frequency, the reporting interval and the reporting range. The syntax and semantics of this leaf are defined in clause 16.3.2.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: see clause 16.3.2.

/`<X>`/Speech/Ext

The Ext node is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized subtrees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video

The Video node is the starting point of the video media level QoE metrics definitions.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/Metrics

This leaf provides in textual format the QoE metrics that need to be reported, the measurement frequency, the reporting interval and the reporting range. The syntax and semantics of this leaf are defined in clause 16.3.2.

- Occurrence: ZeroOrOne
- Format: chr
- Access Types: Get
- Values: see clause 16.3.2.

/`<X>`/Video/Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the Ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Text

The Text node is the starting point of the real-time text media level QoE metrics definitions.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get
- Values: see clause 16.3.2.

/`<X>`/Text/Metrics

This leaf provides in textual format the QoE metrics that need to be reported, the measurement frequency, the reporting interval and the reporting range. The syntax and semantics of this leaf are defined in clause 16.3.2.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: see clause 16.3.2.

/`<X>`/Text/Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/`<LocationFilter>`

When present, this element indicates the geographic area(s) or location(s) where quality metric collection is requested. When not present, quality metric collection is requested regardless of the device's location. The LocationFilter element comprises one or more instances of any combination of targeted cell-IDs, polygons and circular areas. Each cell-ID entry in LocationFilter is announced in cellList, and each polygon and circular area entry is announced in the polygonList or and circularAreaList elements, respectively.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/`<LocationFilter>`/CellList

This element specifies a list of cell identified by the CGI (i.e., NCGI, ECGI, CGI).

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: a list of CGI.

`<X><LocationFilter>/PolygonList`

This leaf specifies a list of shapes defined as 'Polygon' by OMA MLP [159].

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: a list of 'Polygon' defined by OMA MLP [159].

`<X><LocationFilter>/Polygon_Conf_Level`

This leaf indicates the probability in percent that the MTSI client is located in the corresponding polygon area specified by leaf 'PolygonList'. It is defined as 'lev_conf' by OMA MLP [159]. If not present, it has default value of 60.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get
- Values: 'lev_conf' defined by OMA MLP [159].

`<X><LocationFilter>/CircularAreaList`

This leaf specifies a list of shapes defined as 'CircularArea' by OMA MLP [159].

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: a list of 'CircularArea' defined by OMA MLP [159].

`<X><LocationFilter>/Circular_Conf_Level`

This leaf indicates the probability in percent that the MTSI client is located in the corresponding circular area specified by leaf 'CircularAreaList'. It is defined as 'lev_conf' by OMA MLP [159]. If not present, it has default value of 60.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get
- Values: 'lev_conf' defined by OMA MLP [159].

`<X><LocationFilter>/Ext`

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

16.3.2 QoE metric reporting configuration

The syntax of the text contained in the Metrics leaf is similar to the "3GPP-QoE-Metrics" attribute syntax specified in TS 26.234 [60] and TS 26.346 [74]:

- QoE-Metrics = "3GPP-QoE-Metrics:" att-measure-spec *(", " att-measure-spec)) CRLF
- att-measure-spec = Metrics ";" Sending-rate [";" Measure-Range] [";" Measure-Resolution] *([";" Parameter-Ext])
- Metrics = "metrics" "=" "{"Metrics-Name *(",|" Metrics-Name) "}"
- Metrics-Name = 1*((0x21..0x2b) / (0x2d..0x3a) / (0x3c..0x7a) / 0x7e) ;VCHAR except ";", ",", "{", "}"
- Sending-Rate = "rate" "=" 1*DIGIT / "End"
- Measure-Resolution = "resolution" "=" 1*DIGIT ; in seconds
- Measure-Range = "range" ":" Ranges-Specifier
- Parameter-Ext = (1*DIGIT ["." 1*DIGIT]) / (1*((0x21..0x2b) / (0x2d..0x3a) / (0x3c..0x7a) / 0x7c / 0x7e))
- Ranges-Specifier = as defined in RFC 2326 [72].

This attribute is used to indicate which QoE metrics are supported, the reporting interval, the measurement interval and reporting range.

The "Metrics" field contains the list of names that describes the metrics/measurements that are required to be reported in a MTSI call, provided that the MTSI client supports these measurements and the reporting rule conditions are met (see clause 16.3.3). The names that are not included in the "Metrics" field shall not be reported during the session.

The "Sending-Rate" shall be set, and it expresses the maximum time period in seconds between two successive QoE reports. If the "Sending-Rate" value is 0, then the client shall decide the sending time of the reports depending on the events occurred in the client. Values ≥ 1 indicate a precise reporting interval. The shortest interval is one second and the longest interval is undefined. The reporting interval can be different for different media, but it is recommended to maintain a degree of synchronization in order to avoid extra traffic in the uplink direction. The value "End" indicates that only one report is sent at the end of the session.

The optional "Measure-Resolution" field, if used, shall define a time over which each metrics value is calculated. The "Measure-Resolution" field splits the session duration into a number of equally sized periods where each period is of the length specified by the "Measure-Resolution" field. The "Measure-Resolution" field is thus defining the time before the calculation of a QoE parameter starts over. If the "Measure-Resolution" field is not present, the metrics resolution shall cover the period specified by the "Measure-Range" field. If the "Measure-Range" field is not present the metrics resolution shall be the whole session duration.

The optional "Measure-Range" field, if used, shall define the time range in the stream for which the QoE metrics will be reported. There shall be only one range per measurement specification. The range format shall be any of the formats allowed by the media. If the "Measure-Range" field is not present, the metrics range shall be the whole call duration.

16.3.3 QoE reporting rule definition

This clause defines the syntax and semantics of a set of rules which are used to reduce the amount of reporting to the QoE metrics report server. The syntax of the metrics reporting rules is defined below:

- QoE-Rule = "3GPP-QoE-Rule" ":" rule-spec *(", " rule-spec)

- rule-spec = rule-name [";" parameters]
- rule-name = "OnlyCallerReports" / "LimitSessionInterval" / "SamplePercentage"
- parameters = parameter *(";" parameter)
- parameter = Param-Name ["=" Param-Value]
- Param-Name = 1*((0x21..0x2b) / (0x2d..0x3a) / (0x3c..0x7a) / 0x7e) ;VCHAR except ";", " ", "{", "}"
- Param-Value = (1*DIGIT ["." 1*DIGIT]) / (1*((0x21..0x2b) / (0x2d..0x3a) / (0x3c..0x7a) / 0x7c / 0x7e))

The semantics of the rules and the syntax of its parameters is defined below:

The *OnlyCallerReports* rule is used to determine the metrics reporting sources. When this rule is present, only the initiator of the call, i.e., caller, will report metrics to the QoE report server. When absent all parties report metrics.

The *SamplePercentage* rule can be used to set a percentage sample of calls which should report reception. This can be useful for statistical data analysis of large populations while increasing scalability due to reduced total uplink signalling. The *sample_percentage* parameter takes on a value between 0 and 100, including the use of decimals. It is recommended that no more than 3 digits follow a decimal point (e.g. 67.323 is sufficient precision).

When the *SamplePercentage* rule is not present or its *sample_percentage* parameter value is 100 each MTSI client shall send metric report(s). If the *sample_percentage* value is less than 100, the UE generates a random number which is uniformly distributed in the range of 0 to 100. The UE sends the reception report when the generated random number is of a lower value than the *sample_percentage* value.

The *LimitSessionInterval* rule is used to limit the time interval between consecutive calls that report metrics. The *min_interval* parameter for this rule indicates the minimum time distance between the start of two calls that are allowed to report metrics. When this rule is absent there is no limitation on the minimum time interval.

In case multiple rules are defined in the Management Object, the MTSI client should only report metrics when all individual rules evaluate to true (i.e. the rules are logically ANDed). In case no rules are present the MTSI client should always report metrics (see also clause 16.4 for metrics reporting procedures).

An example for a QoE metric reporting rule is shown below:

```
3GPP-QoE-Rule:OnlyCallerReports,SamplePercentage;sample_percentage=10.0,
LimitSessionInterval;min_interval=300,
```

This example rule defines that only the caller shall report, and only for 10% of the sessions, with the minimum time interval between the start times of two consecutive calls that report metrics to be 5 minutes.

16.4 Metrics Reporting

When a session is started, the MTSI client must determine whether QoE reporting is required for the session. If the parameter "Enabled" is set to false, no QoE reporting shall be done. If the "Enabled" parameter is set to true the optional "Rules" parameters are checked (sub-clause 16.3.3) to define if QoE reporting shall be done.

Once the need for QoE reporting has been established, the client shall continuously compute all specified metrics for each measurement interval period, according to the "Measure-Resolution" parameter (sub-clause 16.3.2). In order to bound the resources used by metrics reporting, the minimum values for the Measure-Resolution and Sending-Rate are specified to be 5 seconds and 30 seconds respectively. The computed metrics are represented in a vector format, adding an additional metric value to each metric vector after each new measurement interval period.

Note that the calculated metrics shall only cover one measurement interval. For instance, if the corruption duration extends longer than to the end of the current measurement interval, only the portion which fits into the current measurement interval shall be reported. The remaining portion of the corruption duration shall be reported as belonging to the next measurement interval.

The end of the session will normally not correspond to the end of a measurement interval period, so the metrics for the last measurement interval period will typically be calculated over a time shorter than the configured measurement interval. Note, however, that these last metrics shall still be added to the metrics vectors and reported to the server.

It is possible for the server to use the start and stop timestamps, together with the knowledge of the configured measurement interval, to derive the actual length of the last measurement interval period, but any specific action or interpretation of these last shorter measurements is out of scope of this specification.

The MTSI client shall send QoE report messages to the server in accordance with the specified reporting interval "Sending-Rate" (sub-clause 16.3.2). All stored metrics data shall then be sent to the server, and then deleted from the metrics storage.

Note that if the reporting interval is not an integer multiple of the measurement interval, only the measurement interval periods which have been fully passed shall be included in the report. The ongoing not-passed measurement interval period shall be included in the next report. The only exception is at the end of the session, where also the last ongoing measurement interval period shall be directly calculated and included in the report.

If QoE configuration has been done via the OMA MO, the client shall send QoE reports using the HTTP (RFC 2616 [73]) POST request carrying XML formatted metadata. If the optional "APN" parameter is defined in the OMA managed object, that APN shall be used for establishing the PDP context or EPS bearer on which the QoE metric reports will be transmitted. The MTSI client randomly selects one of the URIs from the MO "Server" parameter, with uniform distribution.

If QoE configuration has been done via the QMC functionality (see clause 16.5), the client shall also send the QoE reports as described in clause 16.5. Note that for QMC scheme, if the SliceScope is included in the QoE configuration and the slice associated with the MTSI service is within the SliceScope, the QoE collection shall be executed and the S-NSSAI and DNN that correspond to the report data shall be included for support of per-slice QoE reporting and evaluation in OAM. This information may be retrieved via the AT Command +CGDCONT [161]) or the specific traffic mapping with URSP rule [182].

Each QoE report is formatted in XML according the following XML schema (sub-clause 16.4.1). An informative example of a single reception report XML object is also given (sub-clause 16.4.2). The reports should be compressed using GZIP only if the MO parameter "Format" specifies this.

Each QoE Metrics element has a set of attributes and any number of media level QoE Metrics elements. All attributes are defined in sub-clause 16.4.1 and correspond to the QoE metrics listed in sub-clause 16.2. Individual metrics can be selected as described in sub-clause 16.3.2.

Except for the media level QoE metrics, the following parameters shall be reported for each report:

- The *callId* attribute identifies the call identity of the SIP session.
- The *clientId* attribute is unique identifier for the receiver, e.g. an MSISDN of the UE as defined in [80].
- The *startTime* and *stopTime* attributes identifies the client NTP time when the measurements included in the report were started and stopped. The time is based on the local real-time clock in the client, and might not be consistent with the true NTP time. However, assuming that the reporting is done without any extra delay the server can use the *stopTime* attribute to correct the timestamps if necessary.
- The *mediaId* attribute shall be reported for each media level QoE report, and identifies the port number for the media.

If the attribute *qoeReferenceId* was defined in the QMC configuration (see clause 16.5.2), the value shall be copied into each QoE report, to facilitate network-side correlation (see [178]). If this attribute was defined, the attribute *recordingSessionId* shall also be returned for each QoE report. The *recordingSessionId* is a two-byte octet defined by the client. It shall remain the same for all QoE reports belonging to the same session, and it should be different for QoE reports belonging to different sessions.

16.4.1 XML schema for QoE report message

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:3gpp:metadata:2008:MTSI:qoereport"
xmlns="urn:3gpp:metadata:2008:MTSI:qoereport"
  elementFormDefault="qualified">
  <xs:element name="QoeReport" type="QoeReportType"/>

  <xs:complexType name="QoeReportType">
```

```

<xs:sequence>
  <xs:element name="statisticalReport" type="starType" minOccurs="0"
    maxOccurs="unbounded"/>
  <xs:any namespace="##other" processContents="skip" minOccurs="0"
    maxOccurs="unbounded"/>
</xs:sequence>
<xs:anyAttribute processContents="skip"/>
</xs:complexType>

<xs:complexType name="starType">
<xs:sequence>
  <xs:element name="mediaLevelQoeMetrics" type="mediaLevelQoeMetricsType" minOccurs="1"
    maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="startTime" type="xs:unsignedLong" use="required"/>
<xs:attribute name="stopTime" type="xs:unsignedLong" use="required"/>
<xs:attribute name="callId" type="xs:string" use="required"/>
<xs:attribute name="clientId" type="xs:string" use="required"/>
<xs:attribute name="qoeReferenceId" type="xs:hexBinary" use="optional"/>
<xs:attribute name="recordingSessionId" type="xs:hexBinary" use="optional"/>
<xs:attribute name="dnn" type="xs:string" use="optional"/>
<xs:attribute name="snssai" type="xs:unsignedLong" use="optional"/>
<xs:anyAttribute processContents="skip"/>
</xs:complexType>

<xs:complexType name="mediaLevelQoeMetricsType">
<xs:sequence>
  <xs:any namespace="##other" processContents="skip" minOccurs="0"
    maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="mediaId" type="xs:integer" use="required"/>
<xs:attribute name="totalCorruptionDuration" type="unsignedLongVectorType"
  use="optional"/>
<xs:attribute name="numberOfCorruptionEvents" type="unsignedLongVectorType"
  use="optional"/>
<xs:attribute name="corruptionAlternative" type="xs:string" use="optional"/>
<xs:attribute name="totalNumberOfSuccessivePacketLoss" type="unsignedLongVectorType"
  use="optional"/>
<xs:attribute name="numberOfSuccessiveLossEvents" type="unsignedLongVectorType"
  use="optional"/>
<xs:attribute name="numberOfReceivedPackets" type="unsignedLongVectorType"
  use="optional"/>
<xs:attribute name="framerate" type="doubleVectorType" use="optional"/>
<xs:attribute name="totalJitterDuration" type="doubleVectorType" use="optional"/>
<xs:attribute name="numberOfJitterEvents" type="unsignedLongVectorType"
  use="optional"/>
<xs:attribute name="totalSyncLossDuration" type="doubleVectorType" use="optional"/>
<xs:attribute name="numberOfSyncLossEvents" type="unsignedLongVectorType"
  use="optional"/>
<xs:attribute name="networkRTT" type="unsignedLongVectorType" use="optional"/>
<xs:attribute name="internalRTT" type="unsignedLongVectorType" use="optional"/>
<xs:attribute name="codecInfo" type="stringVectorType" use="optional"/>
<xs:attribute name="codecProfileLevel" type="stringVectorType" use="optional"/>
<xs:attribute name="codecImageSize" type="stringVectorType" use="optional"/>
<xs:attribute name="averageCodecBitrate" type="doubleVectorType" use="optional"/>
<xs:attribute name="callSetupTime" type="xs:unsignedLong" use="optional"/>

<xs:anyAttribute processContents="skip"/>
</xs:complexType>

<xs:simpleType name="doubleVectorType">
<xs:list itemType="xs:double"/>
</xs:simpleType>

<xs:simpleType name="stringVectorType">
<xs:list itemType="xs:string"/>
</xs:simpleType>

<xs:simpleType name="unsignedLongVectorType">
<xs:list itemType="xs:unsignedLong"/>
</xs:simpleType>
</xs:schema>

```

16.4.2 Example XML for QoE report message

Below is one example of QoE report message, in this example the measurement interval is 20 seconds, the reporting interval is 5 minutes, but the call ends after 55 seconds.

```
<?xml version="1.0" encoding="UTF-8"?>
<QoeReport xmlns="urn:3gpp:metadata:2008:MTSI:qoereport"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:3gpp:metadata:2008:MTSI:qoereport qoereport.xsd">
  <statisticalReport
    startTime="1219322514"
    stopTime="1219322569"
    clientId="clientID"
    callId="callID">
    qoeReferenceId="240F512A"
    snssai="01000FFF"
    dnn="internet.mnc015.mcc234.gprs"
    recordingSessionId="0001"
    <mediaLevelQoeMetrics
      mediaId="1234"
      totalCorruptionDuration="480 0 120"
      numberOfCorruptionEvents="5 0 2"
      corruptionAlternative="a"
      totalNumberOfSuccessivePacketLoss="24 0 6"
      numberOfSuccessiveLossEvents="5 0 2"
      numberOfReceivedPackets="535 645 300"
      framerate="50.0 49.2 50.0"
      numberOfJitterEvents="0 1 0"
      totalJitterDuration="0 0.346 0"
      networkRTT="120 132 125"
      internalRTT="20 24 20"
      codecInfo="AMR-WB/16000/1 = ="
      averageCodecBitRate="12.4 12.65 12.7"/>
    <mediaLevelQoeMetrics
      mediaId="1236"
      totalCorruptionDuration="83 0 0"
      numberOfCorruptionEvents="1 0 0"
      corruptionAlternative="b"
      totalNumberOfSuccessivePacketLoss="3 0 0"
      numberOfSuccessiveLossEvents="2 0 0"
      numberOfReceivedPackets="297 300 225"
      framerate="14.7 15.0 14.9"
      numberOfJitterEvents="0 0 0"
      totalJitterDuration="0 0 0"
      numberOfSyncLossEvents="0 1 0"
      totalSyncLossDuration="0 0.789 0"
      networkRTT="220 232 215"
      internalRTT="27 20 25"
      codecInfo="H263-2000/90000 = ="
      codecProfileLevel="profile=0;level=45 = ="
      codecImageSize="176x144 = ="
      averageCodecBitRate="124.5 128.0 115.1"/>
    </statisticalReport>
  </QoeReport>
```

16.5 QoE Measurement Collection Functionalities

16.5.1 Configuration and reporting

As an alternative to configuration via OMA-DM, the QoE configuration can optionally be specified by the QoE Measurement Collection (QMC) functionality. In this case the QoE configuration is received via specific RRC [158] messages for UMTS, RRC [160] messages for LTE, and RRC [163] messages for NR over the control plane, and the QoE reporting is also sent back via RRC messages over the control plane.

If QMC is supported, the UE shall support the following QMC functionalities:

- QoE Configuration: The QoE configuration will be delivered via RRC to the UE as a container according to "Application Layer Measurement Configuration" (see [158]) for UMTS, "measConfigAppLayer" (see [160]) for

LTE and “AppLayerMeasConfig” for NR (see [163]). The container is an octet string with gzip-encoded data (see [71]) stored in network byte order. The maximum size of the container is 1000 bytes for UMTS (see [158]) and LTE (see [160]), and 8000 bytes for NR (see [163]). When the container is uncompressed it is expected to conform to XML-formatted QoE configuration data according to clause 16.5.2 in the current specification. This uncompressed QoE Configuration shall be delivered to the MTSI client. The interface towards the RRC signalling is handled by the AT command +CAPPLEVMC for UMTS and LTE, and the AT command +CAPPLEVMCNR for NR [161].

- QoE Metrics: QoE Metrics from the MTSI client shall be XML-formatted according to clause 16.4 in the current specification. The XML data shall be compressed with gzip (see [71]) and stored in network byte order into an octet string container. The maximum size is 8000 bytes for UMTS (see [158]) and LTE (see [160]). For NR (see [163]), the maximum size is 8000 bytes if RRC segmentation is not enabled, and 144000 bytes if enabled. The container shall be delivered via RRC to the RNC according to "Application Layer Measurement Reporting" (see [158]) for UMTS, to the eNB according to "measReportAppLayer" (see [160]) for LTE, and to the gNB according to “MeasurementReportAppLayer” for NR (see [163]). The behaviour if the compressed data is larger than the maximum container size is unspecified in this version of the specification. The interface towards the RRC signalling is handled by the AT command +CAPPLEVMR for UMTS and LTE, and the AT command +CAPPLEVMRNR for NR [161].
- The UE shall also set the QMC capability "QoE Measurement Collection for MTSI services" (see [158]) to TRUE for UMTS, include the QMC capability "qoe-mtsi-MeasReport" (see [160]) for LTE, and include the QMC capability “qoe-MTSI-MeasReport” (see [163]) for NR.
- When a new session is started, the QoE reporting AT command +CAPPLEVMRNR [161] shall be used to send a Recording Session Indication. Such an indication does not contain any QoE report, but indicates that QoE recording has started for a session.
- When the QoE configuration is to be released, an unsolicited result code associated with the AT command +CAPPLEVMC or AT command +CAPPLEVMCNR [161] and containing the parameter <start-stop_reporting> or <start-stop_measurement> set to "1", shall be sent to the MTSI client as notification of a discard request. Then the MTSI client shall stop collecting quality metrics and discard any already collected information [178].

The exact implementation is not specified here, but example signalling diagrams for UMTS, LTE and NR below show the QMC functionality with a hypothetical "QMC Handler" entity.

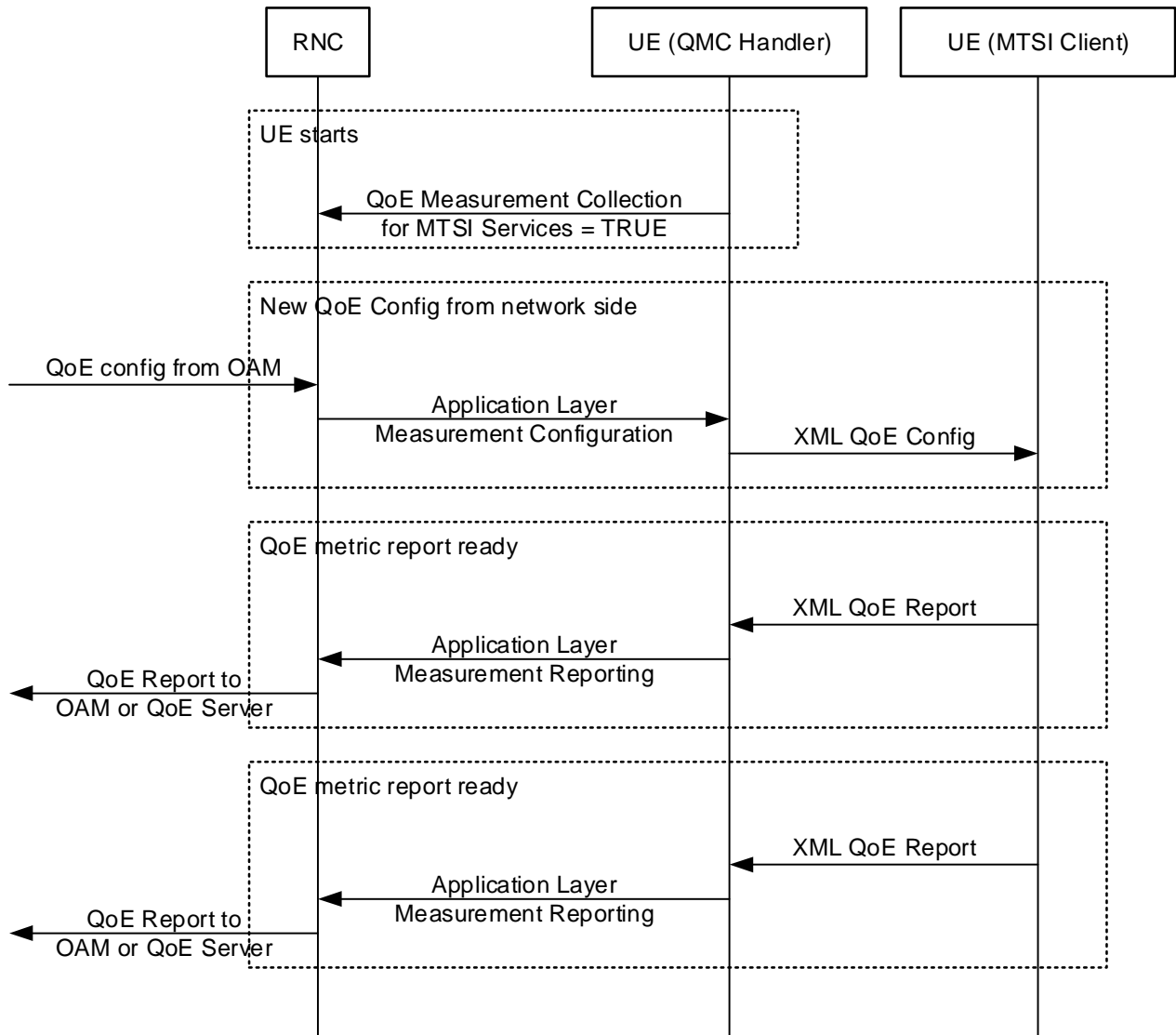


Figure 16.5.1-1: Example signalling diagram for UMTS

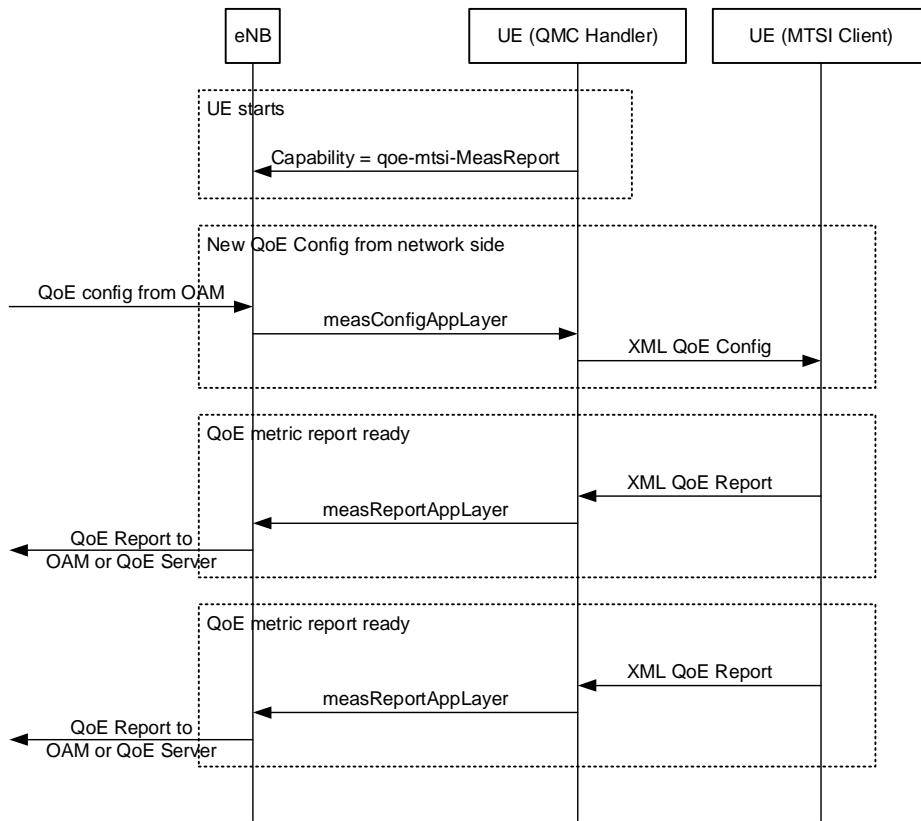


Figure 16.5.1-2: Example signalling diagram for LTE

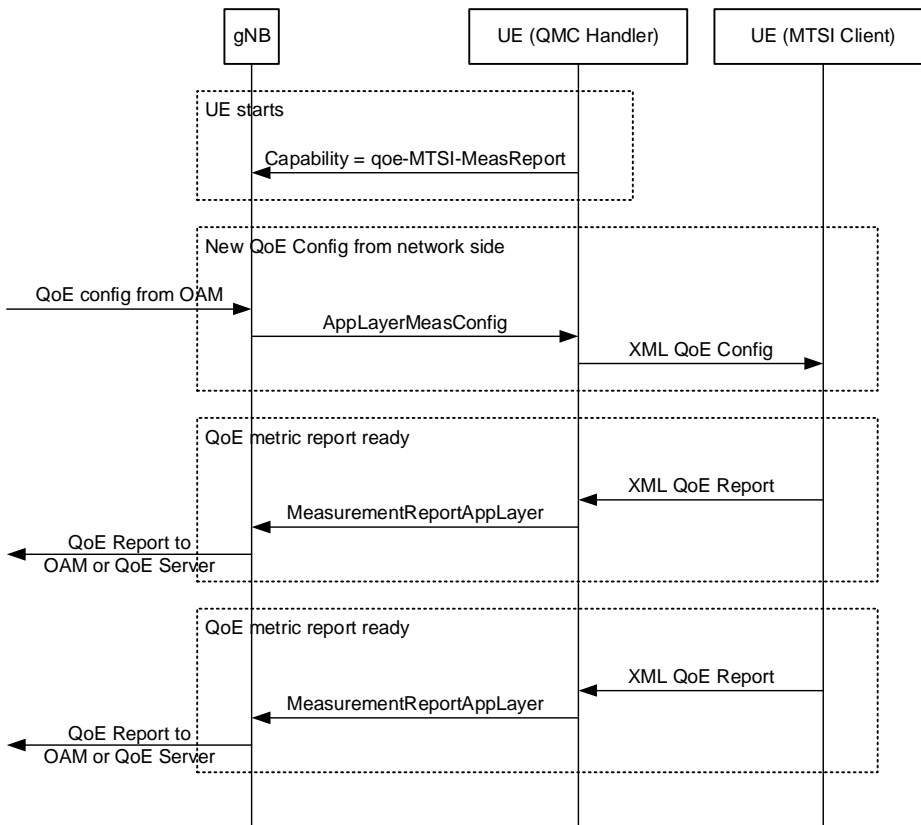


Figure 16.5.1-3: Example signalling diagram for NR

NOTE: The QMC Handler is only shown here as one possible implementation, and it need not be implemented as such. The corresponding QMC functionality could be built into the MTSI client or into other UE entities. In this version of the specification the detailed implementation of the above functionalities is left to the UE vendor.

16.5.2 XML configuration

When QoE reporting is configured via the QMC functionality, the configuration basically contains the same information as in the QoE metrics reporting managed object (see clause 16.3.1), but encapsulated according to the XML scheme below. Note that the managed object leaves "Servers", "APN" and "Format" are not needed for the QMC functionality, and thus not included.

Note that if geographical filtering is handled on the network side (i.e. QoE reporting is turned on/off by the network depending on the UE location), no LocationFilter should be specified in the QoE Configuration, as this would mean two consecutive filterings.

Also note that the optional attribute qoeReferenceId is a reference set by the network side (see [178]), which is not directly used by the client. However, if this attribute is defined, it shall be copied into each QoE report, to facilitate network-side correlation.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:3gpp:metadata:2017:MTSI:qoeconfig"
  elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:sv="urn:3gpp:metadata:2017:MTSI:schemaVersion"
  xmlns="urn:3gpp:metadata:2017:MTSI:qoeconfig">

  <xs:element name="MTSIQualityReporting" type="QualityReportingType"/>

  <xs:complexType name="QualityReportingType">
    <xs:sequence>
      <xs:element name="LocationFilter" type="LocationFilterType" minOccurs="0"/>
      <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="enabled" type="xs:boolean" use="required"/>
    <xs:attribute name="rules" type="xs:string" use="optional"/>
    <xs:attribute name="speechMetrics" type="xs:string" use="optional"/>
    <xs:attribute name="videoMetrics" type="xs:string" use="optional"/>
    <xs:attribute name="textMetrics" type="xs:string" use="optional"/>
    <xs:attribute name="qoeReferenceId" type="xs:hexBinary" use="optional"/>
    <xs:attribute name="sliceScope" type="UnsignedIntVectorType" use="optional"/>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
  </xs:complexType>

  <xs:complexType name="LocationFilterType">
    <xs:sequence>
      <xs:element name="cellID" type="xs:unsignedLong" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="shape" type="ShapeType" minOccurs="0"/>
      <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
  </xs:complexType>

  <xs:complexType name="ShapeType">
    <xs:sequence>
      <xs:element name="PolygonList" type="PolygonListType" minOccurs="0"/>
      <xs:element name="CircularAreaList" type="CircularAreaListType" minOccurs="0"/>
      <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
  </xs:complexType>

  <xs:complexType name="PolygonListType">
    <xs:annotation>
      <xs:documentation> see [OMA MLP] </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="Polygon" minOccurs="0" maxOccurs="unbounded"/>
      <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="ConfLevel" type="xs:unsignedInt" use="optional"/>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
  </xs:complexType>
```



```

<xs:complexType name="CircularAreaListType">
  <xs:annotation>
    <xs:documentation> see [OMA MLP] </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="CircularArea" minOccurs="0" maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="ConfLevel" type="xs:unsignedInt" use="optional"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>
<xs:simpleType name="UnsignedIntVectorType">
  <xs:list itemType="xs:unsignedInt"/>
</xs:simpleType>
</xs:schema>

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:3gpp:metadata:2017:MTSI:schemaVersion"
  xmlns="urn:3gpp:metadata:2017:MTSI:schemaVersion"

  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <xs:element name="schemaVersion" type="xs:unsignedInt"/>
  <xs:element name="delimiter" type="xs:byte"/>
</xs:schema>

```

17 Management of Media Adaptation

17.1 General

For the purpose of quality control or network management, it can be necessary to adjust the speech and video adaptation of the MTSI client in terminal. To effectively manage, i.e., initialize and update, the media adaptation of a large number of terminals, which can be implemented in different fashions, the 3GPP MTSIMA (MTSI Media Adaptation) MO defined in this clause may be used.

The MO, which exploits the information estimated or received from various entities such as ongoing multimedia packet stream, the far-end MTSI client in terminal, IMS, and network node such as eNodeB, provides two sets of parameters that can be used in the design of adaptation state machines for speech and video respectively. The parameters are contrived such that dependence on media codec or radio access bearer technology is avoided as much as possible, not to constrain the evolution of these elements. In addition, vendor specific parameters taking advantage of the implementation can be placed under Ext nodes.

By altering the parameters of the MO via OMA-DM protocol, media adaptation behavior of the MTSI client in terminal can be modified up to extent allowed by the implementation. Note that due to the underlying uncertainties and complexities, one should expect only to shape the expected bit rate trajectory of multimedia stream over time-varying transmission conditions, rather than to control the media flow in a timely and stringent manner. Detailed descriptions of the speech and video adaptation parameters can be found in table 17.1 and 17.2.

The Management Object Identifier shall be: urn:oma:mo:ext-3gpp-mtsima:1.0.

Protocol compatibility: The MO is compatible with OMA Device Management protocol specifications, version 1.2 and upwards, and is defined using the OMA DM Device Description Framework as described in the Enabler Release Definition OMA-ERELED_DM-V1_2 [67].

17.2 Media adaptation management object

The following nodes and leaf objects in figure 17.1 shall be contained under the 3GPP_MTSIMA node if the MTSI client in terminal supports the feature described in this clause. Information of DDF for this MO is given in Annex J.

Node: /<X>

This interior node specifies the unique object id of a MTSI media adaptation management object. The purpose of this interior node is to group together the parameters of a single object.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

The following interior nodes shall be contained if the MTSI client in terminal supports the "MTSI media adaptation management object".

/<X>/Speech

The Speech node is the starting point of parameters related to speech adaptation if any speech codec are available.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/<X>/Speech/<X>

This interior node is used to allow a reference to a list of speech adaptation parameters.

- Occurrence: OneOrMore
- Format: node
- Minimum Access Types: Get

/<X>/Speech/<X>/ID

This leaf node represents the identification number of a set of parameters related to speech adaptation.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/TAG

This leaf node represents the identification tag of a set of parameters for speech adaptation. It is recommended to have at least a node, for example, ID, TAG, or implementation-specific ones, for the identification purpose such that each set of parameters can be distinguished and accessed.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/<X>/Speech/<X>/PLR

This interior node is used to allow a reference to a list of parameters related to packet loss rate (PLR).

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /PLR/MAX

This leaf node represents the maximum PLR tolerated when redundancy is not used, before the receiver signals the sender to attempt adaptation that reduces PLR or operate at modes more robust to packet loss.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /PLR/LOW

This leaf node represents the minimum PLR tolerated, before the receiver signals the sender to probe for higher bit rate, increase the packet rate, reduce redundancy, or perform other procedures that could improve speech quality under such favorable conditions.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /PLR/STATE_REVERSION

This leaf node represents the maximum PLR tolerated after adaptation state machine has taken actions, based on the measured PLR lower than LOW. Once PLR exceeds this threshold, the receiver decides that the actions taken to improve speech quality were not successful.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /PLR/RED_INEFFECTIVE

This leaf node represents the maximum PLR tolerated, after adaptation state machine has taken actions to increase redundancy. Once PLR exceeds this threshold, the receiver decides that the situation was not improved but degraded.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/ $\langle X \rangle$ /Speech/ $\langle X \rangle$ /PLR/DURATION_MAX

This leaf node represents the duration (ms) of sliding window over which PLR is observed and computed. The computed value is compared with the MAX threshold.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/PLR/DURATION_LOW

This leaf node represents the duration (ms) of sliding window over which PLR is observed and computed. The computed value is compared with the LOW threshold.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/PLR/DURATION_STATE_REVERSION

This leaf node represents the duration (ms) of sliding window over which PLR is observed and computed. The computed value is compared with the STATE_REVERSION threshold.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/PLR/DURATION_RED_INEFFECTIVE

This leaf node represents the duration (ms) of sliding window over which PLR is observed and computed. The computed value is compared with the RED_INEFFECTIVE threshold.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/PLR/DURATION

This leaf node represents the duration (ms) of sliding window over which PLR is observed and computed. The computed value is compared with the PLR thresholds.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/PLB

This interior node is used to allow a reference to a list of parameters related to an event, packet loss burst (PLB), in which a large number of packets are lost during a limited period.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/<X>/Speech/<X>/PLB/LOST_PACKET

This leaf node represents the number of packets lost during a period of PLB/DURATION.

- Occurrence: One
- Format: int
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/PLB/DURATION

This leaf node represents the period (ms) for which LOST_PACKET is counted.

- Occurrence: One
- Format: int
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/ECN

This interior node is used to allow a reference to a list of parameters related to Explicit Congestion Notification (ECN) to IP.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/ECN/USAGE

This leaf node represents a Boolean parameter that enables or disables ECN-based adaptation.

- Occurrence: ZeroOrOne
- Format: bool
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/ECN/MIN_RATE

This leaf node represents the minimum bit rate (bps, excluding IP, UDP, RTP and payload overhead) that speech encoder should use during ECN-based adaptation.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/ECN/STEPWISE_DOWNSWITCH

This leaf node represents a Boolean parameter that selects which down-switch method to use, i.e., direct or step-wise, for ECN-triggered adaptation.

- Occurrence: ZeroOrOne
- Format: bool
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/ECN/RATE_LIST

This leaf node represents the list of bit rates to use during stepwise down-switch. This parameter is only applicable when stepwise down-switch is used.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/<X>/Speech/<X>/ECN/INIT_WAIT

This leaf node represents the time (ms) that the sender should wait before an up-switch is attempted in the beginning of the session if no rate control information or reception quality feedback information is received.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/ECN/INIT_UPSWITCH_WAIT

This leaf node represents the time (ms) that the sender should wait at each step during up-switch in the beginning of the session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/ECN/CONGESTION_WAIT

This leaf node represents the minimum interval (ms) between detection of ECN-CE and up-switch from the reduced rate.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/ECN/CONGESTION_UPSWITCH_WAIT

This leaf node represents the waiting time (ms) at each step during up-switch after a congestion event, except for the initial up-switch which uses the ECN/CONGESTION_WAIT time.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/ICM

This interior node is used to allow a reference to a list of parameters related to Initial Codec Mode (ICM).

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/<X>/Speech/<X>/ICM/INITIAL_CODEC_RATE

This leaf node represents the bit rate (bps, excluding IP, UDP, RTP and payload overhead) that the speech encoder should use when starting the encoding in the beginning of the session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/ICM/INITIAL_CODEC_BANDWIDTH

This leaf node represents the audio bandwidth that the EVS speech encoder should use when starting the encoding in the beginning of the session, unless specified by bw, bw-send, or bw-recv parameter.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get
- Values: nb, wb, swb, fb

/<X>/Speech/<X>/ICM/INIT_WAIT

This leaf node represents the time (ms) that the sender should wait before an up-switch is attempted in the beginning of the session if no rate control information or reception quality feedback information is received.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/ICM/INIT_UPSWITCH_WAIT

This leaf node represents the time (ms) that the sender should wait at each step during up-switch in the beginning of the session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/ ICM/INIT_PARTIAL_REDUNDANCY_OFFSET_SEND

This leaf node represents the initial partial redundancy offset (-1, 0, 2, 3, 5, or 7) that the EVS speech encoder should use when starting the encoding in the beginning of the session that uses channel aware mode, unless asked otherwise by the far-end MTSI client in terminal with the ch-aw-recv parameter .

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/ ICM/INIT_PARTIAL_REDUNDANCY_OFFSET_RECV

This leaf node represents the initial partial redundancy offset (-1, 0, 2, 3, 5, or 7) that the MTSI client in terminal should ask the far-end MTSI client in terminal with the ch-aw-recv parameter to use when starting the encoding in the beginning of the session that uses channel aware mode.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/MEDIA_ROBUSTNESS

This interior node is used to allow a reference to a list of parameters related to Media Robustness Adaptation that can be used for the CHEM feature. Each unique codec type is identified by the CODEC_ID under a corresponding instance of the MEDIA_ROBUSTNESS node which groups the parameters associated with the codec type/CODEC_ID.

- Occurrence: ZeroOrMore
- Format: node
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/MEDIA_ROBUSTNESS/CODEC_ID

This leaf node represents the codec MIME type.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/MEDIA_ROBUSTNESS/TAG

This leaf node represents the identification tag of a set of parameters for speech robustness adaptation of a codec type identified by the CODEC_ID. It is recommended to have at least a node, for example, TAG, or implementation-specific ones, for the identification purpose such that each set of parameters can be distinguished and accessed.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/MEDIA_ROBUSTNESS/CFG_BIT_RATE_LIST

This interior node is used to provide a list of the bit rates of the configurations of the codec type (CODEC_ID) listed from the bit rate of the least robust configuration first to the bit rate of the most robust listed last.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/MEDIA_ROBUSTNESS/CFG_RED_LIST

This interior node is used to provide a list of the redundancy levels of the configurations of the codec type (CODEC_ID) listed from the redundancy level of the least robust configuration first to the redundancy level of the most robust listed last.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/MEDIA_ROBUSTNESS/HIGH_PLR_THRESH_LIST

This interior node is used to provide a list of the high PLR thresholds for each codec configuration except for the most robust configuration. A high PLR threshold for a given codec configuration is the highest tolerable PLR at that codec configuration before the MTSI client requests a more robust codec configuration that will yield lower PLR.

- Occurrence: One

- Format: chr
- Minimum Access Types: Get

/<X>/Speech/<X>/MEDIA_ROBUSTNESS/LOW_PLR_THRESH_LI ST

This interior node is used to provide a list of the low PLR thresholds for each codec configuration except for the least robust configuration. A low PLR threshold for a given codec configuration is the lowest tolerable PLR at that codec configuration before the MTSI client requests a less robust codec configuration that will yield better quality.

- Occurrence: One
- Format: chr
- Minimum Access Types: Get

/<X>/Speech/<X>/MEDIA_ROBUSTNESS/DJB_PLR

This interior node indicates whether the estimated PLR is measured before or after de-jitter buffering.

- Occurrence: One
- Format: boolean
- Minimum Access Types: Get

/<X>/Speech/<X>/MEDIA_ROBUSTNESS/PLR_AVG_WINDOW

This interior node indicates the duration of the sliding window used by the media receiver to estimate the received PLR.

- Occurrence: One
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/N_INHIBIT

This leaf node represents the period (number of speech frames) for which adaptation is disabled to avoid the ping-pong effects, when adaptation state machine transitions from one state to another then back to the original state.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/N_HOLD

This leaf node represents the period (proportion of PLR/DURATION) that can substitute other periods such as DURATION_LOW or DURATION_RED_INEFFECTIVE, when they are not available.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Speech/<X>/T_RESPONSE

This leaf node represents the expected response time (ms) for a request to be fulfilled. If a request transmitted to the far-end is not granted within a period of T_RESPONSE, the request can be considered lost during transmission or the far-end MTSI client in terminal might have decided not to grant it.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Speech/`<X>`/Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video

The Video node is the starting point of parameters related to video adaptation if any video codec are available.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`

This interior node is used to allow a reference to a list of video adaptation parameters.

- Occurrence: OneOrMore
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ID

This leaf node represents the identification number of a set of parameters related to video adaptation.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/TAG

This leaf node represents the identification tag of a set of parameters for video adaptation. It is recommended to have at least a node, for example, ID, TAG, or implementation-specific ones, for the identification purpose such that each set of parameters can be distinguished and accessed.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/PLR

This interior node is used to allow a reference to a list of parameters related to PLR.

- Occurrence: ZeroOrOne

- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/PLR/MAX

This leaf node represents the maximum PLR tolerated, before the receiver signals the sender to reduce the bit rate such that PLR is reduced.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/`<X>`/Video/`<X>`/PLR/LOW

This leaf node represents the minimum PLR tolerated, before the receiver signals the sender to increase the bit rate.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/`<X>`/Video/`<X>`/PLR/DURATION_MAX

This leaf node represents the duration (ms) of sliding window over which PLR is observed and computed. The computed value is compared with the MAX threshold.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/PLR/DURATION_LOW

This leaf node represents the duration (ms) of sliding window over which PLR is observed and computed. The computed value is compared with the LOW threshold.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/PLB

This interior node is used to allow a reference to a list of parameters related to PLB.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/PLB/LOST_PACKET

This leaf node represents the number of packets lost during a period of PLB/DURATION.

- Occurrence: One
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/PLB/DURATION

This leaf node represents the period (ms) for which LOST_PACKET is counted.

- Occurrence: One
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/MIN_QUALITY

This interior node is used to allow a reference to a list of parameters related to the minimum video quality.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/MIN_QUALITY/BIT_RATE

This interior node is used to allow a reference to a list of parameters related to the minimum bit rate.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/MIN_QUALITY/BIT_RATE/ABSOLUTE

This leaf node represents the minimum bit rate (kbps) that video encoder should use.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/MIN_QUALITY/BIT_RATE/RELATIVE

This leaf node represents the minimum bit rate (proportion of the bit rate negotiated for the video session) that video encoder should use.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/`<X>`/Video/`<X>`/MIN_QUALITY/FRAME_RATE

This interior node is used to allow a reference to a list of parameters related to the minimum frame rate.

- Occurrence: ZeroOrOne

- Format: node
- Minimum Access Types: Get

/ $\langle X \rangle$ /Video/ $\langle X \rangle$ /MIN_QUALITY/FRAME_RATE/ABSOLUTE

This leaf node represents the minimum frame rate (fps, frames per second) that video encoder should use.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get

/ $\langle X \rangle$ /Video/ $\langle X \rangle$ /MIN_QUALITY/FRAME_RATE/RELATIVE

This leaf node represents the minimum frame rate (proportion of the maximum frame rate limited by the codec profile/level negotiated for the video session) that video encoder should use.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/ $\langle X \rangle$ /Video/ $\langle X \rangle$ /MIN_QUALITY/QP

This interior node is used to allow a reference to a list of parameters related to video quantisation.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/ $\langle X \rangle$ /Video/ $\langle X \rangle$ /MIN_QUALITY/QP/H264

This leaf node represents the maximum value of luminance quantization parameter QP_Y that video encoder should use if H.264 is negotiated for the video session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get
- Values: 0 ~ 51

/ $\langle X \rangle$ /Video/ $\langle X \rangle$ /ECN

This interior node is used to allow a reference to a list of parameters related to Explicit Congestion Notification (ECN) to IP.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/ $\langle X \rangle$ /Video/ $\langle X \rangle$ /ECN/STEP_UP

This leaf node represents the proportion of current encoding rate estimated by video receiver, which is used to ask video sender to increase the rate by this value.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ECN/STEP_DOWN

This leaf node represents the decrease in the requested maximum encoding rate over current rate, when a down-switch is requested by the receiver.

- Occurrence: ZeroOrOne
- Format: chr
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ECN/INIT_WAIT

This leaf node represents the minimum waiting time (ms) before up-switch is attempted in the initial phase of the session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ECN/INIT_UPSWITCH_WAIT

This leaf node represents the waiting time (ms) at each step during up-switch in the beginning of the session.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ECN/CONGESTION_WAIT

This leaf node represents the minimum interval (ms) between detection of ECN-CE and up-switch from the reduced rate.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ECN/CONGESTION_UPSWITCH_WAIT

This leaf node represents the waiting time (ms) at each step during up-switch after a congestion event, except for the initial up-switch which uses the ECN/CONGESTION_WAIT time.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/ECN/MIN_RATE

This interior node is used to allow a reference to a list of parameters related to the minimum bit rate during ECN-based adaptation.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/<X>/Video/<X>/ECN/MIN_RATE/ABSOLUTE

This leaf node represents the minimum bit rate (kbps, excluding IP, UDP, RTP and payload overhead) that video encoder should use during ECN-based adaptation.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get

/<X>/Video/<X>/ECN/MIN_RATE/RELATIVE

This leaf node represents the minimum bit rate (proportion of the bit rate negotiated for the video session) that video encoder should use during ECN-based adaptation.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get

/<X>/Video/<X>/RTP_GAP

This leaf node represents the maximum interval between packets (proportion of the estimated frame period) tolerated, before the receiver declares bursty packet loss or severe congestion condition.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get

/<X>/Video/INC_FBACK_MIN_INTERVAL

This leaf node represents the minimum interval (ms) at which rate adaptation feedback such as TMMBR should be sent from the receiver to the sender, when the bit rate is being increased.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Video/<X>/DEC_FBACK_MIN_INTERVAL

This leaf node represents the minimum interval (ms) at which rate adaptation feedback such as TMMBR should be sent from the receiver to the sender, when the bit rate is being decreased.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Video/<X>/TP_DURATION_HI

This leaf node represents the duration (ms) of sliding window over which the interval between packet arrival and playout is observed. The computed value is compared with TARGET_PLAYOUT_MARGIN_HI.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Video/<X>/TP_DURATION_MIN

This leaf node represents the duration (ms) of sliding window over which the interval between packet arrival and playout is observed. The computed value is compared with TARGET_PLAYOUT_MARGIN_MIN.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Video/<X>/TARGET_PLAYOUT_MARGIN_HI

This leaf node represents the upper threshold of the interval (ms) between packet arrival and its properly scheduled playout.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Video/<X>/TARGET_PLAYOUT_MARGIN_MIN

This leaf node represents the lower threshold of the interval (ms) between packet arrival and its properly scheduled playout.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/<X>/Video/<X>/RAMP_UP_RATE

This leaf node represents the rate (kbps/s) at which video encoder should increase its maximum bit rate from current value to the value indicated in the most recently received TMMBR message.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get

/<X>/Video/<X>/RAMP_DOWN_RATE

This leaf node represents the rate (kbps/s) at which video encoder should decrease its maximum bit rate from current value to the value indicated in the most recently received TMMBR message.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/DECONGEST_TIME

This leaf node represents the time (ms) the receiver should command the sender to spend in decongesting the transmission path, before attempting to transmit at the sustainable rate of the path.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get

/`<X>`/Video/`<X>`/HOLD_DROP_END

This leaf node represents a tri-valued parameter that controls how the sender should behave in case video quality cannot meet the requirements set in BIT_RATE, FRAME_RATE, or QP.

- Occurrence: ZeroOrOne
- Format: int
- Minimum Access Types: Get
- Values: 0, 1, 2

/`<X>`/Video/`<X>`/INITIAL_CODEC_RATE

This leaf node represents the initial bit rate (proportion of the bit rate negotiated for the video session) that the sender should begin encoding video at.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/`<X>`/Video/`<X>`/X_PERCENTILE

This leaf node represents the percentile point of packet arrival distribution used with the TARGET_PLAYOUT_MARGIN parameters.

- Occurrence: ZeroOrOne
- Format: float
- Minimum Access Types: Get
- Values: 0 ~ 100 %

/`<X>`/Video/`<X>`/Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

/`<X>`/Ext

The Ext is an interior node where the vendor specific information can be placed (vendor meaning application vendor, device vendor etc.). Usually the vendor extension is identified by vendor specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include one or more un-standardized sub-trees.

- Occurrence: ZeroOrOne
- Format: node
- Minimum Access Types: Get

Table 17.1: Speech adaptation parameters of 3GPP MTSIMA MO

| Parameter (Unit) | Usage |
|-----------------------------------|--|
| PLR/MAX (%) | Packet loss rate (PLR) above this threshold, when redundancy is not used, indicates that performance is not satisfactory. Adaptation state machine at the receiver should signal the sender to attempt adaptation that reduces PLR or operate at modes more robust to packet loss. When using the example adaptation state machines of Annex C, this parameter corresponds to PLR_1. |
| PLR/LOW (%) | PLR below this threshold indicates that conditions are favorable and better quality can be supported. Adaptation state machine at the receiver should signal the sender to probe for higher bit rate, increase the packet rate, reduce redundancy, or perform other procedures that could improve speech quality under such favorable conditions. When in the probing state, if PLR falls below this threshold, then the sender should adapt to a higher bit rate. When using the example adaptation state machines of Annex C, this parameter corresponds to PLR_2. |
| PLR/STATE_REVERSION (%) | PLR above this threshold, after adaptation state machine has taken actions based on PLR lower than LOW, indicates that the actions taken to improve speech quality were not successful. Adaptation state machine at the receiver should signal the sender to return to the previous state where it stayed before attempting to improve speech quality. When using the example adaptation state machines of Annex C, this parameter corresponds to PLR_3. |
| PLR/RED_INEFFECTIVE (%) | PLR above this threshold, after adaptation state machine has taken actions to increase redundancy, indicates that situation was not improved but degraded. Adaptation state machine at the receiver should signal the sender to use a lower bit rate and no redundancy. When using the example adaptation state machines of Annex C, this parameter corresponds to PLR_4. |
| PLR/DURATION_MAX (ms) | Duration of sliding window over which PLR is observed and computed. The computed value is compared with the MAX threshold. |
| PLR/DURATION_LOW (ms) | Duration of sliding window over which PLR is observed and computed. The computed value is compared with the LOW threshold. |
| PLR/DURATION_STATE_REVERSION (ms) | Duration of sliding window over which PLR is observed and computed. The computed value is compared with the STATE_REVERSION threshold. |
| PLR/DURATION_RED_INEFFECTIVE (ms) | Duration of sliding window over which PLR is observed and computed. The computed value is compared with the RED_INEFFECTIVE threshold. |
| PLR/DURATION (ms) | Duration of sliding window over which PLR is observed and computed. The computed value is compared with the PLR thresholds. This applies as the default duration in case no specific DURATION is specified. |
| PLB/LOST_PACKET (integer) | When loss of LOST_PACKET or more packets is detected in the latest period of PLB/DURATION, this event is categorized as a packet loss burst (PLB) and adaptation state machine should take appropriate actions to reduce the impact on speech quality. |
| PLB/DURATION (ms) | Duration of sliding window over which lost packets are counted. |
| ECN/USAGE (Boolean) | Switch to enable or disable ECN-based adaptation. This parameter should be translated as follows: "0" = OFF, "1" = ON. |

| | |
|-----------------------------------|---|
| ECN/MIN_RATE (bps) | <p>Lower boundary for the media bit-rate adaptation in response to ECN-CE marking. The media bit-rate shall not be reduced below this value as a reaction to the received ECN-CE. The value of this parameter is assigned to the ECN_min_rate parameter defined in Clause 10.2.0.</p> <p>The ECN_min_rate should be selected to maintain an acceptable service quality while reducing the resource utilization.</p> <p>Default value: Same as ICM/INITIAL_CODEC_RATE if defined, otherwise same as Initial Codec Mode (ICM), see Clause 7.5.2.1.6.</p> |
| ECN/STEPWISE_DOWNSWITCH (Boolean) | <p>Switch to select down-switch method. This parameter should be translated as follows: "0" = direct down-switch to ECN/MIN_RATE; "1" = stepwise down-switch according to ECN/RATE_LIST (one step per congestion event).</p> |
| ECN/RATE_LIST (character set) | <p>List of bit rates (e.g. codec modes) to use during stepwise down-switch. This parameter is only applicable when stepwise down-switch is used. If the codec does not support exactly the rate which is indicated then the highest rate supported by the codec below the indicated value should be used. Depending on the codec, the values can be understood as either the highest rate or the average rate. The entries in the list may either be generic, i.e. usable for any codec, but can also be codec-specific.</p> <p>The default usage is the generic list where the bit rates [in bps] are included, e.g. (5000, 6000, 7500, 12500). A codec-specific list may indicate desired modes, e.g. for AMR the list could be (0,2,4,7).</p> <p>The use of certain rates in this list may be prevented by the results of session negotiation involving SDP attributes such as the "mode-set" parameter. The SDP parameter "mode-change-neighbor" may lead to using intermediate modes when transitioning between rates in this list.</p> <p>If this parameter is not defined or contains bit rates not negotiated in the session, then the mode-set included in SDP is used. If no mode-set is defined in SDP, then "4750, 5900, 7400, 12200" is used for AMR, which corresponds to the "0, 2, 4, 7" modes.</p> |
| ECN/INIT_WAIT (ms) | <p>The waiting time before the first up-switch is attempted in the beginning of the session, to avoid premature up-switch. This parameter shall be used instead of the ICM/INIT_WAIT parameter if ECN is used in the session.</p> <p>Default value is defined in Clause 7.5.2.1.6.</p> |
| ECN/INIT_UPSWITCH_WAIT (ms) | <p>This parameter is used in up-switches in the beginning of the session. Note that the first up-switch in the beginning of the session uses the ECN/INIT_WAIT time. Only the subsequent up-switches use the ECN/INIT_UPSWITCH_WAIT time. This parameter shall be used instead of the ICM/INIT_UPSWITCH_WAIT parameter if ECN is used in the session.</p> <p>Default value: is defined in Clause 7.5.2.1.6.</p> |
| ECN/CONGESTION_WAIT (ms) | <p>The waiting time after an ECN-CE marking for which an up-switch shall not be attempted. The value of this parameter is assigned to the ECN_congestion_wait parameter defined in Clause 10.2.0.</p> <p>A negative value indicates an infinite waiting time, i.e. to prevent up-switch for the whole remaining session.</p> <p>Default value: Same as the ECN_congestion_wait parameter defined in Clause 10.2.0.</p> |
| ECN/CONGESTION_UPSWITCH_WAIT (ms) | <p>This parameter is used in up-switches after a congestion event. Note that the first up-switch after a congestion event uses the ECN/CONGESTION_WAIT time. Only the subsequent up-switches use the ECN/CONGESTION_UPSWITCH_WAIT time.</p> <p>Default value is 5000 ms.</p> |
| ICM/INITIAL_CODEC_RATE (bps) | <p>The bit rate that the speech encoder should use for the encoding of the speech at the start of the RTP stream.</p> |

| | |
|---|--|
| ICM/INITIAL_CODEC_BANDWIDTH (character set) | The audio bandwidth that the EVS speech encoder in EVS Primary mode should use for the encoding of the speech at the start of the RTP stream. |
| ICM/INIT_WAIT (ms) | To avoid premature up-switch when ECN is not used in the session, this parameter defines the waiting time before the first up-switch is attempted in the beginning of the session. Default value: Same as Initial Waiting Time as defined in Clause 7.5.2.1.6. |
| ICM/INIT_UPSWITCH_WAIT (ms) | When ECN is not used in the session, this parameter is used in up-switches in the beginning of the session until the first down-switch occurs. Note that the first up-switch in the beginning uses the INIT_WAIT time. Only the subsequent up-switches use the INIT_UPSWITCH_WAIT time. Default value: Same as Initial Upswitch Waiting Time as defined in Clause 7.5.2.1.6. |
| ICM/INIT_PARTIAL_REDUNDANCY_OFFSET_SEND (integer) | The initial partial redundancy offset (-1, 0, 2, 3, 5, or 7) that the EVS speech encoder should use when starting the encoding in the beginning of the session that uses channel aware mode, unless asked otherwise by the far-end MTSI client in terminal with the ch-aw-recv parameter. |
| ICM/INIT_PARTIAL_REDUNDANCY_OFFSET_RECV (integer) | The initial partial redundancy offset (-1, 0, 2, 3, 5, or 7) that the MTSI client in terminal should ask the far-end MTSI client in terminal to use with the ch-aw-recv parameter when starting the encoding in the beginning of the session that uses channel aware mode. |
| N_INHIBIT (integer) | If adaptation state machine transitions from one state to another then back to the original state, adaptation state machine should not return to the other state in less than N_INHIBIT speech frames, to avoid the ping-pong effects. |
| N_HOLD (integer) | N_HOLD x PLR/DURATION can be used as the period for which PLR is observed and computed. For example, the computed value can be compared with the LOW threshold when DURATION_LOW is not defined. |
| T_RESPONSE (ms) | If the receiver does not detect expected responses from the sender within a period of T_RESPONSE after having sent a request, the receiver should consider this request as not fulfilled and take appropriate actions. |
| CODEC_ID (character set) | MIME Type of the codec for which the media robustness adaptation PLR thresholds are configured. |

| <p>CFG_BIT_RATE_LIST (character set)</p> | <p>List of bit rates (or codec modes) describing the codec configurations to use during media robustness adaptation. The entries are listed in order of the bit rate of the least robust configuration listed first to the bit rate of the most robust configuration listed last. If there are multiple codec configurations with the same bit rate but different loss robustness (e.g., EVS 13.2 channel aware and non-channel aware modes, or a codec mode with different levels of application layer redundancy), the same bit rate is listed multiple times in the list.</p> <p>If the codec does not support exactly the rate which is indicated, then the highest rate supported by the codec below the indicated value should be used. Depending on the codec, the values can be understood as either the highest rate or the average rate.</p> <p>The entries in the list may either be generic, i.e. usable for any codec, but can also be codec-specific.</p> <p>The default usage is the generic list where the bit rates [in bps] are included, e.g. (5000, 6000, 7500, 12500).</p> <p>A codec-specific list may indicate desired modes, e.g. for AMR the list could be (0,2,4,7).</p> <p>The use of certain rates in this list may be prevented by the results of session negotiation involving SDP attributes such as the "mode-set" parameter or "b=AS" attribute. The SDP parameter "mode-change-neighbor" may lead to using intermediate modes when transitioning between rates in this list.</p> <p>If this parameter is not defined or contains bit rates not negotiated in the session, then the rates or mode-set included in SDP is used.</p> | | | | | | | | | | | | |
|--|---|-------|-------------|---|---------------|---|--------------------|---|--|---|--|---|--|
| <p>CFG_RED_LIST (character set)</p> | <p>List of redundancy levels describing the codec configurations to use during media robustness adaptation. The redundancy levels are listed in order to correspond to respective CFG_BIT_RATE_LIST entries and describe the redundancy levels from the least robust configuration first to the most robust last. The redundancy level is described using one of the values below:</p> <table border="1" data-bbox="833 1234 1406 1491"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No redundancy</td> </tr> <tr> <td>P</td> <td>Partial Redundancy</td> </tr> <tr> <td>1</td> <td>100% repetition application layer redundancy</td> </tr> <tr> <td>2</td> <td>200% repetition application layer redundancy</td> </tr> <tr> <td>3</td> <td>300% repetition application layer redundancy</td> </tr> </tbody> </table> <p>Commas are used to separate redundancy levels of each codec configuration in the list.</p> <p>If the codec configuration does not support the redundancy level, then the codec configuration shall not be requested by the media receiver for media robustness adaptation.</p> | Value | Description | 0 | No redundancy | P | Partial Redundancy | 1 | 100% repetition application layer redundancy | 2 | 200% repetition application layer redundancy | 3 | 300% repetition application layer redundancy |
| Value | Description | | | | | | | | | | | | |
| 0 | No redundancy | | | | | | | | | | | | |
| P | Partial Redundancy | | | | | | | | | | | | |
| 1 | 100% repetition application layer redundancy | | | | | | | | | | | | |
| 2 | 200% repetition application layer redundancy | | | | | | | | | | | | |
| 3 | 300% repetition application layer redundancy | | | | | | | | | | | | |

| HIGH_PLR_THRESH_LIST (character set) | <p>List of high PLR thresholds for each codec configuration in the order described for the CFG_BIT_RATE_LIST with the exception of not having a high PLR threshold for the most robust codec configuration, i.e., the last entry in the HIGH_PLR_THRESH_LIST corresponds to the threshold for requesting the most robust configuration when using the second most, or a less, robust configuration.</p> <p>When the estimated PLR exceeds a PLR threshold in this list corresponding to a given codec configuration, the media receiver shall request the next more robust codec configuration. E.g., if the first high PLR threshold in the list, which corresponds to the least robust codec configuration is exceeded, then the media receiver requests the second least robust codec configuration.</p> <p>The PLR values are represented as a percent (e.g., 2.5 is 2.5% PLR) and separated by commas for each codec configuration.</p> | | | | | | |
|--------------------------------------|---|-------|-------------|---|---------------------|---|----------------------|
| LOW_PLR_THRESH_LIST (character set) | <p>List of low PLR thresholds for each codec configuration in the order described for the CFG_BIT_RATE_LIST with the exception of not having a low PLR threshold for the least robust codec configuration, i.e., the first entry in the LOW_PLR_THRESH_LIST corresponds to the threshold for requesting the least robust configuration when using the second least, or a more, robust configuration.</p> <p>When the estimated PLR drops below a PLR threshold in this list corresponding to a given codec configuration, the media receiver shall request the next less robust codec configuration. E.g., if the estimated PLR drops below the last low PLR threshold in the list, which corresponding to the most robust codec configuration, then the media receiver requests the second most robust codec configuration.</p> <p>The PLR values are represented as a percent (e.g., 2.5 is 2.5% PLR) and separated by commas for each codec configuration.</p> | | | | | | |
| DJB_PLR (Boolean) | <table border="1" data-bbox="831 1153 1406 1243"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Measure PLR pre-DJB</td> </tr> <tr> <td>1</td> <td>Measure PLR post-DJB</td> </tr> </tbody> </table> | Value | Description | 0 | Measure PLR pre-DJB | 1 | Measure PLR post-DJB |
| Value | Description | | | | | | |
| 0 | Measure PLR pre-DJB | | | | | | |
| 1 | Measure PLR post-DJB | | | | | | |
| PLR_AVG_WINDOW (ms) | <p>Indicates the duration of the sliding window (in ms) over which the PLR is observed and computed.</p> | | | | | | |

Table 17.2: Video adaptation parameters of 3GPP MTSIMA MO

| Parameter (Unit) | Usage |
|---------------------------|---|
| PLR/MAX (%) | Upper threshold of PLR above which adaptation state machine at the receiver should signal the sender to reduce the bit rate. PLR is measured per RTP packet and in addition to packets that do not arrive at the receiver ever, packets that arrive but do not make it in time for their properly scheduled playout are considered as lost. |
| PLR/LOW (%) | Lower threshold of PLR below which adaptation state machine at the receiver may signal the sender to increase the bit rate. |
| PLR/DURATION_MAX (ms) | Duration of sliding window over which PLR is observed and computed. The computed value is compared with the MAX threshold. |
| PLR/DURATION_LOW (ms) | Duration of sliding window over which PLR is observed and computed. The computed value is compared with the LOW threshold. |
| PLB/LOST_PACKET (integer) | When loss of LOST_PACKET or more packets is detected in the last period of PLB/DURATION, this event is categorized as a packet loss burst (PLB) and adaptation state machine should take appropriate actions to reduce the impact on video quality. |
| PLB/DURATION (ms) | Duration of sliding window over which lost packets are counted. |

| | |
|---|---|
| MIN_QUALITY/BIT_RATE /ABSOLUTE (kbps) | Minimum bit rate that video encoder should use. If the MTSI client in terminal is unable to maintain this minimum bit rate, it should drop the video stream component or put it on hold. If both MIN_QUALITY/BIT_RATE/ABSOLUTE and MIN_QUALITY/BIT_RATE/RELATIVE are set, the larger of these two shall be used as the minimum bit rate. |
| MIN_QUALITY/BIT_RATE /RELATIVE (%) | Minimum bit rate (as a proportion of the bit rate negotiated for the video session) that the video encoder should use. If the MTSI client in terminal is unable to maintain this minimum bit rate, it should drop the video stream component or put it on hold. If both MIN_QUALITY/BIT_RATE/ABSOLUTE and MIN_QUALITY/BIT_RATE/RELATIVE are set, the larger of these two shall be used as the minimum bit rate. |
| MIN_QUALITY/FRAME_RATE /ABSOLUTE (fps) | Minimum frame rate that video encoder should use. If the MTSI client in terminal is unable to maintain this minimum frame rate, it should drop the video stream component or put it on hold. The minimum frame rate is considered unmet if the interval between encoding times of video frames is larger than the reciprocal of the minimum frame rate. If both MIN_QUALITY/FRAME_RATE/ABSOLUTE and MIN_QUALITY/FRAME_RATE/RELATIVE are set, the larger of these two shall be used as the minimum frame rate. |
| MIN_QUALITY/FRAME_RATE /RELATIVE (%) | Minimum frame rate (as a proportion of the maximum frame rate supported as specified by the video codec profile/level negotiated for the session) that video encoder should use. If the MTSI client in terminal is unable to maintain this minimum frame rate, it should drop the video stream component or put it on hold. The minimum frame rate is considered unmet if the interval between encoding times of video frames is larger than the reciprocal of the minimum frame rate. If both MIN_QUALITY/FRAME_RATE/ABSOLUTE and MIN_QUALITY/FRAME_RATE/RELATIVE are set, the larger of these two shall be used as the minimum frame rate. |
| MIN_QUALITY/QP/H264 (integer) | Maximum value of QP _Y that video encoder should use if H.264 is negotiated for the video session. The encoder should generate video stream such that QP _Y does not exceed H264. If the MTSI client in terminal is unable to maintain this maximum QP _Y value, it should drop the video stream component or put it on hold. |
| ECN/STEP_UP (%) | When an up-switch is requested by the receiver, this parameter defines the proportion of the session media bandwidth (b=AS) that is used to increment the requested maximum encoding rate over the currently used rate. The receiver estimates the currently used rate over an implementation dependent time period. Default value: 10. |
| ECN/STEP_DOWN (character set) | List of proportions (%) by which video receiver requests that the encoder rate be reduced relative to the currently used rate in response to each congestion event. The receiver estimates the currently used rate over an implementation dependent time period. The receiver uses the first value in the list for the first congestion event, the second value for the second congestion event etc. The list may consist of only one value. If there are more congestion events than there are values in the list, then the last value is used for each additional congestion event. The receiver resets to use the first value in the list after an up-switch has started i.e. after the CONGESTION_WAIT time. Default Value: "30, 20, 10". |
| ECN/INIT_WAIT (ms) | The waiting time before the first up-switch is attempted in the initial phase of the session, to avoid premature up-switch. Default value is 500 ms. The initial phase starts at the beginning of the session and ends when the first congestion event is detected. |
| ECN/INIT_UPSWITCH_WAIT (ms) | This parameter is the waiting time used before attempting up-switches in the initial phase of the session. Note that the first up-switch in the initial phase uses the INIT_WAIT time. Only the subsequent up-switches use the INIT_UPSWITCH_WAIT time. Default value: 500 ms. |
| ECN/CONGESTION_WAIT (ms) | The waiting time after an ECN-CE marking for which an up-switch shall not be attempted. A negative value indicates an infinite waiting time, i.e. to prevent up-switch for the whole remaining session. Default value: 5000 ms. |

| | |
|-----------------------------------|---|
| ECN/CONGESTION_UPSWITCH_WAIT (ms) | This parameter is the waiting time used before attempting up-switches after a congestion event. Note that the first up-switch after a congestion event uses the CONGESTION_WAIT time. Only the subsequent up-switches use the CONGESTION_UPSWITCH_WAIT time. Default value is 5000 ms. |
| ECN/MIN_RATE/ABSOLUTE (kbps) | Lower boundary for the media bit-rate adaptation in response to ECN-CE marking. The media bit-rate shall not be reduced below this value as a reaction to the received ECN-CE. The ECN/MIN_RATE/ABSOLUTE should be selected to maintain an acceptable service quality while reducing the resource utilization. If the GBR is known to the client to be lower than the ECN/MIN_RATE then the GBR value shall be used instead of the ECN/MIN_RATE value. Default value: 48 kbps. If both ECN/MIN_RATE/ABSOLUTE and ECN/MIN_RATE/RELATIVE are set, the larger of these two shall be used as the lower boundary for the media bit-rate adaptation in response to ECN-CE marking. |
| ECN/MIN_RATE/RELATIVE (%) | Lower boundary (as a proportion of the bit rate negotiated for the video session) for the media bit-rate adaptation in response to ECN-CE marking. The media bit-rate shall not be reduced below this value as a reaction to the received ECN-CE. The ECN/MIN_RATE/RELATIVE should be selected to maintain an acceptable service quality while reducing the resource utilization. If the GBR is known to the client to be lower than the ECN/MIN_RATE then the GBR value shall be used instead of the ECN/MIN_RATE value. Default value: Same as INITIAL_CODEEC_RATE for video. If both ECN/MIN_RATE/ABSOLUTE and ECN/MIN_RATE/RELATIVE are set, the larger of these two shall be used as the lower boundary for the media bit-rate adaptation in response to ECN-CE marking. |
| RTP_GAP (float) | If no RTP packets are received for longer than this period (proportion of the estimated frame period), the receiver should declare bursty packet loss or severe congestion condition. Packet loss gap can be detected as follows: based on the reception history of video packets and their time-stamps, the receiver keeps a running estimate of the frame period, T_FRAME_EST. If the receiver does not receive any RTP packets for a duration of RTP_GAP x T_FRAME_EST, then it should react accordingly. Typical RTP_GAP values can range from 0.5 to 5.0. |
| INC_FBACK_MIN_INTERVAL (ms) | Minimum interval between transmitting TMMBR messages that increase the maximum rate limit. |
| DEC_FBACK_MIN_INTERVAL (ms) | Minimum interval between transmitting TMMBR messages that decrease the maximum rate limit. |
| TP_DURATION_HI (ms) | Duration of sliding window over which the interval between packet arrival and playout is observed and computed. The computed value is compared with the TARGET_PLAYOUT_MARGIN_HI threshold. |
| TP_DURATION_MIN (ms) | Duration of sliding window over which the interval between packet arrival and playout is observed and computed. The computed value is compared with the TARGET_PLAYOUT_MARGIN_MIN threshold. |
| TARGET_PLAYOUT_MARGIN_HI (ms) | Upper threshold of the interval between packet arrival and its properly scheduled playout. The interval is measured from playout time to the X percentile point (X_PERCENTILE) of the packet arrival distribution. When this upper threshold is exceeded, the receiver may signal the sender to increase the bit rate. |
| TARGET_PLAYOUT_MARGIN_MIN (ms) | Lower threshold of the interval between packet arrival and its properly scheduled playout. The interval is measured from playout time to the X percentile point (X_PERCENTILE) of the packet arrival distribution. When this lower threshold is exceeded, the receiver should signal the sender to decrease the bit rate. |
| RAMP_UP_RATE (kbps/s) | Rate at which video encoder should increase its target bit rate to a higher max rate limit. |
| RAMP_DOWN_RATE (kbps/s) | Rate at which video encoder should decrease its target bit rate to a lower max rate limit. |

| | |
|-------------------------|--|
| DECONGEST_TIME (ms) | Minimum time the receiver should command the sender to spend in decongesting the transmission path, before attempting to transmit at the sustainable rate of the path. The receiver can achieve decongestion by first sending a TMMBR message with a value below the sustainable rate of the path. Once the receiver concludes that congestion has been cleared, it can send a TMMBR message with a value closer to the sustainable rate of the path. If the receiver concludes that congestion has not been cleared yet, it may attempt to clear the remaining congestion for another period of DECONGEST_TIME. A short DECONGEST_TIME results in a quick and aggressive decongestion by reducing the bit rate radically while a long DECONGEST_TIME results in a long and conservative decongestion. A value of 0 indicates that the receiver should not attempt to perform any decongestion at all. |
| HOLD_DROP_END (integer) | Tri-valued parameter that controls how the sender should behave in case video quality cannot meet the requirements set in BIT_RATE, FRAME_RATE, or QP. This parameter indicates whether the sender should put the video stream on hold while maintaining QoS reservations, drop the video stream and release QoS reservations, or end the session. Allowed values of this parameter are defined as follows: "0" = HOLD, "1" = DROP, "2" = END. |
| INITIAL_CODEC_RATE (%) | Initial bit rate (proportion of the bit rate negotiated for the video session) that the sender should begin encoding video at. |
| X_PERCENTILE (%) | X percentile point of the packet arrival distribution used with TARGET_PLAYOUT_MARGIN parameters. |

17.3 Management procedures

This clause explains how speech and video adaptation of the MTSI client in terminal can be managed using 3GPP MTSIMA MO and OMA-DM protocol. First, it is necessary to describe the expected behavior of media adaptation, i.e., reaction of the MTSI client in terminal to the received RTCP-APP and TMMBR messages, information on the transmission results such as RTCP RR and SR, signalled changes in transport characteristics such as ECN Congestion Experienced (ECN-CE) marking in IP packet headers, and analysis of packet reception status. Such descriptions, which include many parameters of different nature, can be made in the form of adaptation state machines or state transition tables, as in Annex C, based on the criteria for service quality or the policy for network management.

Some parameters in the descriptions can be determined in session setup or measured during session, and therefore do not require to be managed from outside. For example, the maximum or minimum bit rate of speech and video codecs, and round-trip time (RTT) belong to this class of parameters. It is also possible that other parameters are implementation-specific, or related to detailed features of media codec or underlying radio access bearer technology. These classes of parameters are not provided by 3GPP MTSIMA MO but still can be included under Ext nodes as vendor extensions.

The next step will be to select the parameters to be included in 3GPP MTSIMA MO. It might not be practical or necessary to update all parameters in the descriptions and selecting a subset of key parameters might simplify the management. The set of parameters selected should enable the behavior of media adaptation to be controlled up to the necessary extent.

The results of session setup may influence the selection of media adaptation methods to apply. For example, the negotiated media codec and the bandwidth, or whether to use ECN or not may determine the necessary adaptation procedures. Selection of session parameters from 3GPP MTSINP MO falls outside the scope of the present document. Information available to the MTSI client in terminal that may assist such decisions includes, but may not be limited to, the radio access bearer technology, information on service provider broadcast by (e)NodeB, date and time, and service policy.

17.3.1 Management of speech adaptation

3GPP MTSIMA MO contains a set of parameters which can be used in the construction of adaptation state machines. If available, information on the expected behavior of the network, such as the scheduling strategy applied to eNodeB, can assist the design and calibration process. Basically the receiver estimates the encoding and payload packetization status of the sender, and transmits appropriate RTCP-APP messages when the state of adaptation state machine needs to be switched.

Each PLR in table 17.1 is used to specify the conditions, usually as a threshold, to enter or exit a state. MAX, LOW, STATE_REVERSION, and RED_INEFFECTIVE correspond to PLR_1, PLR_2, PLR_3, and PLR_4 in Annex C respectively. Once the measured PLR exceeds or falls below the thresholds, while meeting certain conditions, adaptation state machine triggers the programmed transitions. A subset of PLRs can be used to construct adaptation state machines with fewer states. For example, the two-state adaptation state machine in Annex C can be built with MAX and LOW. DURATION_MAX, DURATION_LOW, DURATION_STATE_REVERSION, and DURATION_RED_INEFFECTIVE can be used to specify the duration of sliding window over which MAX, LOW, STATE_REVERSION, and RED_INEFFECTIVE PLR are observed and computed. DURATION is reserved for the case when it is not necessary to separately specify the durations. N_HOLD allows setting of the duration as an integer multiple of DURATION.

With each pair of a PLR and a DURATION, the observation period of each PLR can be controlled and the sensitivity of each transition path can be tailored to meet the requirements. For example, larger DURATION values are likely to smooth out the impact of bursty loss of packets and reduce the likelihood of frequent transitions between states, i.e., the ping-pong effects, but can delay the reaction to events that require immediate repairing actions. In general, transitions to states designed for better transmission conditions need to be taken more conservatively than transitions to states for worse transmission conditions. Other requirements can be combined with PLR to refine the conditions for transitions.

Packet loss burst (PLB) refers to a devastating event in which a large number of packets are lost during a limited period. Immediate measures, such as changing the bit rate or payload packetization are required to reduce the impact on the perceived speech quality. As PLR and PLR/DURATION enable detailed specification of PLR, PLB can be described efficiently with PLB/LOST_PACKET and PLB/DURATION.

The parameters ICM/INITIAL_CODEC_RATE, ICM/INIT_WAIT and ICM/INIT_UPSWITCH_WAIT can be used to control the rate adaptation during the beginning of the session. ICM/INITIAL_CODEC_RATE is used to define what codec mode should be used when starting the encoding for the RTP stream. In EVS Primary mode, ICM/INITIAL_CODEC_BANDWIDTH is used to define which audio bandwidth should be used when starting the encoding for the RTP stream. ICM/INIT_WAIT defines the period over which the sending MTSI client in terminal should use the Initial Codec Mode when ECN is not used. If no codec mode request or other feedback information is received within this period then the sender is allowed to adapt to a higher rate. Since it is unknown in the beginning of the RTP stream whether the transmission path can support higher rates, the adaptation to higher bit rates needs to be conservative. It is therefore recommended that when adapting to a higher rate the sender increases the rate only to the next higher rate in the list of codec modes allowed in the session. It is also recommended that the sender waits for a while in-between consecutive up-switches, to give the receiver a chance to evaluate whether the new rate can be sustained. This waiting period in-between consecutive up-switches can be controlled with the ICM/INIT_UPSWITCH_WAIT parameter when ECN is not used. For the channel aware mode of EVS Primary, ICM/INIT_PARTIAL_REDUNDANCY_OFFSET_SEND and INIT_PARTIAL_REDUNDANCY_OFFSET_RECV can be used to configure the initial redundancy offset for the send and the receive directions respectively.

When ECN is used in the session, the ECN/INIT_WAIT and ECN/INIT_UPSWITCH_WAIT parameters are used instead of the ICM/INIT_WAIT and ICM/INIT_UPSWITCH_WAIT parameters, respectively.

N_INHIBIT can be used to limit the earliest time for the next transition, after transition is temporarily disabled due to frequent transitions among a limited number of states. Use of N_INHIBIT is suggested as a measure to avoid unnecessary transitions during rapid fluctuations of transmission conditions. It is left as the discretion of the implementation to handle RTCP-APP messages received before the sender is allowed to transition again.

T_RESPONSE refers to the maximum period the receiver can tolerate, before declaring that either the transmitted RTCP-APP message was lost or its execution was denied by the sender. After the timer expires, the receiver may retransmit the request or transmit a new request, or choose to be satisfied with current status.

Adaptation state machines using above parameters collect the information on transmission path by analysing the packet reception process. Another, more direct source of information can be provided by network nodes, such as eNodeB, in the form of Explicit Congestion Notification (ECN) to IP. A key benefit of ECN is more refined initiation of adaptation in which the receiver can be aware of incoming deterioration of transmission conditions even before any packets are dropped by network node, i.e., as an early-warning scheme for congestion.

STEPWISE_DOWNSWITCH can be used to control the path of bit-rate reduction, i.e., whether to directly down-switch to ECN/MIN_RATE or to gradually down-switch via several intermediate bit-rates specified in ECN/RATE_LIST. The former path may be preferred when rapid reduction of the bit-rate is required while the latter path may be employed for more graceful degradation of speech quality.

To avoid premature up-switch before the congestion has been cleared, waiting periods during which the sender is not allowed to increase the bit-rate can be defined with ECN/CONGESTION_WAIT parameter. The

ECN/CONGESTION_UPSWITCH_WAIT parameter is used to prevent congestion from re-occurring during the upswitch after the ECN/CONGESTION_WAIT period.

To align speech adaptation of the MTSI client in terminal with the purpose of quality control or network management, not only the terminals, which might be managed by different service providers, but also the behaviour, such as scheduling strategy or ECN-marking policy, of network nodes should be considered in the construction of adaptation state machines. It is also possible to program the terminals to adapt differently, as a means of differentiating the quality of service.

With 3GPP MTSIMA MO, it is possible to shape a rough trajectory of the bit rate over time-varying transmission conditions but the maximum and minimum bit rates of speech codec are determined during session setup with mode-set, which can be managed with RateSet leaf of 3GPP MTSINP MO (see clause 15).

Adaptation state machines designed to recover the once reduced bit or packet rate at an earliest opportunity might be considered as an adaptation policy oriented to service quality. However, such an aggressive up-switch before the transmission conditions fully recover takes the risk of degrading the quality or even backward transitions, i.e., the ping-pong effects. Such an optimistic adaptation strategy might not necessarily result in higher quality but can influence the service quality of other terminals sharing the same link. On the other hand, adaptation state machines that increase the once reduced bit or packet rate more conservatively are likely to avoid such situations but might be late in the recovery of speech quality after the transmission conditions are restored.

Even at similar total bit rates, bit stream consisting of a smaller number of larger packets can be at a disadvantage during transmission over packet networks or shared links, when the link quality deteriorates or the link becomes congested, than bit stream consisting of a larger number of smaller packets, since many types of schedulers installed in the network nodes base their decisions on the size of packets such that lower priorities are assigned to larger packets. RTCP_APP_REQ_RED, RTCP_APP_REQ_AGG, and RTCP_APP_CMR specify detailed request for the bit rate and packetization. Bit-fields of RTCP_APP_REQ_RED and RTCP_APP_REQ_AGG are restricted by parameters, such as max-red and maxptime, which are negotiated during session setup.

17.3.2 Management of video adaptation

Compared with speech adaptation where the number of allowed bit rates from speech encoder is limited and each encoded speech frame covers the same short period, e.g., 20 ms, or contains the same number of bits when voice activity is present, video adaptation should tolerate a higher level of uncertainty in the control of the bit rate. Moreover, due to the structural dependence between encoded video frames, from motion estimation and compensation, packetization is not likely to be used as an opportunity for adaptation. This dependence necessitates not only controlling the bit rate but also putting an end to error propagation with AVPF NACK or PLI.

Output bit rate from video encoder depends also on the scene being encoded and even if maintaining a constant bit rate is intended, actual output bit rate is likely to fluctuate around a target value. In the design of adaptation state machines for video, this uncertainty needs to be compensated for, for example, with additional implementation margin.

Encoded speech frames have a clear boundary in the bit stream and multiple speech frames can be transported over an RTP packet. In contrast, an encoded video frame, whose size depends on the bit rate, frame rate, and image size, is typically far larger than an encoded speech frame. Multiple packets are usually necessary to transport even a predicted frame, which is usually smaller than an intra frame.

As in speech adaptation, basic information on transmission path can be obtained from analyzing received packet stream. However, perceived video quality can be more sensitive to PLR since the compression ratio of video is typically higher than that of speech and even a minor level of packet loss can initiate error propagation to the following predicted frames, rendering them unrecognizable. For example, at comparable PLR values, speech quality can be acceptable but video quality can be significantly damaged such that dropping the media might be considered. Two parameters for PLR, MAX and LOW, and two additional parameters for their durations, DURATION_MAX and DURATION_LOW, are available for video adaptation.

PLB/LOST_PACKET and PLB/DURATION are also available for video but the fundamental differences in the frame structure need to be taken into account when the event of packet loss burst is defined for video.

INC_FBACK_MIN_INTERVAL and DEC_FBACK_MIN_INTERVAL can be used to control the rate of adaptation and also the amount of signalling overhead. These two minimum intervals are provided separately since the minimum interval between the feedback messages to decrease the bit rate typically needs to be shorter than the one to increase the bit rate. The urgency of rate-decreasing conditions generally requires shorter minimum feedback intervals.

Target bit rate for video is determined during session setup and can be considered as the maximum bit rate to be used during session, which can be configured with the Bandwidth leaf of 3GPP MTSINP MO. On the other hand, BIT_RATE can be used to set a lower threshold for the bit rate. Whether MIN_QUALITY/BIT_RATE/ABSOLUTE or MIN_QUALITY/BIT_RATE/RELATIVE is to be used is left as the discretion of the implementation or service provider. For example, capability of setting a fixed minimum bit rate can be necessary when the lowest quality of MTSI is required to be comparable to the quality of 3G-324M, whose bit rate for video is in general set to 47 ~ 49 kbps. If both MIN_QUALITY/BIT_RATE/ABSOLUTE and MIN_QUALITY/BIT_RATE/RELATIVE are set, the larger of these two shall be used as the minimum bit rate.

In the case of speech adaptation, the MTSI client in terminal limits the initial codec mode (ICM) to a lower mode than the maximum mode negotiated, until at least one frame block or an RTCP message is received with rate control information (see clause 7.5). This policy is recommended to avoid congestion during initial phase of session when the information on transmission path is known to neither the sender nor the receiver. INITIAL_CODEC_RATE can be used for video with similar objectives as that of ICM, i.e., a warming-up process in the beginning of session. Once the session starts and few packets are lost during delivery, the receiver will attempt to increase the bit rate by transmitting TMMBR messages requesting higher bit rates until the negotiated value is reached. However, low INITIAL_CODEC_RATE can reduce the video quality at session setup when the transmission path is free of congestion.

The maximum bit rate allowed for video communication in a session depends on the outcome of the SDP offer-answer negotiation. For inter-working with 3G-324M it is likely that the bit rate is limited to 47 ~ 49 kbps while for high-quality video communication it is foreseen that bit rates in the order of several hundred kbps might be used. This can be challenging when setting the ECN/MIN_RATE parameter since the configuration of the MTSI client in terminal parameters occurs rarely while the maximum allowed bit rate used for video may vary from session to session.

Two parameters, ECN/MIN_RATE/ABSOLUTE and ECN/MIN_RATE/RELATIVE, are therefore provided to enable better control of the video rate adaptation algorithm. The ECN/MIN_RATE/RELATIVE parameter is provided to limit the bit range variations during a session to avoid large quality variations. The ECN/MIN_RATE/ABSOLUTE parameter is provided to avoid reducing the bit rate to an unacceptably low quality level.

FRAME_RATE can be used to set a lower threshold for the frame rate. As the bit rate is controlled during adaptation between two limits, the frame rate also needs to be controlled between the limits while maintaining a balance between spatial quality and temporal resolution (see clause 10.3). As the increase in codec profile/level can result in an abrupt increase of the maximum image size, e.g., from QCIF to CIF, so can quadruple the maximum frame rate, with a fixed image size. With "imageattr" attribute, it is possible that image sizes whose maximum frame rates are unspecified by codec profile/level, such as 272x224, can be negotiated (see clause A.4). In this case, the maximum frame rate is determined as the maximum value at the maximum image size supported by the profile/level negotiated. Whether MIN_QUALITY/FRAME_RATE/ABSOLUTE or MIN_QUALITY/FRAME_RATE/RELATIVE is used to specify the lower threshold of the frame rate is left as the discretion of the implementation or service provider. If both are set, the larger of these two shall be used as the minimum frame rate.

RTP_GAP can be used to set the maximum interval between received packets before the receiver considers repairing actions. During periods of severe congestion or packet loss, the receiver may not receive packets for an unexpectedly long period. Observing such gaps in the reception of packets can be used by the receiver to request the sender to decrease the bit or packet rate. In the case of severe packet loss, this gap can be detected before any other observations are made and thus allows for faster reaction, while detection of packet loss requires reception of at least one packet after the loss.

However, estimating such gaps in the arrival of packets can be challenging because video encoder may not always output packets at regular intervals and typical scheduling strategy of network node, especially in the downlink, can cause jitter in the delivery of video packets. Therefore, it is recommended that RTP_GAP is set conservatively and the measured gap is based on a moving average estimate of the frame period observed by the receiver. The timestamps of the received packets allow the receiver to estimate the frame period based on the past a few received video frames. Since typical video encoders are not likely to abruptly change their encoding frame rates, this estimate can serve as a fairly reliable basis for detecting the gaps in the transport of video packets.

Leaf nodes for luminance quantization parameter, H263, MPEG4, and H264, can be used to set a lower threshold for the image clarity to be maintained. Target range of the quantization parameters depends on the video codec negotiated.

If the MTSI client in terminal cannot maintain the bit rate or the frame rate higher than the lower thresholds, or cannot maintain the quantization parameter lower than the higher threshold, the video stream might be put on hold, dropped, or the session might be ended, depending on the criteria for service quality or policy for network management, with HOLD_DROP_END.

RAMP_UP_RATE and RAMP_DOWN_RATE can be used to control how fast the sender changes its target bit rate from its current target value to the value indicated in the most recently received TMMBR message, when the bit rate is being increased and decreased respectively. As with INC_FBACK_MIN_INTERVAL and DEC_FBACK_MIN_INTERVAL, rates for ramping up and down need to be different, as rapid ramping down is usually necessary whereas rapid ramping up is undesirable as it can cause sudden congestion in the transmission path.

DECONGEST_TIME can be used to control the time spent in resolving the congestion of transmission path. Smaller values of this parameter can result in faster reduction of the bit rate while larger values can be used for slower decongestion. If the situation at the receiver does not improve at the end of initial decongesting, another round of decongestion can be attempted, or the video stream can be dropped or put on hold.

From received packets, video frames are typically reconstructed to YUV format, converted to formats such as RGB, and stored in the frame buffer, before being fed to the display for visual presentation. TARGET_PLAYOUT_MARGIN_HI and TARGET_PLAYOUT_MARGIN_MIN can be used to maintain appropriate playout margin, defined as the interval between packet arrival and its properly scheduled playout. Duration of sliding window over which the interval is observed and computed can be controlled with TP_DURATION_HI and TP_DURATION_MIN.

In general, video should be encoded, packetized, transmitted, de-packetized, decoded, and, played out within a total delay target. In addition, processing of video should be appropriately synchronized to that of speech. If the estimated playout margin exceeds TARGET_PLAYOUT_MARGIN_HI, it is considered that video packets are arriving too early and there remains room for higher bit rate in the transmission path. Therefore the receiver may signal the sender to increase the bit rate with TMMBR messages. If the estimated playout margin falls below TARGET_PLAYOUT_MARGIN_MIN, it is considered that video packets are arriving too late and current transmission path cannot sustain the bit rate. Therefore the receiver should signal the sender to reduce the bit rate to enable earlier arrival of video packets.

X_PERCENTILE can be used to control the target playout margin but the packet arrival distribution is left to the discretion of the implementation, which might be implemented as statistical models or empirical data.

17.4 Management of media robustness adaptation

17.4.1 General

The MEDIA_ROBUSTNESS node defined under the 3GPP MTSIMA MO may be used to manage media robustness adaptation across different vendor terminals in a network. For each codec type, the MO node provides a list of codec configurations arranged from least robust to most robust. For each of these configurations, with the exception the first and last, the MO node also provides two sets of PLR threshold levels (see Figure 17.2):

- A set of high PLR thresholds that trigger the media receiver to request a more robust configuration from the media sender when the PLR is increasing in order to reduce the effects of the higher PLR on QoE, and
- A set of low PLR thresholds that trigger the media receiver to request a less robust configuration from the media sender when the PLR is decreasing in order to take advantage of the improved media quality supported at the lower PLR.

The high PLR and low PLR thresholds between two codec configurations can be set independently to avoid ping-pong switching by introducing hysteresis. The least robust configuration does not have a low PLR threshold and the most robust configuration does not have a high PLR threshold. The MO node does not describe the type of request message that shall be used for adapting the codec configuration as this is determined by the codec configuration being requested, i.e., in-band RTP CMR if requesting a speech codec mode change, RTCP-APP if it requesting speech application layer redundancy change, TMMBR for video.

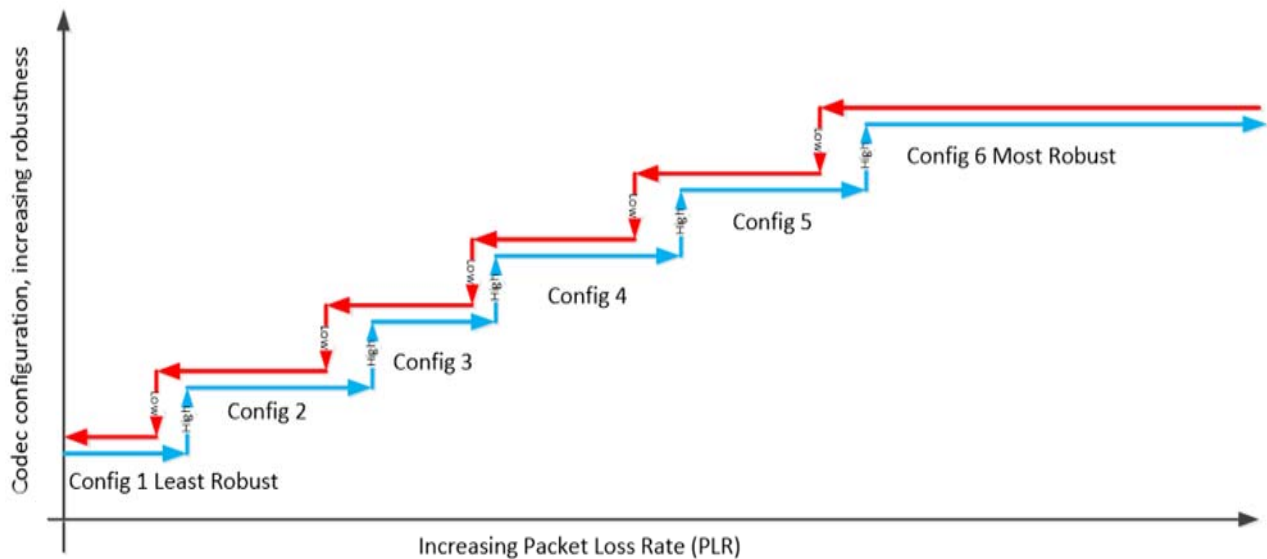


Figure 17.2: High and Low PLR thresholds for media robustness adaptation

The MO node also provides a flag to indicate whether PLR measurements are to be made before or after the de-jitter buffer in the receiver. Measuring before provides an estimate of the PLR over the transport link to the media receiver while measuring after provides a PLR estimate that is closer to the QoE after the media decoder. The MO node also specifies the sliding averaging window over which PLR estimates are to be calculated.

The parameters are specified independent of the media codec or radio access bearer technology. In addition, vendor specific parameters of the implementation can be placed under Ext nodes. Detailed descriptions of the speech robustness adaptation parameters can be found in table 17.1.

18 MTSI client in terminal using fixed access

18.1 General

This clause 18 applies to an MTSI client in terminal using fixed access.

The functional components of an MTSI client in terminal using fixed access are the same as described in clause 4.2 except that another Layer 2 technology may be used instead of the 3GPP L2 data link.

18.2 Media codecs

18.2.1 General

Media codecs for speech and video are specified in TS 181 005 [98]. Additional requirements and recommendations are included below.

18.2.2 Speech

18.2.2.1 Speech codecs

MTSI clients in terminal using fixed access supporting AMR, AMR-WB or EVS shall follow clause 5.2.1.

An MTSI client in terminal using fixed access supporting G.711 [77] shall support either A-law PCM or μ -law PCM and should support both.

MTSI client in terminal using fixed access supporting G.722 shall use the mode operation 1 at 64 kbps as specified in ITU-T Recommendation G.722 [78] when G.722 is used. The bitstream ordering shall be in chronological order with Most Significant Bit (MSB) first.

MTSI client in terminal using fixed access supporting EVRC, EVRC-B, and /or EVRC-WB shall follow 3GPP2 C.S0014-E v1.0 [99] when any of these codecs are used.

Encoding of DTMF is described in Annex G.

18.2.2.2 Error concealment procedures

Error concealment procedures shall be used to reduce the quality degradation of the reconstructed speech when one or more erroneous/lost speech or lost Silence Descriptor (SID) frames are received.

For G.722, it is recommended to use Appendix III or Appendix IV of ITU-T Recommendation G.722 [78].

NOTE: Appendices III and IV meet the same quality requirements but with two different quality/complexity trade-offs:

- Appendix III of ITU-T Recommendation G.722 [78] aims at maximizing the robustness at a price of additional complexity.
- Appendix IV of ITU-T Recommendation G.722 [78] offers an optimized complexity/quality trade off with almost no additional complexity compared with G.722 normal decoding (+0.07 WMOPS).

If another error concealment procedure is used it shall have equivalent or better performance than Appendix III or Appendix IV.

For G.711, it is recommended to use Appendix I of ITU-T Recommendation G.711 [77]. If another error concealment procedure is used, it shall have equivalent or better performance than Appendix I of ITU-T Recommendation G.711.

For G.729, the error concealment procedure shall be used as specified in the Main Body of ITU-T Recommendation G.729 [100].

For G.729.1, the error concealment procedure shall be used as specified in the ITU-T Recommendation G.729.1 [101].

18.2.2.3 Source controlled rate operation

An MTSI client in terminal using fixed access supporting AMR, AMR-WB or EVS shall support source controlled rate operation in accordance with clause 5.2.1.

For an MTSI client in terminal using fixed access supporting other codecs than AMR, AMR-WB or EVS the following recommendations apply:

- Source controlled rate operation for G.729 should be supported according to Annex B of ITU-T G.729 [100].
- Source controlled rate operation for G.729.1 should be supported according to Annex C and Annex F of ITU-T G.729.1 [101]. Annex C specifies a discontinuous transmission (DTX) and comfort noise generation for G.729.1. Annex F specified the voice activity detector (VAD) to be used together with the DTX/CNG scheme of Annex C to provide the complete functionality of the discontinuous transmission system.
- Source controlled rate operation for G.711 should be supported according to Appendix 2 of ITU-T G.711 [77].
- No source controlled rate operation has been standardized for G.722.

NOTE 1: Use of source controlled rate operation is optional. Source controlled rate operation is known to degrade the speech quality, especially in noisy environments or with background music, and is not needed when both MTSI client in terminals are using fixed access and when the bandwidth is sufficient to ensure best possible voice quality.

NOTE 2: Apart from source controlled rate operation (VAD/DTX) specified in clause 4.19 of 3GPP2 C.S0014-E [99] and in 3GPP2 C.S0076 v1.0 [102], EVRC, EVRC-B, and EVRC-WB can dynamically vary the source coding bit-rate for active speech to achieve a targeted active speech average data rate as specified in 3GPP2 C.S0014-E.

18.2.3 Video

MTSI clients in terminals using fixed access offering video communication shall support the video codecs as defined in clause 5.2.2.

NOTE: The video codecs recommended in TS 181 005 [98] are H.263 profile 0 and H.264 Baseline Profile without any constraint on the levels. For 3GPP MTSI clients in terminal using 3GPP access, only H.264 is specified in the present document but with a different profile: the Constrained Baseline Profile (CBP) for which the Level 1.2 is mandatory and Level 3.1 is recommended.

The Baseline Profile and the Constrained Baseline Profile are very close but not compatible. The Baseline Profile includes all features that are supported in the Constrained Baseline Profile but some limited features of the Baseline Profile are not supported in the Constrained Baseline Profile.

18.2.4 Real-time text

An MTSI client in terminal using fixed access and offering real-time text shall support real-time text as defined in clause 5.2.3.

18.3 Media configuration

18.3.1 General

The general clause on media configuration (clause 6.1) applies to an MTSI client in terminal using fixed access.

An MTSI client in terminal using fixed access and supporting RTP/AVPF shall do RTP profile negotiation as defined in clause 6.2.1a.

The support for Explicit Congestion Notification (ECN) is optional for an MTSI client in terminal using fixed access. ECN may be used to perform rate adaptation for speech and video when at least one multi-rate or rate-adaptive codec is supported. If ECN is supported then this shall be done in accordance with the requirements and recommendations specified in clause 6 and in clause 7.3 for RTCP based adaptation.

NOTE: It is beneficial if the MTSI client in terminal using fixed access supports ECN, even if the fixed network is not expected to support or use ECN for RTP. This enables using ECN between fixed and mobile clients when the same codecs are supported end-to-end.

An MTSI client in terminal using fixed access and supporting ECN should negotiate ECN usage when the SDP offer includes at least one multi-rate or rate-adaptive codec, see clause 6.2.2 for speech and clause 6.2.3.2 for video. If only fixed-rate codecs are included in the SDP offer for a media type then ECN shall not be negotiated for that media type.

An MTSI client in terminal using fixed access and supporting ECN may accept using ECN when a multi-rate or rate-adaptive codec is accepted, see clause 6.2.2 for speech and clause 6.2.3.2 for video.

18.3.2 Session setup procedures

18.3.2.1 General

The general clause on session set up procedures (clause 6.2.1) applies to the MTSI client in terminal using fixed access.

If an MTSI client in terminal using fixed access supports AVPF for a media type then it shall also support the complete SDPCapNeg framework, RFC 5939 [69], for that media type in order to negotiate the RTP profiles.

18.3.2.2 Speech

If an MTSI client in terminal using fixed access supports AMR and/or AMR-WB and/or EVS, then clause 6.2.2 applies for session set up.

An MTSI client in terminal using fixed access shall support RTP/AVP. When at least one multi-rate codec is supported (AMR, AMR-WB, EVS or G.729.1) then RTP/AVPF should be supported to allow for end-to-end rate adaptation.

If an MTSI client in terminal using fixed access supports AMR and/or AMR-WB, or EVS, then clause 6.2.2.2 applies for generating SDP offers for AMR-NB, AMR-WB and EVS.

An MTSI client in terminal using fixed access supporting both A-law PCM and μ -law PCM shall offer both variants when sending an SDP offer for G.711.

When an MTSI client in terminal using fixed access supports EVRC-B or EVRC-WB, then clauses 14-18 of RFC 5188 [103] apply when generating SDP offers and answers for EVRC-B and EVRC-WB.

If an MTSI client in terminal using fixed access supports G.729.1 then it also supports G.729 and should offer both G.729.1 and G.729 when sending an SDP offer.

An MTSI client in terminal offering G.729 with source controlled rate operation shall use the parameter "annexb" according to RFC 4855 [107].

The following codec preference order applies for the SDP offer in the session negotiation:

- If AMR-WB is offered it shall be listed first in the codec list (in order of preference, the first codec being preferred).
- If both narrowband codecs and wideband codecs are offered, wideband codecs shall be listed first in the codec list.

When sending the SDP answer, if a wide-band speech session is possible, then selection of narrow-band speech should be avoided whenever possible, unless another preference order is indicated in the SDP offer.

Session setup for sessions including speech and DTMF events is described in Annex G.

18.3.2.3 Video

An MTSI client in terminal using fixed access and supporting video shall support RTP/AVP and shall support RTP/AVPF.

If an MTSI client in terminal using fixed access supports video, then clause 6.2.3 applies for the video session set up.

18.3.2.4 Text

An MTSI client in terminal using fixed access and offering text shall follow clause 6.2.4.

18.3.2.5 Bandwidth negotiation

The general clause 6.2.5.1 related to the use of Application Specific (AS) bandwidth modifier applies also to the MTSI client in terminal using fixed access.

If an MTSI client in terminal using fixed access supports AMR and/or AMR-WB and/or EVS, then clause 6.2.5.2 applies for the bandwidth negotiation for these codecs.

If an MTSI client in terminal using fixed access supports video, then clause 6.2.5.3 applies for the bandwidth negotiation.

When the SDP offer includes multiple codecs then the bandwidth indicated with $b=AS$ shall be set based on the codec that requires the highest bandwidth.

An MTSI client in terminal using fixed access should include the 'a=bw-info' attribute (see clause 19) in the SDP offer for speech and video media types and should include the attribute for other media types, as described in clause 6.2.5.1. For AMR, AMR-WB or EVS, the setting of the bandwidth properties are defined in clause 6.2.5.2 and Annex K.

18.3.3 Session control procedures

The clause 6.3 on session set up procedures applies also to an MTSI client in terminal that uses fixed access.

18.4 Data transport

18.4.1 General

The clauses data transport general (7.1) and RTCP usage (7.3.1) apply also to the MTSI client in terminal using fixed access.

An MTSI client in terminal using fixed access shall transport real-time media using RTP (RFC 3550 [9]) over UDP (RFC 0768 [39]). See clause 18.3.2.2, 18.3.2.3 and 18.3.2.4 for requirements on RTP/AVP and RTP/AVPF for speech, video and text, respectively.

The support of AVPF requires an MTSI client in terminal to implement the RTCP transmission rules, the signalling mechanism for SDP and the feedback messages explicitly mentioned in the present document.

For a given RTP based media stream, the MTSI client in terminal shall use the same port number for sending and receiving RTP packets. This facilitates interworking with fixed/broadband access. However, the MTSI client shall accept RTP packets that are not received from the same remote port where RTP packets are sent by the MTSI client.

18.4.2 Packetization

For G.711 both 10 ms and 20 ms frame length shall be supported, and 20 ms frame length shall be used unless another packetization is negotiated. The terminal shall offer to receive 20 ms frame length packetization.

For G.722, 20 ms frame length shall be supported.

The default packetization time for the codecs used in the MTSI client in terminal using fixed access shall be 20 ms (1 non-redundant speech frame per RTP packet). The packetization could change as a result of adaptation by using frame aggregation when adapting to packet rate limited operating conditions, see also clause 18.7.

Packetization time shall be indicated using the `a=ptime` SDP attribute.

An MTSI clients in terminal using fixed access shall support encapsulating up to 4 non-redundant speech frames into the RTP packets and 12 speech frames in total if redundancy is used (`maxptime = 240`).

18.4.3 RTP payload format

For each of the following codecs the payload formats are defined in:

- For AMR and/or AMR-WB, or EVS as specified in clause 7.4.2.
- For G.729.1 as specified in RFC 4749 [104], and as specified in RFC 5459 [105] when DTX is used.
- For EVRC and EVRC-B as specified in RFC 4788 [106].
- For EVRC-WB as specified in RFC 5188 [103].
- For DTMF events is described in Annex G.

The RTP payload types for G.711, G.729, G.722 shall be supported as specified in Section 6, Table 4 of RFC 3551 [10].

The following payload types should be used for G.711, G.729 and G.722:

Table 18.4.3-1: Recommended payload type numbers

| Codec | Payload Type |
|------------------|--------------|
| G.711 A-law | PT= 8 |
| G.711 μ -law | PT= 0 |
| G.729 | PT= 18 |
| G.722 | PT=9 |

For other codecs, dynamic payload types shall be used.

18.5 Jitter buffer management

An MTSI client in terminal using fixed access shall be able to handle delay jitter in order to minimize the speech quality degradation due to jitter (jitter induced frame losses) while limiting the additional end to end delay due to jitter buffering time.

The jitter buffer management (JBM) shall comply with the functional requirements defined in clause 8.2.2 and with the minimum performance requirements defined in clause 8.2.3.

NOTE: The delay and error profiles defined in clause 8.2.3.2.3 were derived for mobile access but they were selected to span the different jitter and packet loss conditions that may occur in different types of networks and in different combination of networks. For fixed-mobile interworking the jitter may be small and packet loss rate may be low in the fixed part of the path, but the jitter may be large and the packet loss rate may be fairly high in the mobile part of the path. This can give end-to-end jitter and packet loss characteristics that are similar to the characteristics found in these profiles. The JBM therefore needs to handle these profiles to support end-to-end fixed-mobile interworking.

18.6 Packet-loss handling

An MTSI client in terminal using fixed access may use redundancy to handle operating conditions with high packet loss rates. When redundancy is supported then this shall be done in accordance with the requirements and recommendations defined in clause 9.

When redundancy is used then the multi-rate or rate-adaptive capabilities of the codec should be used to avoid increasing the load in the network. Redundancy shall be used for fixed-rate codecs only when permitted by the allocated bandwidth.

NOTE: It is beneficial if the MTSI client in terminal using fixed access supports redundancy since this enables using such solutions end-to-end between fixed and mobile clients, at least when the same codecs are supported end-to-end.

18.7 Adaptation

An MTSI client in terminal using fixed access supporting AMR and/or AMR-WB for speech or supporting video should support adaptation as described in clause 10 for speech and video, respectively.

Frame aggregation and redundancy should be supported also for fixed-rate codecs.

The media bit rate adaptation for G.729.1 should use the same principles as described for AMR and AMR-WB in clause 10.2. For example, if G.729.1 is used at a bit rates up to 32 kbps, the adaptation may be configured to reduce the media bit-rate to 8 kbps when ECN-CE is detected.

18.8 Front-end handling

For terminals only supporting fixed access, performance requirements for terminal acoustics and test methods are specified in ETSI TS 103 737 [109], ETSI TS 103 738 [110], ETSI TS 103 739 [111], ETSI TS 103 740 [112], ETSI ES 202 737 [113], ETSI ES 202 738 [114], ETSI ES 202 739 [115], ETSI ES 202 740 [116] and for DECT terminals in ETSI specifications EN 300 175-8 [117] and EN 300 176-2 [118].

Other terminals supporting fixed and 3GPP access shall meet or exceed the minimum performance requirements on the acoustic characteristics of 3G terminals specified in TS 26.131 [35] in order to harmonize the acoustic front-end.

18.9 Supplementary services

An MTSI client in terminal using fixed access shall support media handling for supplementary services as defined in clause 14, except that the codecs are defined in clause 18.2 and the data transport is defined in clause 18.4.

19 Additional bandwidth information

19.1 General

This clause describes additional bandwidth properties that can be used when the existing bandwidth modifiers (b=AS, etc) give insufficient information. The additional information can be used to, for example (but not limited to): align the resource allocation end-to-end; align the bearer setup in different networks; or to assist how the MTSI clients should adapt in case of degraded operating conditions. The bandwidth properties are defined in clause 19.2.

A new SDP attribute 'a=bw-info' is defined in clause 19.3 and can be used to negotiate the additional bandwidth properties end-to-end. The SDP attribute allows for future extensions.

The IANA registration information on the new SDP attribute is provided in Annex M.6.

19.2 Bandwidth properties

19.2.1 General description

This clause defines in total seven bandwidth properties, which can be used both for sending and receiving direction. When the Maximum Supported Bandwidth property is defined for the receiving direction then this is semantically very similar to the b=AS parameter.

Four bandwidth properties are defined to describe different transport bandwidths:

- Maximum Supported Bandwidth
- Maximum Desired Bandwidth
- Minimum Desired Bandwidth
- Minimum Supported Bandwidth

These bandwidth properties shall include the IP, UDP and RTP overhead.

NOTE 1: Corresponding bandwidth parameters excluding IP, UDP and RTP overhead can be defined in the future but are not included here.

Since the above bandwidth properties include the IP, UDP and RTP overhead, the following bandwidth properties are defined to assist the re-calculation of the above bandwidth properties, e.g. when a MGW does conversion between different IP versions and therefore need to re-calculate these values:

- IP version
- Maximum packet rate
- Minimum packet rate

Detailed descriptions for these bandwidth properties are given in the following sub-sections.

The motivations for defining these properties are:

- The best compromise between quality and network utilization is reached when the media uses bandwidths from the Minimum Desired Bandwidth up to the Maximum Desired Bandwidth, which is therefore the most preferred bandwidth range. The higher end of this range should preferably be used for most sessions. When bitrate adaptation is needed due to degraded operating conditions, it may require changing the bandwidth down towards the Minimum Desired Bandwidth.
- Bandwidths below the Minimum Desired Bandwidth, down to the Minimum Supported Bandwidth, may be used during poor operating conditions, although this should happen rarely. If the operating conditions are so poor that not even media using Minimum Supported Bandwidth can be maintained then the media may be stopped or the session may be closed.

- Bandwidths above the Maximum Desired Bandwidth, up to the Maximum Supported Bandwidth, can be used to provide room for redundancy so that the media may survive during very bad operating conditions and when bandwidth reduction alone is unable to provide sufficient quality. This range should be used rarely.

NOTE 2: Increasing the bandwidth during bad operating conditions is an exception. Normally, the end-points ought to reduce the bandwidth. However, there are cases when it is not possible to reduce the bandwidth. For example, when AMR 4.75 kbps is used then the encoding bitrate cannot be reduced any further. It is then possible to use frame aggregation to reduce the IP/UDP/RTP overhead. However, if the limitation is in the RAN, where ROHC is used, then this will have limited effect, possibly even no effect at all. In this case, the only remaining option to improve the probability that speech reaches the receiving end-point is to allow the end-points to use redundancy, even if this means using a higher bandwidth. Otherwise, it is likely that the session will be terminated prematurely. This specification therefore allows for using up to 300% redundancy, as described in clause 9.2.1.

This means that the following relationships apply:

- Minimum Supported Bandwidth \leq Minimum Desired Bandwidth
- Minimum Desired Bandwidth \leq Maximum Desired Bandwidth
- Maximum Desired Bandwidth \leq Maximum Supported Bandwidth

19.2.2 Maximum Supported Bandwidth

Definition:

Identifies the highest bandwidth that can be used in the session during any operating conditions (including redundancy).

Should be used to set MBR.

Should also be used to set GBR for MBR=GBR bearers.

The unit for this bandwidth property shall be kbps.

Usage during the session:

The additional bandwidth for redundancy should only be used if adapting the bitrate to lower values, down to the Minimum Supported Bandwidth, fails to provide sufficient quality.

Quality aspects:

When additional bandwidth is allocated for redundancy, the resilience against losses should be improved. It should however be expected that the end-to-end delay will be longer than for the normal operating range.

19.2.3 Maximum Desired Bandwidth

Definition:

Identifies the highest bandwidth that should be used in most cases during normal operating conditions. This normally corresponds to the maximum bitrate allowed for the encoding.

The unit for this bandwidth property shall be kbps.

Usage during the session:

The adaptation should ensure that bandwidths up to the Maximum Desired Bandwidth are used whenever possible.

Quality aspects:

Using bandwidths in the higher end of the Minimum Desired Bandwidth ~ Maximum Desired Bandwidth range should give the intended encoding quality and end-to-end delay.

19.2.4 Minimum Desired Bandwidth

Definition:

Identifies the lowest bandwidth that should be used in the session during relatively normal or slightly degraded operating conditions.

Used for setting GBR for MBR>GBR bearers.

The unit for this bandwidth property shall be kbps.

Usage during the session:

Bandwidths in the lower end of the Minimum Desired Bandwidth ~ Maximum Desired Bandwidth should be used less frequently, mainly during periods with high load and/or degraded operating conditions.

The bandwidth used by media can be lower than the Minimum Desired Bandwidth, for example during DTX periods or lower rate operation of Source-Controlled Variable Bit Rate modes, e.g. EVS 5.9 VBR, for speech or when DTMF is being transmitted instead of speech.

Quality aspects:

Using bandwidths in the lower end of this range can give slightly reduced encoding quality but should not give increased end-to-end delay.

For video, this bandwidth should be selected such that the video is still relatively smooth.

19.2.5 Minimum Supported Bandwidth

Definition:

Identifies the lowest bandwidth that may be used in the session during exceptional operating conditions.

The unit for this bandwidth property shall be kbps.

Usage during the session:

Bandwidths below the Minimum Desired Bandwidth, down to the Minimum Supported Bandwidth, should be used quite rarely, mainly for severely degraded operating conditions.

If the operating conditions are so poor that not even Minimum Supported Bandwidth can be maintained then the session can be terminated.

The bandwidth used by media can deliberately be lower than the Minimum Supported Bandwidth, for example during DTX periods or lower rate operation of Source-Controlled Variable Bit Rate modes, e.g. EVS 5.9 VBR, for speech or when DTMF is being transmitted instead of speech. This is not to be considered a violation of the bandwidth parameter.

Quality aspects:

It can be expected that the encoding quality is reduced for these bandwidths and also that the end-to-end delay is increased.

19.2.6 IP version

Definition:

Identifies the IP version used for the calculation of the bandwidth properties.

This bandwidth property shall have the numerical values 4 or 6.

Usage during the session:

It may happen that gateways performing IPv4-IPv6 conversion re-calculate the bandwidth modifiers, e.g. b=AS, while leaving the bandwidths indicated with the 'a=bw-info' attribute unchanged. It is therefore beneficial to indicate which IP version that was assumed when the bandwidth properties were calculated.

The bandwidth properties may be calculated either using IPv4 and/or IPv6. The SDP may include bandwidth properties for both IPv4 and IPv6 in which case different attribute lines are used for the different IP versions and the IP version needs to be indicated

If no IP version is included for any of the 'a=bw-info' lines related to a certain payload type and direction then IPv6 is assumed for all bandwidth properties related to the same direction and payload type, on all of the related 'a=bw-info' lines.

19.2.7 Maximum Packet Rate

Definition:

Identifies the maximum packet rate assumed when calculating the bandwidth properties.

The unit for this bandwidth property shall be packets per second.

Usage during the session:

The overhead when transmitting media using IP/UDP/RTP depends on the packet rate, especially for media using a relatively low bit-rate media, e.g. speech. When IPv4-IPv6 conversion is performed and when the bandwidth properties are re-calculated for the 'a=bw-info' attribute, it is necessary to know which packet rate that was assumed when the bandwidth properties were originally calculated.

The maximum packet rate is used when re-calculating the Maximum Supported Bandwidth, Maximum Desired Bandwidth and Minimum Desired Bandwidth bandwidth properties.

19.2.8 Minimum Packet Rate

Definition:

Identifies the minimum packet rate assumed when calculating the bandwidth properties.

The unit for this bandwidth property shall be packets per second.

Usage during the session:

The overhead when transmitting media using IP/UDP/RTP depends on the packet rate, especially for media using a relatively low bit-rate media, e.g. speech. When IPv4-IPv6 conversion is performed and when the bandwidth properties are re-calculated for the 'a=bw-info' attribute, it is necessary to know which packet rate that was assumed when the bandwidth properties were originally calculated.

The minimum packet rate is used when re-calculating the Minimum Supported Bandwidth bandwidth property.

19.3 SDP attribute

19.3.1 Definition

The syntax for the SDP attribute is:

```
a=bw-info:<pt-def> <direction> <bw-prop-1>=<bw-value-def-1>; ...; <bw-prop-N>=<bw-value-def-N>
```

where:

<pt-def> is the definition of the RTP payload type(s) that the bandwidth information applies to. This can be a single value, a comma-separated list of RTP payload type numbers, or a wild card (*).

<direction> is either 'send' or 'recv'. The direction shall be defined for each 'a=bw-info' line.

<bw-prop-X>=<bw-value-def-X> define the bandwidth property and the related bandwidth value(s).

The new attribute is designed to allow for future extensibility.

19.3.2 SDP grammar

The ABNF RFC 5234 [153] for this attribute is the following:

```

bw-attrib      = "a=bw-info:" pt-def SP direction SP bw-def *("; " [SP] bw-def)

pt-def        = "*" / pt-val *(", " pt-val)
pt-val        = 1*3DIGIT

direction     = "send" / "recv" / "sendrecv" / direction-ext
direction-ext = 1*VCHAR

bw-def        = bw-name "=" bw-val-def
bw-name       = 1*VCHAR           ; Label defining the bandwidth property
bw-val-def    = zero-based-int-or-real / bw-val-def-ext ; Bandwidth value for the bandwidth property
bw-val-def-ext = zero-based-int-or-real *(":" zero-based-int-or-real)
                ; Extension possibility

; DIGIT as defined by IETF RFC 4566
; zero-based-int-or-real as defined by IETF RFC 8866

```

The 'a=bw-info' attribute defines the following possible directionalities for the bandwidth. Three directionalities are defined here:

- send: In the send direction for SDP Offer/Answer agent providing the SDP or in case of declarative use in relation to the device that is being configured by the SDP.
- recv: In the receiving direction for the SDP Offer/Answer agent providing the SDP or in case of declarative use in relation to the device that is being configured by the SDP.
- sendrecv: In both the send and receiving directions for the SDP Offer/Answer agent providing the SDP or in case of declarative use in relation to the device that is being configured by the SDP.

Additional directionalities may be defined in the future, if needed. If an 'a=bw-info' line is received with an unknown directionality then the entire 'a=bw-info' line is ignored.

The directionality shall be specified when the 'a=bw-info' attribute is used. Only one directionality can be specified on each 'a=bw-info' line.

It is allowed to define different bandwidth properties on different attribute lines for the same payload type and direction. However, special care should be taken to avoid conflicting definitions. Therefore, a single bandwidth property shall not be included in several attribute lines applicable to the same payload type, direction and IP version. For example, if bandwidth property 'MaxSupBw' is defined for payload number 96 on one 'a=bw-info' line for direction 'send' and IPv6, then 'MaxSupBw' cannot be defined on another 'a=bw-info' line for the same payload type number, direction and IP version. However, for example, 'MaxSupBw' and 'MinSupBw' may be defined in different 'a=bw-info' lines, even for the same payload type, direction and IP version. This applies also when a wild card is used for the payload type number.

The 'bw-name' is a character string describing the name (or label) used for the bandwidth property that is defined. Four bandwidth property names are defined here for indicating bandwidth needs:

- MaxSupBw: The Maximum Supported Bandwidth, see clause 19.2.2. The Maximum Supported Bandwidth may be a real value.
- MaxDesBw: The Maximum Desired Bandwidth, see clause 19.2.3. The Maximum Desired Bandwidth may be a real value.
- MinDesBw: The Minimum Desired Bandwidth, see clause 19.2.4. The Maximum Desired Bandwidth may be a real value.
- MinSupBw: The Minimum Supported Bandwidth, see clause 19.2.5. The Maximum Supported Bandwidth may be a real value.

These bandwidth properties include IP/UDP/RTP overhead but not RTCP bandwidth, similar to b=AS. This means that the Maximum Supported Bandwidth for the receiving direction corresponds to the b=AS value used in an SDP offer-answer negotiation.

The following bandwidth property names are also defined:

- IpVer: The IP version, see clause 19.2.6. Allowed values are 4 and 6 for IPv4 and IPv6, respectively. If the IP version is not indicated then IPv6 is assumed.
- MaxPRate: The Maximum Packet Rate, see clause 19.2.7. The maximum packet rate may be a real value.
- MinPRate: The Minimum Packet Rate, see clause 19.2.8. The minimum packet rate may be a real value.

Unknown bandwidth property names shall be ignored, and shall not be included in an answer if received in an offer. A received SDP may include 'a=bw-info' lines containing both known and unknown bandwidth properties. Ignoring unknown bandwidth properties shall not cause known bandwidth properties to be ignored.

19.3.3 Declarative use

In declarative usage the SDP attribute is interpreted from the perspective of the end-point being configured by the particular SDP. An interpreter may ignore 'a=bw-info' attribute lines that contain only unknown payload numbers.

19.3.4 Usage in offer/answer

The offer/answer negotiation is performed for each 'a=bw-info' attribute line, payload type, direction and bandwidth property individually.

An offerer may use the 'a=bw-info' attribute line for some or all of the offered payload types. The offerer may use the wild card to describe that a bandwidth property applies to all payload types included in the offer. Different bandwidth properties may be defined on different 'a=bw-info' attribute lines, even for the same payload type (including wild card) and directionality. However, the offerer shall ensure that there is no contradicting information. Therefore, a bandwidth property shall not occur on multiple 'a=bw-info' lines for the same payload type (including wild card), direction and IP version.

An answerer may remove the 'a=bw-info' attribute line(s) for the payload types where it was used in the SDP offer. If 'a=bw-info' is included in the SDP offer, the answerer may add additional 'a=bw-info' lines for payload types it has added in the SDP answer compared to the SDP offer. An answerer may also remove or add individual bandwidth properties.

The SDP may include an offer for some of the defined bandwidth properties without including an offer for other defined bandwidth properties, in which case the values of the omitted properties are undefined, but may still be implicitly limited by the general property equalities defined in clause 19.2.1.

An offer may include any defined directionality in the 'a=bw-info' attribute (e.g. 'send', 'recv' or 'sendrecv') regardless of the directional attribute defined for the media stream ('a=sendrecv', 'a=sendonly', 'a=recvonly' or 'a=inactive').

An agent understanding the 'a=bw-info' attribute and answering to an offer including the 'a=bw-info' attribute should include the attribute in the answer for all payload types for which it was offered.

If an answerer has received 'a=bw-info' in an SDP offer with a certain set of bandwidth properties, and would like to add additional bandwidth properties, possibly using other directionality, then it may do so by adding such definitions in the SDP answer.

An agent may also divide an 'a=bw-info' line into several 'a=bw-info' lines. One example is when the SDP offer included an 'a=bw-info' lines listing several different RTP payload types, or using a wild card. The agent may then divide the attribute line with the payload type list into several attribute lines with payload type lists consisting of fewer payload type numbers or even several attribute lines with only a single payload type number each. Another example is separating a list of bandwidth properties from a single 'a=bw-info' line onto multiple 'a=bw-info' lines.

An agent may also merge several 'a=bw-info' offers into fewer offers or even a single offer.

An agent responding to an 'a=bw-info' offer will need to consider the directionalities and reverse them in the answer when responding to media streams using unicast.

If the answerer removes one or several RTP Payload Types from the SDP when creating the SDP answer then the corresponding payload type numbers should be removed from the 'a=bw-info' lines as well.

An agent accepting an offer with the 'a=bw-info' attribute may modify the bandwidth properties in the following ways when including them in the answer:

- The Maximum Supported Bandwidth may be reduced.
- The Maximum Desired Bandwidth may be reduced.
- The Minimum Desired Bandwidth may be reduced.
- The Minimum Supported Bandwidth may be increased. If the reason for increasing the Minimum Supported Bandwidth is that the Minimum Packet Rate needs to be increased then the Minimum Packet Rate shall also be adjusted.
- The Maximum Packet Rate may be reduced.
- The Minimum Packet Rate may be increased.

19.4 Modifications of the bandwidth information by intermediate network nodes

Networks nodes in the path, capable of understanding and modifying the SDP, may modify the new bandwidth information in the SDP offer for each codec and configuration in the following manner:

- The first node may decrease the maximum supported, maximum desired and minimum desired bandwidth according to network policies. However, the first node should not increase the maximum supported, maximum desired and minimum desired bandwidth except when required to correct undesired or erroneous UE behavior, or for IPv4/IPv6 transport interworking.
- The first node may increase the minimum supported bandwidth according to network policies. However, the first node should not decrease the minimum supported bandwidth except when required to correct UE misbehavior, or for IPv4/IPv6 transport interworking.
- Subsequent nodes may decrease the maximum supported, maximum desired and minimum desired bandwidths and increase the minimum supported bandwidth, but shall not increase the maximum supported, maximum desired and minimum desired bandwidths, and shall also not decrease the minimum supported bandwidth except when required due to IPv4/IPv6 transport interworking.
- Networks nodes may remove an unwanted codec or configuration together with all related bandwidth information.
- Networks nodes may add codecs or configurations accessible via transcoding together with all related bandwidth information.
- If a network node desires to use a MBR=GBR bearer, it should decrease maximum supported bandwidth down to the maximum desired bandwidth in the SDP offer.
- The following relationships shall be maintained by any network node when modifying the bandwidth properties:

Minimum Supported Bandwidth \leq Minimum Desired Bandwidth

Minimum Desired Bandwidth \leq Maximum Desired Bandwidth

Maximum Desired Bandwidth \leq Maximum Supported Bandwidth

NOTE: These rules allow all both the originating and terminating operators in the call setup direction to implement certain policies, but avoid that subsequent operators in the call setup chain counteract the policies of the first operator, and guarantee that the used bandwidths remain in the supported range of the originating UE. For instance, the following policies are supported:

- An operator desiring to set a lower limit to the acceptable QoS can increase the Minimum Supported Bandwidth.
- An operator desiring to limit the GBR for $MBR > GBR$ bearers can decrease the Minimum Desired Bandwidth.
- An operator desiring to limit the GBR for $MBR = GBR$ bearers can decrease the Maximum Supported Bandwidth.

In the SDP answer, the first node should not modify the bandwidth values except when required to correct UE misbehavior, when replacing the selected codec and configuration in the SDP answer due to transcoding, or due to IPv4/IPv6 transport interworking. Subsequent node also shall not modify the bandwidth values except when replacing the selected codec and configuration in the SDP answer due to transcoding, or due to IPv4/IPv6 transport interworking.

Annex A (informative): Examples of SDP offers and answers

A.1 SDP offers for speech sessions initiated by MTSI client in terminal

This Annex includes several SDP examples for session setup for speech. SDP examples for sessions with speech and DTMF are shown in Annex G. These SDP offer and answer examples are designed to highlight the respective area that is being described and should therefore not be considered as complete SDP offers and answers. See TS 24.229 [7] for a complete description of the SDPs. Therefore mandated session parameters such as b=AS should be assumed as present in the media and session level, even if they are not included in the SDP examples.

Some of the SDP examples contain a=fmtp lines that are too long to meet the column width constraints of this document and are therefore folded into several lines using the backslash ("\") character. In a real SDP, long lines would appear as one single line and not as such folded lines.

Some of the examples included in this Annex outline configurations that have been optimized for HSPA. These configurations are equally applicable to E-UTRAN and NR access since the packetization recommendations for HSPA and E-UTRAN and NR in Clauses 7.4.2 and 7.5.2.1 for MTSI clients and Clause 12.3.2 for MTSI media gateways are identical.

A.1.1 HSPA or unknown access technology

When the access technology is unknown to the MTSI client in terminal, the client uses the encapsulation parameters of default operation as defined in clause 7.5.2.1.2. The SDP examples below apply to HSPA as well as the default operation since the encapsulation parameters are the same.

A.1.1.1 Only AMR-NB supported by MTSI client in terminal

In this example one RTP Payload Type (97) is defined for the bandwidth-efficient payload format and another RTP payload type (98) for the octet-aligned payload format. In this case, the MTSI client in terminal supports mode changes at any time, mode changes to any mode and mode change restrictions.

Table A.1.1: SDP example

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 a=maxptime:240</pre> |

Comments:

The UDP port number (49152) and the payload type numbers (97 and 98) are examples and the offerer is free to select other numbers within the restrictions of the UDP and RTP specifications. It is recommended to use the dynamic port numbers in the 49152 to 65535 range. RTP should use even numbers for RTP media and the next higher odd number for RTCP. It is however allowed to use any number within the registered port range 1 024 to 49 151. The receiver must be capable of using any combination of even and odd numbers for RTP and RTCP.

The SDP Capabilities Negotiation framework (SDPCapNeg) [69] is used to negotiate what RTP profile to use. The offer includes RTP/AVP in the conventional SDP part by including it in the media (m=) line, while RTP/AVPF is given as a transport capability using the SDPCapNeg framework "a=tcap:1 RTP/AVPF". A potential configuration gives RTP/AVPF as an alternative "a=pcfg:1 t=1". Given by the rules in SDPCapNeg, the RTP/AVPF profile has higher preference than RTP/AVP.

It is important that the MTSI client in terminal does not define any mode-set because then the answerer is free to respond with any mode-set that it can support. If the MTSI client in terminal would define mode-set to any value, then the answerer only has the option to either accept it or reject it. The latter case might require several ping-pong between the MTSI clients before they can reach an agreement on what mode set to use in the session. This would increase the setup time significantly. This is also one important reason for why the MTSI clients in terminals must support the complete codec mode set of the AMR and AMR-WB codecs, because then a media gateway interfacing GERAN or UTRAN can immediately define the mode-set that it supports on the GERAN or UTRAN circuit switched access.

Since the MTSI client in terminal is required to support mode changes at any frame border and also to any mode in the received media stream, it does not set the mode-change-period and mode-change-neighbor parameters.

The mode-change-capability and max-red parameter are new in the updated AMR payload format [28]. With mode-change-capability=2, the MTSI client in terminal shows that it does support aligning mode changes every other frame and the answerer then knows that requesting mode-change-period=2 in the SDP answer will work properly. The max-red parameter indicates the maximum interval between a non-redundant frame and a redundant frame. Note that the maxptime and max-red parameters do not need to be synchronized.

The payload type for the bandwidth-efficient payload format (97) is listed before the payload type for the octet-aligned payload format (98) because it is the preferred one.

With the combination of ptime:20 and maxptime:240, the MTSI client in terminal shows that it desires to receive one speech frame per packet but can handle up to 12 speech frames per packet. However, no more than 4 original speech frames can be encapsulated in one packet.

A suitable SDP answer from another MTSI client in terminal is shown in Table A.3.0.

A.1.1.2 AMR and AMR-WB are supported by MTSI client in terminal

A.1.1.2.1 One-phase approach

The size of the SDP may become quite big, depending on how many configurations the MTSI client in terminal supports for different media. Therefore, the session setup may be divided into phases where the most desirable configurations are offered in the first phase. If the first phase fails, then the remaining configurations can be offered in a second phase.

In table A.1.2 an example is shown where a one-phase approach is used and where the SDP includes both AMR and AMR-WB and both the bandwidth-efficient and octet-aligned payload formats.

Table A.1.2: SDP example: one-phase approach

| SDP offer |
|--|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240 </pre> |

Comments:

It is easy to imagine that the SDP offer can become quite large if the client supports many different configurations for one or several media. A solution to this problem is shown in Clause A.1.1.2.2.

A few possible SDP answers are outlined in Tables A.3.1, A.3.1a, A.3.2, A.3.3 and A.3.4.

A.1.1.2.2 Two-phase approach

Tables A.1.3 and A.1.4 show the same configurations as in table A.1.2 but when the SDP has been divided into 2 phases.

Table A.1.3: SDP example: 1st phase SDP offer

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 </pre> |

Table A.1.4: SDP example: 2nd phase SDP offer

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVPF 97 98 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240 </pre> |

Comments:

Many types of media and maybe even many different configurations for some or all media types, may give quite large SIP messages. When constructing the offer, the access type and the radio bearer(s) for the answerer are not yet known. To maintain a reasonable setup time, a 2-phase approach may be useful where the most desirable configurations are included in the 1st phase and the 2nd phase is entered only if all payload types for one media type are rejected.

There is however a drawback with the two-phase approach. If the 2nd phase is not entered, then a cell change that would require configurations from the 2nd phase SDP is likely to give long interruption times, several seconds, while the session parameters are re-negotiated.

The SDPCapNeg framework is only used in the 1st SDP offer because when generating the 2nd SDP offer the profile is already agreed. In this example, it is assumed that AVPF was accepted in the first round.

If the 1st SDP offer, shown in Table A.1.3, is accepted by the answerer then a few possible example SDP answers are shown in: Table A.3.1 if the answerer is an MTSI client in terminal and supports AMR-WB; Table A.3.2 if the answerer is an MTSI client in terminal but does not support AMR-WB; Table A.3.3 if the answerer is an MTSI client in terminal, supports AMR-WB and is using EGPRS access; Table A.3.4 if the answerer is a CS terminal, supports AMR-WB and an MTSI MGW is therefore needed.

A.1.2 EGPRS

In this example one RTP Payload Type (97) is defined for the bandwidth-efficient payload format and another RTP Payload Type (98) is defined for the octet-aligned payload format.

Table A.1.5: SDP example

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=200 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=200; octet-align=1 aptime:40 a=maxptime:240</pre> |

Comments:

The only difference compared with the SDP offer for HSPA is ptime: 40. This definition is used to optimize capacity by reducing the amount of overhead that lower layers introduce. Defining ptime:20 will also work, but will be less optimal. Thus, when performing a cell change from HSPA to EGPRS, it is not an absolute necessity to update the session parameters immediately. It can be done after a while, which would also reduce the amount of SIP signalling if a MTSI client in terminal is switching frequently between HSPA and EGPRS or some other access type.

It is recommended to set the max-red parameter to an integer multiple of the ptime.

An example of a suitable SDP answer to this SDP offer is shown in Table A.3.3a.

A.1.3 Generic Access

In this example one RTP Payload Type (97) is defined for the bandwidth-efficient payload format and another RTP Payload Type (98) is defined for the octet-aligned payload format.

Table A.1.6: SDP example

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=160 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=160; octet-align=1 a=ptime:80 a=maxptime:240</pre> |

Comments:

In this case the MTSI client in terminal has detected that the load on the WLAN network is quite high and therefore ptime is set to 80. For other operating conditions, it could set ptime to 20, 40 or 60. This parameter may be updated during the session if the load of the WLAN network changes.

An example of a suitable SDP answer to this SDP offer is shown in Table A.3.3b.

A.2 SDP offers for speech sessions initiated by media gateway

A.2.1 General

These examples show only SDP offers when the MTSI media gateway does not support the same configurations as for the MTSI terminal in clause A.1. A media gateway supporting the same configurations as for the examples in clause A.1 should create the same SDP offers.

A.2.2 MGW between GERAN UE and MTSI

This example shows the SDP offer when the call is initiated from GSM CS using the AMR with the {12.2, 7.4, 5.9 and 4.75} codec mode set. In this example, it is also assumed that only the bandwidth-efficient payload format is supported and that it will not send any redundant speech frames.

Table A.2.1: SDP example

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=0,2,4,7; mode-change-period=2, \ mode-change-neighbor=1; mode-change-capability=2; max-red=0 a=ptime:20 a=maxptime:80</pre> |

Comments:

Since the MGW only supports a subset of the AMR codec modes, it needs to indicate this in the SDP. The same applies for the mode change restrictions.

An example of a suitable SDP answer to this SDP offer is shown in Table A.3.5.

A.2.3 MGW between legacy UTRAN UE and MTSI

This example shows the SDP offer when the call is initiated from legacy UTRAN CS mobile that only the AMR 12.2 mode. In this example, it is also assumed that only the bandwidth-efficient payload format is supported.

Table A.2.2: SDP example

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=7; max-red=0 aptime:20 a=maxptime:20</pre> |

Comments:

Since only one mode is supported, the mode-change-period, mode-change-neighbor and mode-change-capability parameters do not apply.

In this case it is advisable to not allow redundancy since the legacy UTRAN CS mobile does not support any lower rate codec modes and then redundancy would almost double the bitrate on the PS access side. Therefore, maxptime is set to 20 and max-red is set to 0.

If a mode-set with several codec modes was defined and if max-red and maxptime are set to larger values than what Table A.1.8 shows, then redundancy is possible on the PS access side but not together with TFO.

An example of a suitable SDP answer to this SDP offer is shown in Table A.3.6.

A.2.4 MGW between CS UE and MTSI

This example shows the SDP offer when two mode sets are supported by the MGW.

Table A.2.3: SDP example

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=0,2,4,7; mode-change-period=2, \ mode-change-neighbor=1; mode-change-capability=2; max-red=20 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-set=0,3,5,6; mode-change-period=2, \ mode-change-neighbor=1; mode-change-capability=2; max-red=20 aptime:20 a=maxptime:80</pre> |

Comments:

Redundancy up to 100 % is supported in this case since max-red is set to 20.

An example of a suitable SDP answer to this SDP offer is shown in Table A.3.6.

A.2.5 MGW between GERAN UE and MTSI when wideband speech is supported

This example shows the SDP offer when the call is initiated from GSM CS when AMR is supported with the {12.2, 7.4, 5.9 and 4.75} codec mode set and when AMR-WB is supported with the {12.65, 8.85 and 6.60} mode set. In this example, it is also assumed that only the bandwidth-efficient payload format is supported and that the MTSI MGW will not send any redundant speech frames.

Table A.2.4: SDP example

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 98 97 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=0,2,4,7; mode-change-period=2, \ mode-change-neighbor=1; mode-change-capability=2; max-red=0 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-set=0,1,2; mode-change-period=2, \ mode-change-neighbor=1; mode-change-capability=2; max-red=0 aptime:20 a=maxptime:80</pre> |

Comments:

Since the MGW only supports a subset of the AMR codec modes and of the AMR-WB codec modes, it needs to indicate this in the SDP. The same applies for the mode change restrictions.

A.3 SDP answers to SDP speech session offers

A.3.1 General

This clause gives a few examples of possible SDP answers. The likelihood that these SDP answers will be used may vary from case to case since the SDP answer depends on circumstances outside the scope of this specification for example: availability of resources, radio bearer assignment and policy control. It is impossible to cover all the possible variants and hence these examples should be regarded as just a few examples of many possible alternatives. They were however selected because they span the range of possible SDP answers quite well.

The SDP offers are included to clarify what is being answered.

A.3.1a SDP answer from an MTSI client in terminal when only narrowband speech was offered

These SDP offers and answer are likely when the offering MTSI client in terminal (or other client) only supports narrowband speech (AMR).

The SDP offer included in this example is identical to the SDP offer shown in Table A.1.1.

Table A.3.0: SDP example

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 a=maxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVPF 99 a=acfg:1 t=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 aptime:20 a=maxptime:240</pre> |

Comments:

The SDP answer contains only one encoding format since TS 24.229 [7] requires that the answerer shall select exactly one codec for the answer. Since both MTSI clients in terminals support the same configurations for narrowband speech, it is likely that the selected configuration included in the answer is identical to the configuration in the offer and that no mode-set is defined by the terminating client.

The conclusion from this offer-answer procedure is that the offerer can only send AMR encoded speech to the answerer using the bandwidth-efficient payload type with RTP Payload Type 99, since this was the only configuration included in the answer. The answerer sends AMR encoded speech to the offerer using the bandwidth-efficient payload format, in this case RTP Payload Type 97.

Even though both MTSI clients in terminals support all codec modes, it is desirable to mainly use the codec modes from the AMR {12.2, 7.4, 5.9 and 4.75} mode set because the set includes codec modes frequently used in GERAN and UTRAN, and enables to control quality and capacity with appropriate bit-rate granularity.

Unless transmission conditions necessitate other encapsulation types it is also desirable to encapsulate only 1 speech frame per packet, even though both MTSI clients in terminals support receiving several frames per packet.

In the above example it is assumed that AVPF will be accepted since the MTSI client is required to support this RTP profile.

A.3.2 SDP answer from an MTSI client in terminal

These SDP offers and answers are likely when both MTSI clients in terminals support AMR and AMR-WB and also both the bandwidth-efficient and the octet-aligned payload formats.

The SDP offer included in this example is identical to the SDP offer shown in Table A.1.2, with the exception that the number of channels is omitted for each of the codecs. This implies that the terminal is offering one channel for each codec.

Table A.3.1: SDP example

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 98 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240</pre> |
| SDP answer if AVPF is accepted |
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtpmap:97 AMR-WB/16000 a=fmtp:97 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240</pre> |

Comments:

The SDP answer contains only one encoding format since TS 24.229 [7] requires that the answerer shall select exactly one codec for the answer. Since both MTSI clients in terminals support the same configurations, it is likely that the selected configuration included in the answer is identical to the configuration in the offer and that no mode-set is defined by the terminating client. The conclusion from this offer-answer process is that AMR-WB will be used during the session with RTP Payload Type 97. The SDP answer does not include the number of audio channels, implying that one channel has been accepted.

Even though both MTSI clients in terminals support all codec modes, it is desirable to mainly use the codec modes from the AMR-WB {12.65, 8.85 and 6.60} mode set because the set includes codec modes frequently used in GERAN and UTRAN, and enables to control quality and capacity with appropriate bit-rate granularity.

Unless transmission conditions necessitate other encapsulation types it is also desirable to encapsulate only 1 speech frame per packet, even though both MTSI clients in terminals support receiving several frames per packet.

In the above example it is assumed that AVPF will be accepted since the MTSI client is required to support this RTP profile.

This SDP answer is also a possible answer to the SDP offer shown in Table A.1.3.

A.3.2a SDP answer from a non-MTSI UE with AVP

The MTSI client must be prepared to receive an SDP answer with AVP. This is likely to occur for legacy clients that do not support AVPF or SDPCapNeg. The example in Table A.3.1a shows a possible SDP answer with AVP to an SDP offer as shown in Table A.1.2.

Table A.3.1a: SDP answer example with AVP

| SDP answer with AVP |
|---|
| <pre>m=audio 49152 RTP/AVP 97 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 aptime:20 a=maxptime:240</pre> |

Comments:

A client that does not support SDPCapNeg would not understand the attributes defined by the SDPCapNeg framework and would therefore ignore the lines with 'a=tcap' and 'a=pcfg'.

A.3.3 SDP answer from an MTSI client in terminal supporting only AMR

These SDP offers and answers are likely when the answering MTSI client in terminal supports only AMR.

The SDP offer included in this example is identical to the SDP offer shown in Table A.1.2.

Table A.3.2: SDP example

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 98 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 a=maxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVPF 99 a=acfg:1 t=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 aptime:20 a=maxptime:240</pre> |

Comments:

In the answer, RTP Payload Types 97 and 98 have been removed since AMR-WB is not supported and RTP Payload Type 100 is removed since the answerer is required to answer with only one encoding format.

Even though both MTSI clients in terminals support all codec modes, it is desirable to mainly use the codec modes from the AMR [12.2, 7.4 5.9 and 4.75] mode set because the set includes codec modes frequently used in GERAN and UTRAN, and enables to control quality and capacity with appropriate bit-rate granularity.

This SDP answer is also a possible answer to the SDP offer shown in Table A.1.3.

A.3.4 SDP answer from an MTSI client in terminal using EGPRS access when both AMR and AMR-WB are supported

In this case the answering MTSI client in terminal is using EGPRS access and supports both narrowband and wideband speech, i.e. both AMR and AMR-WB.

The SDP offer is identical to the SDP offer shown in Table A.1.2.

Table A.3.3: SDP example

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 98 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 a=maxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=200 aptime:40 a=maxptime:240</pre> |

Comments:

The answering MTSI client in terminal responds that it desires to receive 2 frames encapsulated in each packet. It will however send with 1 frame per packet since the offering MTSI client in terminal desires to receive this format. A future SIP UPDATE may change this so that 2 frames per packet are used in both directions.

The answering MTSI client in terminal also responds with max-red defined to 200 ms since this is the closes multiple of the desired frame aggregation. It should however be noted that it is not a requirement to define max-red to be a multiple of ptime, but it is recommended to do so.

This SDP answer is also a possible answer to the SDP offer shown in Table A.1.3.

A.3.4a SDP answer from an MTSI client in terminal using EGPRS access when only AMR is supported

In this case the answering MTSI client in terminal is using EGPRS access but supports only narrowband speech, i.e. only AMR.

The SDP offer is identical to the SDP offer shown in Table A.1.2 although the SDP answer here would also work nicely as a response to the SDP offer shown in Table A.1.5.

Table A.3.3a: SDP example

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 98 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 amaxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=200 aptime:40 amaxptime:240</pre> |

Comments:

The answering MTSI client in terminal responds that it desires to receive 2 frames encapsulated in each packet. It will however send with 1 frame per packet since the offering MTSI client in terminal desires to receive this format. A future SIP UPDATE may change this so that 2 frames per packet are used in both directions.

The answering MTSI client in terminal also responds with max-red defined to 200 ms since this is the closes multiple of the desired frame aggregation. It should however be noted that it is not a requirement to define max-red to be a multiple of ptime, but it is recommended to do so.

This SDP answer is also a suitable response to an SDP offer as shown in Table A.1.5, even if the answering MTSI client in a terminal is using HSPA access. This is because it is wise to harmonize the packetization, and the ptime in the SDP answer, with the ptime in the SDP offer so that 2 frames per packet will be used in both directions when one of the end-points is using EGPRS access.

A.3.4b SDP answer from an MTSI client in terminal using WLAN

In this example, the MTSI client in terminal is using a WLAN network.

The SDP offer shown here is identical to the SDP offer shown in Table A.1.1.

Table A.3.3b: SDP example

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=160 a=ptime:80 a=maxptime:240</pre> |

Comments:

This SDP answer, with ptime=80, is suitable if the MTSI client in terminal can measure the load in the WLAN network and has detected that the load is high. If, on the other hand, the load is low then the MTSI client in terminal may very well choose to use a lower value for the ptime attribute, for example 20 or 40.

This SDP answer is also suitable when the answering MTSI client in terminal is using HSPA access but when the offerer is using WLAN and indicates ptime=80, e.g. as shown in the SDP offer in Table A.1.6. In such a case, it is wise to harmonize the ptime values in both directions to increase the likelihood that several frames will be encapsulated in the RTP packets.

This SDP answer is also a possible answer to the SDP offer shown in Table A.1.3.

A.3.5 SDP answer from MTSI MGW supporting only one codec mode set for AMR and AMR-WB each

In this case the MTSI MGW supports only one codec mode set for AMR, {12.2, 7.4, 5.9 and 4.75}, and one codec mode set for AMR-WB, {12.65, 8.85 and 6.60}, since the CS terminal only supports these mode sets. The MTSI MGW also only supports the bandwidth-efficient payload format.

The SDP offer included in this example is identical to the SDP offer shown in Table A.1.2.

Table A.3.4: SDP example

| SDP offer (from MTSI client in terminal on HSPA) |
|--|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240 </pre> |
| SDP answer (from MTSI MGW) |
| <pre> m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-set=0,1,2; mode-change-period=2, mode-change-neighbor=1; \ mode-change-capability=2; max-red=0 a=ptime:20 a=maxptime:80 </pre> |

Comments:

The MTSI MGW is allowed to define the mode-set parameter since the MTSI client in terminal did not define it. Thereby, it is possible to avoid several SDP offers and answers.

The SDP answer contains only one encoding format since TS 24.229 [7] requires that the answerer shall select exactly one codec for the answer. In this case, the CS terminal supports wideband speech and the MTSI MGW therefore chooses to establish a wideband speech session.

Since the MTSI client in terminal has defined that it does support restrictions in mode changes, the MTSI MGW can safely set the mode-change-period and mode-change-neighbor parameters.

In this example, the MTSI MGW also does not support redundancy so it sets max-red to zero.

This SDP answer is also a possible answer to the SDP offer shown in Table A.1.3.

A.3.5a SDP offer from MTSI MGW supporting only one codec mode set for AMR

In this case the MTSI MGW supports only one codec mode set for AMR, {12.2, 7.4, 5.9 and 4.75}, since the CS terminal supports only this mode set. The MTSI MGW also only supports the bandwidth-efficient payload format.

Table A.3.4a: SDP example

| SDP offer (from MTSI client in terminal on HSPA) |
|---|
| <pre>m=audio 49152 RTP/AVP 97 98 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 a=maxptime:240</pre> |
| SDP answer (from MTSI MGW) |
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=0,2,4,7; mode-change-period=2, mode-change-neighbor=1; \ mode-change-capability=2; max-red=0 aptime:20 a=maxptime:80</pre> |

Comments:

The MTSI MGW is allowed to define the mode-set parameter since the MTSI client in terminal did not define it. Thereby, it is possible to avoid several SDP offers and answers.

The SDP answer contains only one encoding format since TS 24.229 [7] requires that the answerer shall select exactly one codec for the answer. In this case, the CS terminal does not supports wideband speech and the MTSI MGW therefore selects to establish a narrowband speech session.

Since the MTSI client in terminal has defined that it does support restrictions in mode changes, the MTSI MGW can safely set the mode-change-period and mode-change-neighbor parameters.

In this example, the MTSI MGW also does not support redundancy so it sets max-red to zero.

A.3.6 SDP answer from MTSI client in terminal on HSPA for session initiated from MTSI MGW interfacing UE on GERAN

This example shows the offers and answers for a session between a GERAN CS UE, through a MTSI media gateway, and a MTSI client in terminal.

The SDP offer shown here is very similar to the SDP offer shown in Table A.2.1. The only difference is that maxptime is set to 20.

Table A.3.5: SDP example

| SDP offer (from MTSI MGW) |
|--|
| <pre>m=audio 49152 RTP/AVP 97 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=0,2,4,7; mode-change-period=2, \ mode-change-neighbor=1; mode-change-capability=2; max-red=0 aptime:20 amaxptime:20</pre> |
| SDP answer (from MTSI client in terminal) |
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=0,2,4,7; mode-change-period=2, \ mode-change-neighbor=1; mode-change-capability=2; max-red=0 aptime:20 amaxptime:240</pre> |

Comments:

The MTSI media gateway offers only a restricted mode set since the CS terminal does not support anything else. The MTSI client in terminal has to accept this, if it wants to continue with the session setup.

This example also shows that the MTSI media gateway wants to receive only 1 frame per packet. The maxptime parameter is therefore set to 20. With max-red set to 0 the MTSI media gateway also shows that it will not send redundancy. The MTSI terminal can support receiving up to 12 frames per packet. It therefore set the maxptime parameter to 240.

The MTSI client in terminal detects that the MTSI media gateway does not want to receive redundancy and therefore sets max-red to 0.

The SDP answer shown in this example is also a suitable answer to the SDP offer shown in Table A.2.1. This SDP answer is also suitable for the SDP offer shown Table A.2.3.

A.3.7 SDP answer from MTSI client in terminal on HSPA for session initiated from MTSI MGW interfacing legacy UE on UTRAN

This example shows the offers and answers for a session between a legacy UTRAN CS UE that only supports AMR 12.2, through a MTSI media gateway, and a MTSI client in terminal.

The SDP offer shown here is identical to the SDP offer shown in Table A.2.2.

Table A.3.6: SDP example

| SDP offer (from MTSI MGW) |
|--|
| <pre>m=audio 49152 RTP/AVP 97 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=7; max-red=0 aptime:20 amaxptime:20</pre> |
| SDP answer (from MTSI client in terminal) |
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-set=7; max-red=0 aptime:20 amaxptime:240</pre> |

Comments:

The MTSI media gateway offers only one codec mode set since the CS terminal does not support anything else. The MTSI client in terminal has to accept this, if it wants to continue with the session setup.

This example also shows that the MTSI media gateway want to receive only 1 frame per packet. The maxptime parameter is therefore set to 20. With max-red set to 0 the MTSI media gateway also shows that it will not send redundancy. The MTSI terminal can support receiving up to 12 frames per packet. It therefore set the maxptime parameter to 240.

The MTSI client in terminal detects that the MTSI media gateway does not want to receive redundancy and therefore sets max-red to 0.

A.4 SDP offers and answers for video sessions

A.4.1 Void

A.4.2 Void

A.4.2a H.264/AVC

In this example the SDP offer includes H.264/AVC.

Table A.4.2a1: Example SDP offer for H.264/AVC

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The offered codec is H.264/AVC. The packetization-mode parameter indicates single NAL unit mode. This is the default mode and it is therefore not necessary to include this parameter (see RFC 6184 [25]). The profile-level-id parameter indicates Constrained Baseline profile at level 1.2, which supports bitrates up to 384 kbps. It also indicates, by using so-called constraint-set flags, that the bit stream can be decoded by any Baseline, Main or Extended profile decoder. The third parameter, sprop-parameter-sets, includes base-64 encoded sequence and picture parameter set NAL units that are referred by the video bit stream. The sequence parameter set used here includes syntax that specifies the number of re-ordered frames to be zero so that latency can be minimized. The bandwidth (including IP, UDP and RTP overhead) for video is restricted to 315 kbps.

The negotiation of AVPF features is also shown. By setting 'trr-int' to 5000 the MTSI client indicates that the minimum interval between two regular RTCP packets needs to be 5 seconds, [40]. The 'nack' and 'nack pli' parameters indicate that the MTSI client supports NACK (Generic NACK) and PLI (Picture Loss Indication) as defined by AVPF, [40]. The 'ccm fir' and 'ccm tmmbr' parameters indicate that the MTSI client supports the FIR (Full Intra Request) and TMMBR (Temporary Maximum Media Stream Bit Rate Request), [43]. The wildcard ('*') indicates that it is possible to use these features for all RTP payload types for the video stream.

The negotiation of the video orientation header extension is made with the a=extmap attribute [95]. In this example, the local identifier (ID) is set to '4'. This number is only an example and other values may be used.

An example SDP answer to the offer is given below.

Table A.4.2a2: Example SDP answer

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The responding MTSI client is capable of using H.264/AVC. As the offer already indicated the lowest level (level 1.2) of H.264/AVC as well as the minimum constraint set, there is no room for further negotiation of profiles and levels. However, the bandwidth could be constrained further by reducing the bandwidth in b=AS.

A.4.2b High Granularity CVO example

This example is identical to A.4.2a with the exception of higher granularity CVO being offered.

Table A.4.2b.1: Example SDP offer with High Granularity

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=extmap:5 urn:3gpp:video-orientation:6 </pre> |

The offer for higher granularity is indicated in the last SDP line above.

Table A.4.2b.2: Example SDP answer with High Granularity

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:5 urn:3gpp:video-orientation:6 </pre> |

The answer indicates that higher granularity has been accepted as indicated by the last SDP line above.

A.4.2c RTP Retransmission

This example is identical to A.4.2a with the exception of retransmission being offered.

Table A.4.2c.1: Example SDP offer with Retransmission

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVP 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtpmap:100 rtx/90000 a=fmtp:100 apt=99;rtx-time=400 a=rtcp-fb:* trr-int 5000 a=rtcp-fb:99 nack a=rtcp-fb:99 nack pli a=rtcp-fb:99 ccm fir a=rtcp-fb:99 ccm tmmbr </pre> |

The offer for retransmission and associated parameters are listed after the line describing the format properties of the H.264 video.

Table A.4.2c.2: Example SDP answer with Retransmission

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 99 100 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtpmap:100 rtx/90000 a=fmtp:100 apt=99;rtx-time=400 a=rtcp-fb:* trr-int 5000 a=rtcp-fb:99 nack a=rtcp-fb:99 nack pli a=rtcp-fb:99 ccm fir a=rtcp-fb:99 ccm tmmbr </pre> |

The answer indicates that retransmission has been accepted as indicated.

A.4.2d RTP Forward Error Correction (FEC)

This example is identical to A.4.2a with the exception of FEC being offered.

Table A.4.2d.1: Example SDP offer with FEC

| SDP offer |
|---|
| <pre> m=video 49154 RTP/AVP 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtpmap:100 flexfec/90000 a=rtpmap:100 repair-window:150000 a=ssrc:1234 a=ssrc:2345 a=ssrc-group:FEC-FR 1234 2345 a=rtcp-fb:* trr-int 5000 a=rtcp-fb:99 nack a=rtcp-fb:99 nack pli a=rtcp-fb:99 ccm fir a=rtcp-fb:99 ccm tmmbr </pre> |

The offer for FEC and associated parameters are listed after the line describing the format properties of the H.264 video.

Table A.4.2d.2: Example SDP answer with FEC

| SDP answer |
|--|
| <pre> m=video 49154 RTP/AVPF 99 100 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtpmap:100 flexfec/90000 a=rtpmap:100 repair-window:150000 a=ssrc:1234 a=ssrc:2345 a=ssrc-group:FEC-FR 1234 2345 a=rtcp-fb:* trr-int 5000 a=rtcp-fb:99 nack a=rtcp-fb:99 nack pli a=rtcp-fb:99 ccm fir a=rtcp-fb:99 ccm tmmbr </pre> |

The answer indicates that FEC has been accepted as indicated.

A.4.2e SDP Examples with ROI

This example is identical to A.4.2a with the exception of ‘Arbitrary ROI’ and ‘Sent ROI’ being offered.

Table A.4.2e.1: Example SDP offer with ‘Arbitrary ROI’ and ‘Sent ROI’

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=rtcp-fb:* 3gpp-roi-arbitrary a=extmap:7 urn:3gpp:roi-sent </pre> |

The offer for ‘Arbitrary ROI’ and ‘Sent ROI’ are indicated in the last two lines.

The use of RTCP feedback messages carrying ‘Arbitrary ROI’ is negotiated with the ‘3gpp-roi-arbitrary’ parameter. The wildcard (*) indicates that it is possible to use the ROI features for all RTP payload types including video.

The use of the ‘Sent ROI’ header extension carrying arbitrary ROI information is negotiated with the a=extmap attribute [95] based on the URN ‘urn:3gpp:roi-sent’. In this example, the local identifier (ID) is set to 7, which is only an example. Other values may be used as long as a distinct ID is assigned for each extmap attribute corresponding to different URNs.

Table A.4.2e.2: Example SDP answer with ‘Arbitrary ROI’ and ‘Sent ROI’

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=rtcp-fb:* 3gpp-roi-arbitrary a=extmap:7 urn:3gpp:roi-sent </pre> |

The answer indicates that both ‘Arbitrary ROI’ and ‘Sent ROI’ have been accepted.

The following example is identical to A.4.2a with the exception of ‘Pre-defined ROI’ and ‘Sent ROI’ being offered.

Table A.4.2e.3: Example SDP offer with 'Pre-defined ROI' and 'Sent ROI'

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=predefined_ROI:99 [ROI_ID=0,Position_X=1,Position_Y=1,Size_X=1080,Size_Y=720, \ Name=fullview] [ROI_ID=1,Position_X=1,Position_Y=1,Size_X=540,Size_Y=360, \ Name=museum] [ROI_ID=2,Position_X=541,Position_Y=1,Size_X=540,Size_Y=360, \ Name=cinema] [ROI_ID=3,Position_X=1,Position_Y=361,Size_X=540,Size_Y=360, \ Name=park] [ROI_ID=4,Position_X=541,Position_Y=361,Size_X=540,Size_Y=360,Name= zoo] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=rtcp-fb:* 3gpp-roi-predefined a=extmap:7 urn:3gpp:predefined-roi-sent </pre> |

The offer for 'Pre-defined ROI' and 'Sent ROI' are indicated in the last two lines.

The offered set of pre-defined ROIs is provided by the "a=predefined_ROI" attribute. The use of RTCP feedback messages carrying 'Pre-defined ROI' is negotiated with the '3gpp-roi-predefined' parameter. The wildcard ('*') indicates that it is possible to use the ROI features for all RTP payload types including video.

The use of the 'Sent ROI' header extension carrying pre-defined ROI information is negotiated with the a=extmap attribute [95] based on the URN 'urn:3gpp:predefined-roi-sent'. In this example, the local identifier (ID) is set to 7, which is only an example. Other values may be used as long as a distinct ID is assigned for each extmap attribute corresponding to different URNs.

Table A.4.2e.4: Example SDP answer with 'Pre-defined ROI' and 'Sent ROI'

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=predefined_ROI:99 [ROI_ID=0,Position_X=1,Position_Y=1,Size_X=1080,Size_Y=720, \ Name=fullview] [ROI_ID=1,Position_X=1,Position_Y=1,Size_X=540,Size_Y=360, \ Name=museum] [ROI_ID=2,Position_X=541,Position_Y=1,Size_X=540,Size_Y=360, \ Name=cinema] [ROI_ID=3,Position_X=1,Position_Y=361,Size_X=540,Size_Y=360, \ Name=park] [ROI_ID=4,Position_X=541,Position_Y=361,Size_X=540,Size_Y=360,Name= zoo] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=rtcp-fb:* 3gpp-roi-predefined a=extmap:7 urn:3gpp:predefined-roi-sent </pre> |

The answer indicates that both 'Pre-defined ROI' and 'Sent ROI' have been accepted.

The following example is identical to A.4.2a with the exception of FECC, 'Arbitrary ROI' and 'Sent ROI' being offered.

Table A.4.2e.5: Example SDP offer with FECC, 'Arbitrary ROI' and 'Sent ROI'

| SDP offer |
|---|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=rtcp-fb:* 3gpp-roi-arbitrary a=extmap:7 urn:3gpp:roi-sent m=application 50000 RTP/AVPF 99 a=rtpmap:99 h224/4800 a=sendonly </pre> |

The offer for FECC is made according to the procedures specified in [139]. In this example, the MTSI client offers a sendonly channel since it is unwilling to adjust the video ROI during encoding based on PTZF commands received

from the far end and it does not intend to use H.224 to learn the capabilities of the far end. At the same time, if the far end is capable of FECC, it indicates that it can take advantage of this capability and send PTZF commands to adjust video ROI for the video stream in the receive direction.

Table A.4.2e.6: Example SDP answer with FECC, 'Arbitrary ROI' and 'Sent ROI'

| SDP answer |
|--|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=rtcp-fb:* 3gpp-roi-arbitrary a=extmap:7 urn:3gpp:roi-sent m=application 50000 RTP/AVPF 99 a=rtpmap:99 h224/4800 a=recvonly </pre> |

The answer indicates that FECC, 'Arbitrary ROI' and 'Sent ROI' are all accepted. On FECC, the MTSI client answers with a recvonly confirming that it supports the FECC protocol and would be willing to adjust the video ROI during encoding based on PTZF commands received from the far end. As such, FECC can only be used for video in one direction.

The following example is identical to A.4.2a with the exception of FECC, 'Pre-defined ROI' and 'Sent ROI' being offered.

Table A.4.2e.7: Example SDP offer with FECC, 'Pre-defined ROI' and 'Sent ROI'

| SDP offer |
|---|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=predefined_ROI:99 [ROI_ID=0,Position_X=1,Position_Y=1,Size_X=1080,Size_Y=720, \ Name=fullview] [ROI_ID=1,Position_X=1,Position_Y=1,Size_X=540,Size_Y=360, \ Name=museum] [ROI_ID=2,Position_X=541,Position_Y=1,Size_X=540,Size_Y=360, \ Name=cinema] [ROI_ID=3,Position_X=1,Position_Y=361,Size_X=540,Size_Y=360, \ Name=park] [ROI_ID=4,Position_X=541,Position_Y=361,Size_X=540,Size_Y=360,Name= zoo] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=rtcp-fb:* 3gpp-roi-predefined a=extmap:7 urn:3gpp:predefined-roi-sent m=application 50000 RTP/AVPF 99 a=rtpmap:99 h224/4800 a=sendrecv </pre> |

In this example, the MTSI client offers a sendrecv channel for FECC since it is willing to adjust the video ROI during encoding based on PTZF commands received from the far end and it intends to use H.224 to learn the capabilities of the far end.

Table A.4.2e.8: Example SDP answer with FECC, 'Pre-defined ROI' and 'Sent ROI'

| SDP answer |
|--|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=predefined_ROI:99 [ROI_ID=0,Position_X=1,Position_Y=1,Size_X=1080,Size_Y=720, \ Name=fullview] [ROI_ID=1,Position_X=1,Position_Y=1,Size_X=540,Size_Y=360, \ Name=museum] [ROI_ID=2,Position_X=541,Position_Y=1,Size_X=540,Size_Y=360, \ Name=cinema] [ROI_ID=3,Position_X=1,Position_Y=361,Size_X=540,Size_Y=360, \ Name=park] [ROI_ID=4,Position_X=541,Position_Y=361,Size_X=540,Size_Y=360,Name= zoo] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=rtcp-fb:* 3gpp-roi-predefined a=extmap:7 urn:3gpp:predefined-roi-sent m=application 50000 RTP/AVPF 99 a=rtpmap:99 h224/4800 a=sendonly </pre> |

The answer indicates that FECC, 'Pre-defined ROI' and 'Sent ROI' are all accepted. For FECC, the MTSI client answers with a sendonly since it is unwilling to adjust the video ROI during encoding based on PTZF commands received from the far end and it does not intend to use H.224 to learn the capabilities of the far end. At the same time, since the far end is capable of FECC, it indicates that it can take advantage of this capability and send PTZF commands to adjust video ROI for the video stream in the receive direction. As such, FECC can only be used for video in one direction.

A.4.3 Void

A.4.4 Void

A.4.4a H.264/AVC with "imageattr" attribute

In this example the SDP offer includes H.264/AVC with negotiation of the image size using the "imageattr" attribute.

Table A.4.10a: Example SDP offer for H.264/AVC with image size negotiation

| SDP offer |
|--|
| <pre> a=tcap:1 RTP/AVPF m=video 49154 RTP/AVP 99 a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtppmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=imageattr:99 send [x=176,y=144] [x=224,y=176] [x=272,y=224] [x=320,y=240] recv [x=176,y=144] [x=224,y=176] [x=272,y=224,q=0.6] [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The offered codec is H.264/AVC. The packetization-mode parameter indicates single NAL unit mode. This is the default mode and it is therefore not necessary to include this parameter (see RFC 6184 [25]). The profile-level-id parameter indicates Baseline profile at level 1.2, which supports bitrates up to 384 kbps. It also indicates, by using so-called constraint-set flags, that the bit stream can be decoded by any Baseline, Main or Extended profile decoder. The bandwidth (including IP, UDP and RTP overhead) for video is 315 kbps. The third parameter, sprop-parameter-sets, includes base-64 encoded sequence and picture parameter set NAL units that are referred by the video bit stream. The sequence parameter set used here includes syntax that specifies the number of re-ordered frames to be zero so that latency can be minimized. sprop-parameter-sets is constructed assuming the offered conditions and image size of 320x240, which is the largest of all offered sizes for send direction. The offering MTSI client offers four image sizes for both send and receive directions but prefers 272x224 for receive direction, which might fit the available space on its display better than the other image sizes.

Since the support of a particular codec level does not imply that the video encoder has to produce a bitstream up to the maximum capability of the level, it may be useful for an MTSI client to indicate the image sizes it can encode video at for each codec it supports, using the "imageattr" SDP attribute [76]. Then on the receiving side, the MTSI client can indicate which of these image sizes it prefers to receive. This reduces the loss of quality from rescaling the decoded image to fit the available space on the receiver's display.

An example SDP answer to the offer is given below.

Table A.4.10b: Example SDP answer

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtppmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=imageattr:99 send [x=320,y=240] recv [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The responding MTSI client is capable of using H.264/AVC. The responding MTSI client agreed to use a bandwidth of 315 kbps and to use the Baseline profile at level 1.2. From the four image sizes offered, the responding MTSI client included 320x240 for both send and receive directions. Although the offering MTSI client preferred 272x224 for receive direction, the responding MTSI client might not be able to offer 272x224 or not allow encoding and decoding of video of different image sizes simultaneously. The responding MTSI client sent new sprop-parameter-sets for the video decoder of the offering MTSI client, which was constructed assuming the agreed conditions and image size of 320x240.

Table A.4.11: Example SDP answer

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:107 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0;profile-level-id=42e00b; \ sprop-parameter-sets=Z0LgC5ZUCg/I,aM4BrFSAA a=imageattr:99 send [x=272,y=224] recv [x=320,y=240,q=0.6] [x=272,y=224] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbbr a=extmap:4 urn:3gpp:video-orientation </pre> |

In this alternative answer, the responding MTSI client has restricted the video bandwidth to 107 kbps and restricted the H.264/AVC level to 1.1 which supports bitrates up to 192 kbps. The restricted H.264/AVC level should be high enough to enable the image sizes for both send and receive directions. From the four image sizes offered, the responding MTSI client included 272x224 for send direction, which was preferred by the offering MTSI client. For the receive direction, the responding MTSI client included 320x240 as preferred and 272x224 as fallback because the responding MTSI client was unsure whether the offering MTSI client can encode and decode video of different image sizes simultaneously at the conditions. The responding MTSI client sent new sprop-parameter-sets for the video decoder of the offering MTSI client, which was constructed assuming the restricted conditions. Though not required in response to the SDP answer indicating more than one image size in the receive direction, the offering MTSI client may chose to send a second offer as shown in Table A.4.12 to confirm that it is able to encode and decode at different image sizes.

Table A.4.12: Example Second SDP offer

| Second SDP Offer |
|--|
| <pre> m=video 49154 RTP/AVPF 99 b=AS:107 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0;profile-level-id=42e00b; \ sprop-parameter-sets=Z0LgC5ZUCg/I,aM4BrFSAA a=imageattr:99 send [x=320,y=240] recv [x=272,y=224] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbbr a=extmap:4 urn:3gpp:video-orientation </pre> |

Since now the offering MTSI client knows that the responding MTSI client supports and prefers to use AVPF, AVPF is offered without SDPCapNeg and AVP.

A.4.4a.1 H.264/AVC with "imageattr" attribute – different image sizes in SDP offer and answer

In this example the SDP offer includes H.264/AVC with negotiation of the image size using the "imageattr" attribute.

Table A.4.12ad: Example SDP offer for H.264/AVC with image size negotiation

| SDP offer |
|---|
| <pre> a=tcap:1 RTP/AVPF m=video 49154 RTP/AVP 99 a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA== a=imageattr:99 send [x=320,y=240] recv [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The offer in Table A.4.12ad is identical to that in Table A.4.10a except that the MTSI client offers an image size of 320x240 for both directions.

An example SDP answer to the offer is given below.

Table A.4.12ae: Example SDP answer

| SDP answer |
|--|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:107 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0;profile-level-id=42e00b; \ sprop-parameter-sets=Z0LgC5ZUCg/I,aM4BrFSaA a=imageattr:99 send [x=272,y=224] recv [x=272,y=224] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

In this case, the offering MTSI client should proceed with the session setup without issuing another SDP offer to perform image size negotiation.

A.4.4a.2 H.264/AVC with "imageattr" attribute – different payload type numbers in offer and answer

In this example the SDP offer includes H.264/AVC with negotiation of the image size using the "imageattr" attribute.

Table A.4.12af: Example SDP offer for H.264/AVC with image size negotiation

| SDP offer |
|---|
| <pre> a=tcap:1 RTP/AVPF m=video 49154 RTP/AVP 99 a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=imageattr:99 send [x=320,y=240] recv [x=272,y=224] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The offer in Table A.4.12af is identical to that in Table A.4.10a except that the MTSI client offers image sizes of 320x240 for the send direction and 272x224 for the receive direction.

An example SDP answer to the offer is given below.

Table A.4.12ag: Example SDP answer

| SDP answer |
|--|
| <pre> m=video 49154 RTP/AVPF 101 a=acfg:1 t=1 b=AS:107 b=RS:0 b=RR:2500 a=rtpmap:101 H264/90000 a=fmtp:101 packetization-mode=0;profile-level-id=42e00b; \ sprop-parameter-sets=Z0LgC5ZUCg/I,aM4BrFSAA a=imageattr:101 send [x=272,y=224] recv [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

In this case, the responding MTSI client's answer matches the image sizes in the offer. However, payload type number 99 is not available (e.g., already used for other media) and payload type number 101 is included instead for video media. After the session is established, the offering MTSI client sends the video with payload type number 101 with image size 320x240 towards the responding MTSI client. Similarly, the answering MTSI client sends the video with payload type number 99 with image size 272x224 towards the offering MTSI client.

A.4.4b H.264/AVC with "imageattr" attribute with multiple rtpmaps

In this example the SDP offer includes H.264/AVC with negotiation of the image size using the "imageattr" attribute with multiple rtpmaps for different image sizes.

Table A.4.12ac: Example SDP offer for H.264/AVC with image size negotiation and multiple rtpmaps

| SDP offer |
|--|
| <pre> a=tcap:1 RTP/AVPF m=video 49154 RTP/AVP 99 100 a=pcfg:1 t=1 b=AS:900 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KAHpWgKA9oB/U=,aM46gA== a=imageattr:99 send [x=640,y=480] recv [x=640,y=480] a=rtpmap:100 H264/90000 a=fmtp:100 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=imageattr:100 send [x=320,y=240] recv [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The offered codec is H.264/AVC. The packetization-mode parameter indicates single NAL unit mode. Two image sizes for send and receive directions are offered. The profile-level-id parameter of the first rtpmap indicates Constrained Baseline profile at level 2.2, which supports bitrates up to 4000 kbps. It also indicates, by using so-called constraint-set flags, that the bit stream can be decoded by any Baseline, Main or Extended profile decoder. The bandwidth (including IP, UDP and RTP overhead) for video is 900 kbps. The third parameter, sprop-parameter-sets, includes base-64 encoded sequence and picture parameter set NAL units that are referred by the video bit stream. The sequence parameter set used here includes syntax that specifies the number of re-ordered frames to be zero so that latency can be minimized. sprop-parameter-sets is constructed assuming the offered conditions and image size of 640x480. The profile-level-id parameter of the second rtpmap indicates Constrained Baseline profile at level 1.2, which supports bitrates up to 384kbps. It also indicates, by using so-called constraint-set flags, that the bit stream can be decoded by any Baseline, Main or Extended profile decoder. The third parameter, sprop-parameter-sets, includes base-64 encoded sequence and picture parameter set NAL units that are referred by the video bit stream. The sequence parameter set used here includes syntax that specifies the number of re-ordered frames to be zero so that latency can be minimized. sprop-parameter-sets is constructed assuming the offered conditions and image size of 320x240.

An example SDP answer to the offer is given below.

Table A.4.12b: Example SDP answer

| SDP answer |
|--|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=imageattr:99 send [x=320,y=240] recv [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The responding MTSI client is capable of using H.264/AVC. The responding MTSI client agreed to use a bandwidth of 315 kbps and to use the Constrained Baseline profile at level 1.2. The responding MTSI client included 320x240 for both send and receive directions. Therefore, the final negotiated level is 1.2. The offering MTSI is expecting receiving H.264 RTP packets of resolution 320x240 with payload type number either 99 or 100, while the answering MTSI is expecting receiving H.264 RTP packets of resolution 320x240 with payload type number 99.

A.4.5 H.264 with asymmetric video streams

As described in Clause 5.2.2 (Note 5), when a certain level of H.264 (AVC) is offered using the ‘profile-level-id’ parameter then the video codec is capable of receiving a bitstream up to the offered level and the bandwidth offered with b=AS. This however does not necessarily mean that the H.264 video codec will produce a bitstream up to the offered level when sending video. An MTSI client in terminal can use this method to set up asymmetric video streams. One drawback with using this method is that there is no information in the SDP that resource reservation functions in the network, e.g. RAN bearer allocation, could use to allocate transmission resources asymmetrically for the different directions.

A better method to allocate asymmetric video for H.264 (AVC) is to use the ‘level-asymmetry-allowed’ and the ‘max-recv-level’ parameters defined in the H.264 payload format, [25]. The ‘profile-level-id’ parameter then defines the default level while the ‘max-recv-level’ parameter defines the maximum level for the receiving direction. With this method, there is codec-specific information in the SDP that could be used by resource reservation functions to allocate for example radio bearers in a more optimal way.

The SDP example below shows how a session with asymmetric video can be setup using these SDP parameters. The SDP offer sets the default level to 1.2 (max 384 kbps). The maximum receive level is set to 3.1 (max 14 Mbps) but is limited to 2 Mbps using the b=AS bandwidth modifier. The answerer also declares asymmetric video but using lower levels and bitrates. The default level is set to 1.1 (max 192 kbps) and receiving direction is limited to level 1.2 (max 384 kbps)

Table A.4.13: Example SDP offer and answer for asymmetric video with H.264/AVC

| SDP offer |
|---|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:2000 b=RS:0 b=RR:5000 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00d; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA==; level-asymmetry-allowed=1; \ max-recv-level=e01f a=imageattr:99 send [x=320,y=240] [x=640,y=480] recv [x=320,y=240] [x=640,y=480] [x=1280,y=720] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |
| SDP answer |
| <pre> m=video 49156 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:416 b=RS:0 b=RR:5000 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00b; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA==; level-asymmetry-allowed=1; \ max-recv-level=e00c a=imageattr:99 send [x=640,y=480] recv [x=640,y=480] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

A resource reservation function that only uses the bandwidth information in the b=AS bandwidth modifiers will probably allocate 416 kbps in both directions. This means that an overallocation will occur but this does not present any other problems. The offering MTSI client offers two and three image sizes for both send and receive directions. A larger size, 1280x720, is offered only for the receive direction, which has a higher level.

With the 'profile-level-id', 'level-asymmetry-allowed' and the 'max-recv-level' in the SDP, a codec-aware resource allocation function can take advantage of this information and allocate transmission resources more efficiently, e.g. to allocate 192 kbps from the answerer to the offerer and 384 kbps from the offerer to the answerer. From the image sizes offered, the responding MTSI client included 640x480 for both send and receive directions.

A.4.6 H.264/AVC with "imageattr" attribute for non-CVO operation

In this example the SDP offer includes H.264/AVC with negotiation of the image size using the "imageattr" attribute allowing for orientation compensation in case of non-CVO operation (see clause 6.2.3.3 and clause 7.4.5).

Table A.4.14: Example SDP offer for H.264/AVC with image size negotiation

| SDP offer |
|--|
| <pre> a=tcap:1 RTP/AVPF m=video 49154 RTP/AVP 99 a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtppmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=Z0LADJWgUH6Af1A=, aM46gA==, Z0LADEVoPCmgH9Q=, aEjjqA== a=imageattr:99 send [x=320,y=240] [x=240,y=320] recv [x=320,y=240] [x=240,y=320] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The offered codec is H.264/AVC. The packetization-mode parameter indicates single NAL unit mode. This is the default mode and it is therefore not necessary to include this parameter (see RFC 6184 [25]). The profile-level-id parameter indicates Constrained Baseline profile at level 1.2, which supports bitrates up to 384 kbps. It also indicates, by using so-called constraint-set flags, that the bit stream can be decoded by any Baseline, Main or Extended profile decoder. The bandwidth (including IP, UDP and RTP overhead) for video is 315 kbps. The third parameter, sprop-parameter-sets, includes base-64 encoded sequence and picture parameter set NAL units that are referred by the video bit stream. The sequence parameter set used here includes syntax that specifies the number of re-ordered frames to be zero so that latency can be minimized. sprop-parameter-sets is constructed assuming the offered conditions and image size of 320x240 and 240x320, and contains separate sequence and picture parameter sets with separate ID for both of the supported image resolutions in the send direction. In this example there are two Sequence Parameter Sets (SPS), numbered 0 and 1. SPS 0 has resolution 320x240. SPS 1 has resolution 240x320. There are two Picture Parameter Sets (PPS), numbered 0 and 1. PPS 0 refers to SPS 0. PPS 1 refers to SPS 1. There are four comma-separated and Base64 encoded NAL units as value for sprop-parameter-sets. The order in the example is SPS0,PPS0,SPS1,PPS1.

While the offering MTSI client indicates support for CVO operation, it also offers a couple of image sizes following the format [x,y] and [y,x] for both send and receive directions which allows for signalling of image rotation by a change of resolution in the bitstream in case the receiving MTSI client does not support CVO operation.

An example SDP answer to the offer is given below.

Table A.4.15: Example SDP answer

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtppmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=Z0LADJWgUH6Af1A=, aM46gA==, Z0LADEVoPCmgH9Q=, aEjjqA== a=imageattr:99 send [x=320,y=240] [x=240,y=320] recv [x=320,y=240] [x=240,y=320] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbbr </pre> |

The responding MTSI client is capable of using H.264/AVC. The responding MTSI client agreed to use a bandwidth of 315 kbps and to use the Constrained Baseline profile at level 1.2. The responding MTSI client did not agree CVO operation (removed the extmap attribute) but agreed both offered images sizes 320x240 for both send and receive directions allowing for image rotation compensation in non-CVO operation. The responding MTSI client sent new sprop-parameter-sets for the video encoder of the offering MTSI client, which was constructed assuming the agreed conditions and image sizes.

A.4.7 H.264 (AVC) and H.265 (HEVC)

A.4.7.1 MTSI client with 848x480 resolution 5 inch display

This example SDP offer is for an MTSI client with a 5 inch display that supports 848x480 video and a frame rate of 25 fps. The MTSI client supports H.264 (AVC) Constrained Baseline Profile (CBP) level 3.1. The MTSI client also supports H.265 (HEVC) Main Profile, Main tier level 3.1. When encoding video with H.264 (AVC), the required bandwidth is 690 kbps, including 36 kbps IPv6/UDP/RTP overhead (3 RTP packets per frame), but when H.265 (HEVC) is used, the required bandwidth is only 540 kbps, including 36 kbps overhead (3 RTP packets per frame).

Since the SDP offer includes both codecs, then the b=AS bandwidth must be set to the higher of the bandwidths for those codecs.

Table A.4.16: Example SDP offer for H.264 (AVC) and H.265 (HEVC)

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVP 98 97 100 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:690 b=RS:0 b=RR:5000 a=rtptime:100 H264/90000 a=fmtp:100 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KAHpWgNQ9oB/U=,aM46gA== a=imageattr:100 send [x=848,y=480] recv [x=848,y=480] a=rtptime:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KADZWgUH6Af1A=,aM46gA== a=imageattr:99 send [x=320,y=240] recv [x=320,y=240] a=rtptime:98 H265/90000 a=fmtp:98 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaWAAwBaLAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaWAAwA8oAoIDxZS5LSL0AdwgEE=; \ sprop-pps=RAHAcYDZIA== a=imageattr:98 send [x=848,y=480] recv [x=848,y=480] a=rtptime:97 H265/90000 a=fmtp:97 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaWAAwA8LAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaWAAwA8oAoIDxZS5LSL0AdwgEE=; \ sprop-pps=RAHAcYDZIA== a=imageattr:97 send [x=320,y=240] recv [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The SDP offer includes the image sizes that are supported in sending and receiving directions. It is recommended to provide codec parameter sets for each image size in the SDP offer.

Table A.4.17 shows an example SDP answer where the answerer receives the SDP offer described in Table A.4.16 and accepts using the H.265 (HEVC) codec. The answerer chooses to use the H.265 (HEVC) codec for increased quality and therefore sets the bandwidth to the same value as in the SDP offer.

Table A.4.17: Example SDP answer when H.265 (HEVC) is used to increase the quality

| SDP answer |
|---|
| <pre> m=video 49156 RTP/AVPF 98 a=acfg:1 t=1 b=AS:690 b=RS:0 b=RR:5000 a=rtptime:98 H265/90000 a=fmtp:98 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaWAAwBaLAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaWAAwA8oAoIDxZS5LSL0AdwgEE=; \ sprop-pps=RAHAcYDZIA== a=imageattr:98 send [x=848,y=480] recv [x=848,y=480] a=rtcp-fb:* trr-int 5000 </pre> |

```

a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbbr
a=extmap:4 urn:3gpp:video-orientation

```

The SDP offer in Table A.4.16 and the SDP answer in Table A.4.17 mean that symmetric bandwidths are requested with 690 kbps in both directions.

Table A.4.18 shows another example SDP answer where the answerer receives the SDP offer described in Table A.4.16 and accepts using the H.265 (HEVC) codec. In this case, the answerer chooses to use the H.265 (HEVC) codec to save bandwidth and therefore sets the bit-rate to 540 kbps.

Table A.4.18: Example SDP answer when H.265 (HEVC) is used to reduce the bit-rate

| SDP answer |
|---|
| <pre> m=video 49156 RTP/AVPF 98 a=acfg:1 t=1 b=AS:540 b=RS:0 b=RR:5000 a=rtpmap:98 H265/90000 a=fmtp:98 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBaLAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaAAAwBaoAaiAeFlLktIvQB3CAQQ; \ sprop-pps=RAHAcYDZIA== a=imageattr:98 send [x=848,y=480] recv [x=848,y=480] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The SDP offer in Table A.4.16 and the SDP answer in Table A.4.18 mean that asymmetric bandwidths are requested. The offerer requested to receive 690 kbps while the answerer requested to receive 540 kbps. This discrepancy however can be solved by sending a new SDP offer with only the selected codec.

A.4.7.2 MTSI client with 1280x720 resolution 5 inch display

This example SDP offer is for an MTSI client with a 5 inch display that supports 1280x720 video and a frame rate of 25 fps. The MTSI client supports H.264 (AVC) Constrained Baseline Profile (CBP) level 3.1 and H.265 (HEVC) Main Profile, Main tier level 3.1. When encoding video with H.264 (AVC), the required bandwidth is 950 kbps, including 48 kbps IPv6/UDP/RTP overhead (4 RTP packets per frame), but when H.265 (HEVC) is used, the encoder uses only 640 kbps, including 36 kbps overhead (3 RTP packets per frame).

The answerer is also an MTSI client that supports H.264 (AVC) and H.265 (HEVC) in the same way as the offerer.

Table A.4.19: Example SDP offer for H.264 (AVC) and H.265 (HEVC) and example SDP answer for H.265 (HEVC)

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVPF 98 97 100 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:950 b=RS:0 b=RR:5000 a=rtpmap:100 H264/90000 a=fmtp:100 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KAH5WgFAFugH9Q,aM46gA= a=imageattr:100 send [x=1280,y=720] recv [x=1280,y=720] a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KAHpWgKA9oB/U=,aM46gA= a=imageattr:99 send [x=640,y=480] recv [x=640,y=480] a=rtpmap:98 H265/90000 a=fmtp:98 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBdLAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaAAAwBdoAKAgC0WUuS0i9AHcIBB; \ </pre> |

| |
|--|
| <pre> sprop-pps=RAHAcYDZIA== a=imageattr:98 send [x=1280,y=720] recv [x=1280,y=720] a=rtptime:97 H265/90000 a=fmtp:97 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBaLAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaAAAwBaoAUCAeFlLktIvQB3CAQQ; \ sprop-pps=RAHAcYDZIA== a=imageattr:97 send [x=640,y=480] recv [x=640,y=480] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |
| SDP answer |
| <pre> m=video 49156 RTP/AVPF 98 a=acfg:1 t=1 b=AS:500 b=RS:0 b=RR:5000 a=rtptime:98 H265/90000 a=fmtp:98 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBdLAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaAAAwBdoAKAgCOWUuS0i9AHcIBB; \ sprop-pps=RAHAcYDZIA== a=imageattr:98 send [x=1280,y=720] recv [x=1280,y=720] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The SDP offer includes the image sizes that are supported in sending and receiving directions.

The answerer could also have chosen to use H.265 (HEVC) to improve the quality, similar to what is discussed for Table A.4.17. In this case, the answerer would set the bandwidth to the same value as in the SDP offer.

Another possibility is that the answerer wants to use H.265 (HEVC) partly to increase the quality and partly to reduce the bit-rate. In this case the answerer would select a bit-rate that is in-between the bit-rate in the SDP offer and the bit-rate in the SDP answer as shown in Table A.4.19.

A.4.7.3 MTSI client with 848x480 resolution 10 inch display

This example SDP offer is for an MTSI client with 10 inch display that supports 848x480 video and a frame rate of 25 fps. The MTSI client supports H.264 (AVC) Constrained Baseline Profile (CBP) level 3.1 and H.265 (HEVC) Main Profile, Main tier level 3.1. When encoding video with H.264 (AVC), the required bandwidth is 900 kbps, including 48 kbps IPv6/UDP/RTP overhead (4 RTP packets per frame), but when H.265 (HEVC) is used, the encoder uses only 690 kbps, including 36 kbps of overhead (3 RTP packets per frame).

The answerer is also an MTSI client that supports H.264 (AVC) and H.265 (HEVC) in the same way as the offerer. The answerer chooses to use the H.265 (HEVC) codec to save bandwidth and therefore sets the bit-rate to 690 kbps.

Table A.4.20: Example SDP offer for H.264 (AVC) and H.265 (HEVC) and example SDP answer for H.265 (HEVC)

| |
|---|
| SDP offer |
| <pre> m=video 49154 RTP/AVP 98 97 100 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:900 b=RS:0 b=RR:5000 a=rtptime:100 H264/90000 a=fmtp:100 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=ZOKAHpWgNQ9oB/U=,aM46gA== a=imageattr:100 send [x=848,y=480] recv [x=848,y=480] a=rtptime:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=ZOKADZWgUH6Af1A=,aM46gA== a=imageattr:99 send [x=320,y=240] recv [x=320,y=240] a=rtptime:98 H265/90000 </pre> |

```

a=fmtp:98 profile-id=1; level-id=93; \
  sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBaLAUg; \
  sprop-sps=QgEBAWAAAAMAgAAAaAAAwBaoAaiAeFlLktIvQB3CAQQ; \
  sprop-pps=RAHAcYDZIA==
a=imageattr:98 send [x=848,y=480] recv [x=848,y=480]
a=rtpmap:97 H265/90000
a=fmtp:97 profile-id=1; level-id=93; \
  sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwA8LAUg; \
  sprop-sps=QgEBAWAAAAMAgAAAaAAAwA8oAoIDxZS5LSL0AdwgEE=; \
  sprop-pps=RAHAcYDZIA==
a=imageattr:97 send [x=320,y=240] recv [x=320,y=240]
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation

```

SDP answer

```

m=video 49156 RTP/AVPF 98
a=acfg:1 t=1
b=AS:690
b=RS:0
b=RR:5000
a=rtpmap:98 H265/90000
a=fmtp:98 profile-id=1; level-id=93; \
  sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBaLAUg; \
  sprop-sps=QgEBAWAAAAMAgAAAaAAAwBaoAaiAeFlLktIvQB3CAQQ; \
  sprop-pps=RAHAcYDZIA==
a=imageattr:100 send [x=848,y=480] recv [x=848,y=480]
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation

```

The SDP offer includes the image sizes that are supported in sending and receiving directions.

Similar to the previous examples, the answerer could also have chosen to use H.265 (HEVC) to improve the quality or to use the codec partly to improve the quality and partly to reduce the bit-rate.

A.4.7.4 MTSI client with 1280x720 resolution 10 inch display

This example SDP offer is for an MTSI client with a 10 inch display that supports 1280x720 video and a frame rate of 25 fps. The MTSI client supports H.264 (AVC) Constrained Baseline Profile (CBP) level 3.1 and H.265 (HEVC) Main Profile, Main tier level 3.1. When encoding video with H.264 (AVC), the required bandwidth is 1060 kbps, including 60 kbps IPv6/UDP/RTP overhead (5 RTP packets per frame), but when H.265 (HEVC) is used, the required bandwidth is only 800 kbps, including 48 kbps overhead (4 RTP packets per frame).

The answerer is also an MTSI client that supports H.264 (AVC) and H.265 (HEVC) in the same way as the offerer. The answerer chooses to use the H.265 (HEVC) codec to save bandwidth and therefore sets the bit-rate to 800 kbps.

Table A.4.21: Example SDP offer for H.264 (AVC) and H.265 (HEVC) and example SDP answer for H.265 (HEVC)

| SDP offer |
|---|
| <pre> m=video 49154 RTP/AVP 98 97 100 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:1060 b=RS:0 b=RR:5000 a=rtpmap:100 H264/90000 a=fmtp:100 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KAH5WgFAFugH9Q,aM46gA== a=imageattr:100 send [x=1280,y=720] recv [x=1280,y=720] a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KAHpWgKA9oB/U,aM46gA== a=imageattr:99 send [x=640,y=480] recv [x=640,y=480] a=rtpmap:98 H265/90000 a=fmtp:98 profile-id=1; level-id=93; \ </pre> |

```

sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBdLAUg; \
sprop-sps=QgEBAWAAAAMAgAAAaAAAwBdoAKAgC0WUuS0i9AHcIBB; \
sprop-pps=RAHAcYDZIA==
a=imageattr:98 send [x=1280,y=720] recv [x=1280,y=720]
a=rtptime:97 H265/90000
a=fmtp:97 profile-id=1; level-id=93; \
sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBaLAUg; \
sprop-sps=QgEBAWAAAAMAgAAAaAAAwBaoAUCAeFlLktIvQB3CAQQ; \
sprop-pps=RAHAcYDZIA==
a=imageattr:97 send [x=640,y=480] recv [x=640,y=480]
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation

```

SDP answer

```

m=video 49156 RTP/AVPF 100
a=acfg:1 t=1
b=AS:800
b=RS:0
b=RR:5000
a=rtptime:98 H265/90000
a=fmtp:98 profile-id=1; level-id=93; \
sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBdLAUg; \
sprop-sps=QgEBAWAAAAMAgAAAaAAAwBdoAKAgC0WUuS0i9AHcIBB; \
sprop-pps=RAHAcYDZIA==
a=imageattr:98 send [x=1280,y=720] recv [x=1280,y=720]
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation

```

The SDP offer includes the image sizes that are supported in sending and receiving directions.

Similar to the previous examples, the answerer could also have chosen to use H.265 (HEVC) to improve the quality or to use the codec partly to improve the quality and partly to reduce the bit-rate.

A.4.8 H.264 (AVC) and H.265 (HEVC) with asymmetric video streams

This example SDP offer shows how an asymmetric video session can be set up. The SDP offer is based on the example SDP offer shown in Annex A.4.7.4 (10 inch display, 1280x720 resolution) with modifications to allow for setting up an asymmetric session where the receive level is higher than the default level. The following video encoding and decoding capabilities apply:

- For H.264 (AVC):
 - The Constrained Baseline Profile (CBP) is used.
 - The default level is 1.2, max 384 kbps, as shown with 'profile-level-id=42e00c'. This is then used for the maximum level in the sending direction if H.264 (AVC) is accepted by the answerer.
 - The maximum level in the receiving direction is 3.1, as shown with 'max-recv-level=e01f'.
 - Asymmetric session is allowed as shown with 'level-asymmetry-allowed=1'.
 - The maximum bitrate in the receiving direction is limited to 1060 kbps with 'b=AS:1060'.
- For H.265 (HEVC):
 - The Main Profile is used.
 - The default level is 1.0, max 128 kbps, as shown with 'level-id=30'. This is then used for the maximum level in the sending direction if H.265 (HEVC) is accepted by the answerer.
 - The maximum level in the receiving direction is 3.1, as shown with 'max-recv-level-id=93'.

- Asymmetric session is allowed as shown by including the 'max-recv-level-id' parameter.
- The offerer would like to receive max 800 kbps if H.265 (HEVC) is accepted but there is no possibility to indicate this in the SDP offer.

Table A.4.22: Example SDP offer and answer for asymmetric video with H.264 (AVC) and H.265 (HEVC)

| SDP offer |
|---|
| <pre> m=video 49154 RTP/AVP 98 97 100 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:1060 b=RS:0 b=RR:5000 a=rtmpmap:100 H264/90000 a=fmtp:100 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA=;; \ level-asymmetry-allowed=1; max-recv-level=e01f a=imageattr:100 send [x=320,y=240] recv [x=1280,y=720] [x=320,y=240] a=rtmpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA=;; \ level-asymmetry-allowed=1; max-recv-level=e01f a=imageattr:99 send [x=320,y=240] recv [x=640,y=480] [x=320,y=240] a=rtmpmap:98 H265/90000 a=fmtp:98 profile-id=1; level-id=30; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwA8LAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaAAAwA8oAoIDxZS5LSL0AdwgEE=; \ sprop-pps=RAHAcYDZIA=;; \ max-recv-level-id=93 a=imageattr:98 send [x=320,y=240] recv [x=1280,y=720] [x=320,y=240] a=rtmpmap:97 H265/90000 a=fmtp:97 profile-id=1; level-id=30; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwA8LAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaAAAwA8oAoIDxZS5LSL0AdwgEE=; \ sprop-pps=RAHAcYDZIA=;; \ max-recv-level-id=90 a=imageattr:97 send [x=320,y=240] recv [x=640,y=480] [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation </pre> |

The SDP offer includes the image sizes that are supported in sending and receiving directions. Different resolutions are offered by including two RTP payload types for H.264 (AVC) and H.265 (HEVC), respectively.

For PT=98, the MTSI client in terminal specifies max-recv-level-id=93 since this is needed for 1280x720 resolution. But for PT=97, it specifies max-recv-level-id=90 since this is sufficient for 640x480 resolution.

A.5 SDP offers for text

A.5.1 T.140 with and without redundancy

An offer to use T.140 real-time text may be realized by using SDP according to the following example in session setup or for addition of real-time text during a session.

Table A.5.1: Example SDP offer for T.140 real-time text

| SDP offer |
|--|
| m=text 53490 RTP/AVP 100 98 b=AS:2 b=RS:0 b=RR:500 a=rtpmap:100 red/1000/1 a=rtpmap:98 t140/1000/1 a=fmtp:100 98/98/98 |

The example in table A.5.1 shows that RTP payload type 98 is used for sending text without redundancy, whereas RTP payload type 100 is used for sending text with 200 % redundancy. IPv4 addressing is assumed in the computation of bandwidth values.

A.6 SDP example with bandwidth information

A.6.1 General

This clause gives an example where the bandwidth modifiers have been included in the SDP offer. Clause A.6.2 gives some examples with the b=AS, b=RS and b=RR bandwidth modifiers. Clause A.6.3 gives some examples where the SDP are enhanced with the 'a=bw-info' attribute defined in clause 19.

A.6.2 SDP examples with bandwidth information declared with bandwidth modifiers

Table A.6.1: SDP example with bandwidth information

| SDP offer |
|---|
| <pre>v=0 o=Example_SERVER 3413526809 0 IN IP4 server.example.com s=Example of using AS in MTSI c=IN IP4 aaa.bbb.ccc.ddd b=AS:345 t=0 0 a=tcap:1 RTP/AVPF m=audio 49152 RTP/AVP 97 98 a=pcfg:1 t=1 b=AS:30 b=RS:0 b=RR:4000 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240 m=video 49154 RTP/AVP 99 a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:5000 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation</pre> |

The b=AS value indicates the media bandwidth, excluding RTCP, see RFC 3550, section 6.2. On session level, the b=AS value indicates the sum of the media bandwidths, excluding RTCP.

In this example, the bandwidth for RTCP is allocated such that it allows for sending at least 2 compound RTCP packets per second when AVPF immediate mode is used. The size of a RTCP Sender Report is estimated to 110 bytes, given

IPv4 and point-to-point sessions. The corresponding bandwidth then becomes 1760 bps which means that compound RTCP packets can be sent a little more frequently than twice per second.

For speech sessions, the total RTCP bandwidth is set to 4000 bps (2000 bps for each terminal) to give room for adaptation requests with APP packets according to clause 10.2 in at least some of the RTCP messages. This adds 16 bytes to the RTCP packet.

The `b=AS` of AMR, 30, is set in the media level as the larger of the `b=AS` for bandwidth-efficient payload format, 29, and the `b=AS` for octet-aligned payload format, 30, with IPv4.

For video, the total RTCP bandwidth is set to 5000 bps (2500 bps for each terminal) to give room for slightly more frequent reporting and also to give room for codec-control messages (CCM) [43].

Setting the `RS` value to 0 does not mean that senders are not allowed to send RTCP packets. It instead means that sending clients are treated in the same way as receive-only clients, see also RFC 3556 [42].

The `tcap` attribute is in this example given on the session level to avoid repeating it for each media type.

A.6.3 SDP examples giving additional bandwidth information using the `a=bw-info` attribute

The SDP offer in Table A.6.2 below shows an SDP offer for speech where the MTSI client in terminal include the '`a=bw-info`' attribute to signal some additional bandwidth properties. This example is based on the SDP offer in Table A.6.1 (only the audio part) with some changes, as described below.

Both AMR and AMR-WB are offered with all codec modes and also with both bandwidth-efficient and octet-aligned payload format. The MTSI client in terminal includes the '`a=bw-info`' attribute both to show that this attribute is supported and to suggest suitable bandwidth properties. The assumptions used for the SDP offer follow the recommendations in clause 6.2.5.2, with the following description and a few changes:

- Bandwidth properties for both IPv4 and IPv6 are included. However, the `b=AS` bandwidth assumes IPv4.
- The Maximum Supported Bandwidth property suggests that 100% redundancy should be possible. The resource allocation in the networks may reduce this, if needed.
 - The originating MTSI client must set the Maximum Supported Bandwidth and the Maximum Desired Bandwidth properties for AMR-WB to higher values than what is described in clause 6.2.5.2 Table 6.10-2 because it is required to offer all codec modes as described in clause 6.2.2.2 Tables 6.1 and 6.2. This also means that the `b=AS` bandwidth needs to be set to a higher value. A network in the path may however change the offered mode-sets, in which case it should also modify these bandwidth properties and the `b=AS` bandwidth.
 - The originating MTSI client can also set the Maximum Supported Bandwidth and the Maximum Desired Bandwidth properties to the same value since offering the full mode sets means that redundancy can be used for both AMR and AMR-WB without allocating any additional bandwidth for this purpose. If a network changes the offset mode-sets, e.g. to AMR-WB {6.6, 8.85, 12.65} then it should modify these bandwidth properties correspondingly, and the Maximum Supported Bandwidth property may then show a higher value than the Maximum Desired Bandwidth.
- The Maximum Desired Bandwidth property is based on the maximum codec rates without redundancy, i.e. AMR 12.2 and AMR 23.85, respectively.
 - If a network changes the offered mode-sets then this bandwidth property should be modified correspondingly.
- The Minimum Desired Bandwidth properties are based on AMR 5.9 and AMR-WB 6.6, respectively, without redundancy while still packetizing 1 frame per packet, as described in clause 6.2.5.2.
- The Minimum Supported Bandwidth properties are based on AMR 4.75 and AMR-WB 6.6, respectively, without redundancy while packing 4 frames per packet, as described in clause 6.2.5.2.

Table A.6.2: SDP example for speech with AMR and AMR-WB with additional bandwidth information signalled with the 'a=bw-info' attribute

| SDP offer |
|---|
| <pre> v=0 o=Example_SERVER 3413526809 0 IN IP4 server.example.com s=Example of using bw-info attribute in MTSI c=IN IP4 aaa.bbb.ccc.ddd b=AS:41 t=0 0 a=tcap:1 RTP/AVPF m=audio 49152 RTP/AVP 99 100 97 98 a=pcfg:1 t=1 b=AS:41 b=RS:0 b=RR:4000 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=bw-info:97 sendrecv IpVer=4; MaxSupBw=29; MaxDesBw=29 a=bw-info:97 sendrecv IpVer=6; MaxSupBw=37; MaxDesBw=37 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=bw-info:98 sendrecv IpVer=4; MaxSupBw=30; MaxDesBw=30 a=bw-info:98 sendrecv IpVer=6; MaxSupBw=38; MaxDesBw=38 a=bw-info:97,98 sendrecv IpVer=4; MinDesBw=23; MinSupBw=10 a=bw-info:97,98 sendrecv IpVer=6; MinDesBw=31; MinSupBw=12 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=bw-info:99 sendrecv IpVer=4; MaxSupBw=41; MaxDesBw=41; MinDesBw=24; MinSupBw=11 a=bw-info:99 sendrecv IpVer=6; MaxSupBw=49; MaxDesBw=49; MinDesBw=32; MinSupBw=13 a=rtpmap:100 AMR-WB/16000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=bw-info:100 sendrecv IpVer=4; MaxSupBw=41; MaxDesBw=41; MinDesBw=24; MinSupBw=12 a=bw-info:100 sendrecv IpVer=6; MaxSupBw=49; MaxDesBw=49; MinDesBw=32; MinSupBw=14 a=bw-info:* sendrecv MaxPRate=50; MinPRate=12.5 a=ptime:20 a=maxptime:240 </pre> |

Comments:

The most relevant lines are highlighted with bold font.

Since the SDP offer describes that the session should be symmetric, the 'sendrecv' directionality is used to reduce the size of the SDP. It would also have been possible to include separate 'a=bw-info' attribute lines with 'send' and 'recv', respectively.

The MinDesBw and MinSupBw bandwidths are the same for AMR with bandwidth-efficient and octet-aligned payload format, which allows for defining them on one 'a=bw-info' attribute line for both RTP payload types. However, since the MaxSupBw and MaxDesBw are different for bandwidth-efficient and octet-aligned payload format these bandwidth properties need to be defined on separate attribute lines.

The maximum packet rate and minimum packet rate are the same for all payload types and are therefore defined on separate attribute lines to allow for using a wildcard. It would also be possible to have one attribute line for each RTP payload type for these properties but this would in this case be just a waste of bytes.

The SDP offer in Table A.6.3 below shows an SDP offer for video where the MTSI client in terminal include the 'a=bw-info' attribute to signal some additional bandwidth properties. This example is based on the SDP offer in Table

6.1 (only the video part) with some changes, as described below. Bandwidth properties for both IPv4 and IPv6 are included. The conversion factor between IPv4 and IPv6 bandwidths defined in clause 12.7.5 for the b=AS parameter is used also for the additional bandwidth properties.

The MTSI client in terminal supports H.264 Constrained Baseline Profile (CBP) with level 3.1 but video is offered with asymmetric video streams, max 1 Mbps for the sending direction and max 2 Mbps for the receiving direction. The maximum bandwidth in the receiving direction can be limited to 2 Mbps with the b=AS bandwidth modifier. However, the b=AS bandwidth modifier cannot be used to describe the limitation for the sending direction. The MTSI client in terminal therefore includes the 'a=bw-info' attribute to able to describe the limitation also for the sending direction. Since the Maximum Supported Bandwidth and Maximum Desired Bandwidth properties are different for different the sending and receiving directions it is necessary to describe them using two "a=bw-info" attribute lines with directions 'send' and 'recv', respectively.

The MTSI client in terminal also proposes that the Minimum Desired Bandwidth and the Minimum Supported bandwidth based on the QoS examples in Annex E.21 for IPv4 (202 kbps) and in Annex E.22 for IPv6 (208 kbps). These bandwidth properties are the same for both sending and receiving directions, which means the number of lines can be reduced by using the 'sendrecv' directionality.

Including the 'a=bw-info' also shows to the networks and the remote end-point that this attribute is supported and that other bandwidth modifiers may be added, if needed, even though they are not included in the SDP offer.

Table A.6.3: SDP example for video with H.264 with additional bandwidth information signalled with the 'a=bw-info' attribute

| SDP offer |
|---|
| <pre> v=0 o=Example_SERVER 3413526809 0 IN IP4 server.example.com s=Example of using bw-info attribute in MTSI c=IN IP4 aaa.bbb.ccc.ddd b=AS:2000 t=0 0 a=tcap:1 RTP/AVPF m=video 49154 RTP/AVP 99 a=pcfg:1 t=1 b=AS:2000 b=RS:0 b=RR:5000 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets= Z0KAHpWgNQ9oB/U=,aM46gA= a=bw-info:99 send IpVer=4; MaxSupBw=1000; MaxDesBw=1000 a=bw-info:99 recv IpVer=4; MaxSupBw=2000; MaxDesBw=2000 a=bw-info:99 send IpVer=6; MaxSupBw=1040; MaxDesBw=1040 a=bw-info:99 recv IpVer=6; MaxSupBw=2080; MaxDesBw=2080 a=bw-info:99 sendrecv IpVer=4; MinDesBw=202; MinSupBw=202 a=bw-info:99 sendrecv IpVer=6; MinDesBw=208; MinSupBw=208 a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbbr a=extmap:4 urn:3gpp:video-orientation </pre> |

Comments:

The most relevant lines are highlighted with bold font.

The MTSI client in terminal only describes the Maximum Supported Bandwidth and the Maximum Desired Bandwidth properties.

The Minimum Desired Bandwidth and the Minimum Supported Bandwidth are left undefined since the MTSI client in terminal can reduce the bitrate virtually down to zero by reducing the frame rate. Since the 'a=bw-info' attribute is included in the SDP, the network knows that this attribute is supported and it can then set these bandwidth properties based on the bearer allocation and/or operator policies.

The MTSI client in terminal also does not define the maximum and minimum packet rates. The network can still add these bandwidth properties if needed.

A.7 SDP examples with "3gpp_sync_info" attribute

A.7.1 Synchronized streams

In the example given below in table A.7.1, streams identified with "mid" attribute 1 and 2 are to be synchronized (default operation if the "3gpp_sync_info" attribute is absent).

Table A.7.1: SDP example with requirement on synchronization

| SDP offer |
|--|
| <pre>v=0 o=Laura 289083124 289083124 IN IP4 one.example.com s=Demo c=IN IP4 224.2.17.12/127 t=0 0 a=group:LS 1 2 a=3gpp_sync_info:Sync a=tcap:1 RTP/AVPF m=audio 30000 RTP/AVP 0 a=pcfg:1 t=1 a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=mid:1 m=video 30002 RTP/AVP 31 a=pcfg:1 t=1 a=mid:2 m=audio 30004 RTP/AVP 2 a=pcfg:1 t=1 i=This media stream contains the Spanish translation a=mid:3</pre> |

A.7.2 Nonsynchronized streams

The SDP in table A.7.2 gives an example of the usage of "3gpp_sync_info" attribute at media level. In this example, there are two H.264 video streams where the first video stream, using port number 6000, should not be synchronized with any other media stream in the session.

Table A.7.2: SDP example with no requirement on synchronization

| SDP offer |
|----------------|
| <pre>v=0</pre> |

```

o=Laura 289084412 2890841235 IN IP4 123.124.125.1
s=Demo
c=IN IP4 123.124.125.1
t=0 0
a=tcap:1 RTP/AVPF
m=video 6000 RTP/AVP 98
a=pcfg:1 t=1
a=rtpmap:98 H264/90000
a=fmtp:98 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA==
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=3gpp_sync_info:No Sync
a=extmap:4 urn:3gpp:video-orientation
m=video 5000 RTP/AVP 99
a=pcfg:1 t=1
a=rtpmap 99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA==
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation
m=audio 7000 RTP/AVP 100
a=pcfg:1 t=1
a=rtpmap:100 AMR/8000/1
a=fmtp:97 mode-change-capability=2; max-red=220
a=rtpmap:98 AMR/8000/1
a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1
a=ptime:20
a=maxptime:240

```

A.8 SDP example with QoS negotiation

This clause gives an example of an SDP interchange with negotiated QoS parameters.

Table A.8.1: SDP example with QoS negotiation

| SDP offer from MTSI client in terminal A to B in SIP INVITE message |
|---|
| <pre> v=0 o=Example_SERVER 3413526809 0 IN IP4 server.example.com s=Example of using AS to indicate negotiated QoS in MTSI c=IN IP4 aaa.bbb.ccc.ddd b=AS:345 t=0 0 a=tcap:1 RTP/AVPF m=audio 49152 RTP/AVP 97 98 </pre> |

```
a=pcfg:1 t=1
b=AS:30
b=RS:0
b=RR:2000
a=rtpmap:97 AMR/8000/1
a=fmtp:97 mode-change-capability=2; max-red=220
a=rtpmap:98 AMR/8000/1
a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1
aptime:20
a=maxptime:240
m=video 49154 RTP/AVP 99
a=pcfg:1 t=1
b=AS:315
b=RS:0
b=RR:2500
a=rtpmap:99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA==
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation
```


SDP answer from UE B to A in 200/OK message

```
v=0
o=Example_SERVER2 34135268010 IN IP4 server2.example.com
s=Example of using AS to indicate negotiated QoS in MTSI
c=IN IP4 aaa.bbb.ccc.ddd
b=AS:344
t=0 0
m=audio 49152 RTP/AVPF 97
a=pcfg:1 t=1
b=AS:29
b=RS:0
b=RR:2000
a=rtpmap:97 AMR/8000/1
a=fmtp:97 mode-change-capability=2; max-red=220
a=ptime:20
a=maxptime:240
m=video 49154 RTP/AVPF 99
a=acfg:1 t=1
b=AS:315
b=RS:0
b=RR:2500
a=rtpmap:99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA==
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation
```

SDP offer from MTSI client in terminal B to A in SIP UPDATE message

```

v=0
o=Example_SERVER2 34135268010 IN IP4 server2.example.com
s=Example of using AS to indicate negotiated QoS in MTSI
c=IN IP4 aaa.bbb.ccc.ddd
b=AS:59
t=0 0
m=audio 49252 RTP/AVPF 97
b=AS:29
b=RS:0
b=RR:2000
a=rtpmap:97 AMR/8000/1
a=fmtp:97 mode-change-capability=2; max-red=220
a=ptime:20
a=maxptime:240
m=video 49254 RTP/AVPF 99
b=AS:30
b=RS:0
b=RR:2500
a=rtpmap:99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=Z0LgC5ZUCg/I,aM4BrFSaA
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation

```

SDP answer from MTSI client in terminal A to B in 200/OK RESPONSE to UPDATE message

```

v=0
o=Example_SERVER 3413526809 0 IN IP4 server.example.com
s=Example of using AS to indicate negotiated QoS in MTSI
c=IN IP4 aaa.bbb.ccc.ddd
b=AS:77
t=0 0
m=audio 49152 RTP/AVPF 97
b=AS:29
b=RS:0
b=RR:2000
a=rtpmap:97 AMR/8000/1
a=fmtp:97 mode-change-capability=2; max-red=220
a=ptime:20
a=maxptime:240
m=video 49154 RTP/AVPF 99
b=AS:48
b=RS:0
b=RR:2500
a=rtpmap:99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=Z0LgC5ZUCg/I,aM4BrFSaA
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr

```

```
a=extmap:4 urn:3gpp:video-orientation
```

The example in table A.8.1 shows an SDP exchange that reflects the signalling of negotiated QoS during initial session setup when there is only one PDP context or EPS bearer for the whole session. The first offer-answer procedure is initiated by the MTSI client in terminal A at session setup. The responding MTSI client chose the bandwidth-efficient payload format, by excluding the octet-align parameter, and reduced the bandwidth in b=AS to 29. The second offer-answer procedure is initiated by the MTSI client in terminal B when it receives a different negotiated QoS, only 30 kbps for video, than what was indicated in the first SDP offer from A. To notify A, B sends a new SDP offer, in this case embedded in an UPDATE message, to A indicating the lower negotiated QoS bit rate. The MTSI client in terminal A responds with its negotiated QoS value to B.

NOTE: The bit rate in the second SDP answer, 48 kbps, was deliberately chosen to show that this is a fully valid SDP answer even though the second SDP offer only defines 30 kbps. It is however recommended that the UEs choose the same bandwidths whenever possible.

The SDP offer in the SIP UPDATE message contains only one encoding format since the answerer has already removed all but one encoding format in the SDP answer to the initial SDP offer.

In this example it is assumed that the SDPCapNeg framework is not needed in the UPDATE since the RTP profile has already been chosen in the initial invitation.

A.9 Void

A.9a SDP offer/answer regarding the use of Reduced-Size RTCP

This example shows the offers and answers for a session between two MTSI clients controlling the use of Reduced-Size RTCP.

Table A.9a.1: SDP example for Reduced-Size RTCP

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtcp-fb:* trr-int 5000 a=rtcp-rsize a=rtptime:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtptime:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 a=rtcp-fb:* trr-int 5000 a=rtcp-rsize a=rtptime:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240</pre> |

Comments:

This example allows the use of Reduced-Size RTCP (attribute `rtcp-rsize`) for the adaptation feedback. Moreover the minimum interval between two regular compound RTCP packets is set to 5000 milliseconds.

A.10 Examples of SDP offers and answers for inter-working with other IMS or non-IMS IP networks

A.10.1 General

The session between an MTSI client in a terminal and a client in a remote (IMS or non-IMS) network can be established in many ways, especially when the session is initiated by the remote network and when the remote network does not use the MTSI service.

The SDP will also depend on how and when the MTSI MGW chooses to add information about what formats (other than AMR and AMR-WB) that can be used for inter-working. There are, in general, two methods for MTSI MGWs to add information about the alternative formats. The first method is to add the alternative formats to the original SDP offer from the initiating client as a pre-emptive action. The second method is to leave the original SDP offer unchanged, forward it to the remote network and wait for the answerer to respond and only add the alternative formats if/when the SDP offer was rejected. A further complication is that there might be multiple MGWs in the path for this kind of inter-working and different MGWs might work differently.

The SDP examples included below should therefore be regarded as a few samples of possible SDPs and should not be regarded as a complete description of what might occur in a real implementation.

A.10.2 Session initiated by MTSI client in terminal

A.10.2.1 SDP offers from an MTSI client in terminal

The MTSI client in terminal could send an SDP offer as shown in Table A.10.1 (narrow-band speech only) or Table A.10.2 (wide-band and narrow-band speech).

Table A.10.1: Original SDP offer from an MTSI client in terminal for narrow-band speech

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240</pre> |

Comments:

This SDP offer is identical to the SDP offer in Table A.1.1.

Table A.10.2: Original SDP offer from an MTSI client in terminal for narrow-band and wide-band speech

| SDP offer |
|--|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240 </pre> |

Comments:

This SDP offer is identical to the SDP offer in Table A.1.2.

A.10.2.2 SDP offers modified by MTSI MGW when pre-emptively adding inter-working formats

In this example, the MTSI MGW intercepts the SIP INVITE with the original SDP offer from the MTSI client in terminal and adds the codecs and formats that are supported for inter-working before forwarding the SDP offer to the remote network.

When an MTSI MGW pre-emptively adds codecs and formats for inter-working it will also remove lines that it does not support. These examples show an MTSI MGW that does not support AVPF nor SDPCapNeg and it will therefore remove the corresponding lines. The SDP offers could look like the examples included in Table A.10.3 (narrow-band speech only) and Table A.10.4 (wide-band and narrow-band speech).

Table A.10.3: SDP offer for narrow-band speech which has been modified by the MTSI MGW before it is sent to the remote network

| Modified SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 PCMA/8000/1 a=rtpmap:100 PCMU/8000/1 a=rtpmap:101 L16/8000/1 a=ptime:20 a=maxptime:240 </pre> |

Comments:

The SDP offer from Table A.10.1 has been modified by adding RTP Payload Types 99 (A-law PCM), 100 (μ -law PCM) and 101 (linear 16 bit PCM with 8 kHz sampling frequency).

The lines "a=tcap:1 RTP/AVPF" and "a=pcfg:1 t=1" are removed because the MTSI MGW does not support AVPF nor SDPCapNeg in this example.

To allow for end-to-end adaptation for AMR and AMR-WB, the MTSI MGW keeps `a=maxptime:240`.

If the remote network supports AMR, then the received SDP answer should contain at least one RTP Payload Type for AMR but there may also be one or more RTP Payload types for non-AMR codecs. In this case, the MTSI MGW does not need to perform transcoding and may forward the SDP offer to the MTSI client in terminal unchanged.

If the SDP answer contains no AMR RTP Payload Type then the MTSI MGW needs to perform transcoding to and from the format indicated by the remote network. In this case, the MTSI MGW needs to add AMR to the SDP answer that is sent back to the MTSI client in terminal.

Table A.10.4: SDP offer for wide-band and narrow-band speech which has been modified by the MTSI MGW before it is sent to the remote network

| Modified SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 101 102 99 100 103 104 105 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:101 G722/8000/1 a=rtpmap:102 L16/16000/1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:103 PCMA/8000/1 a=rtpmap:104 PCMU/8000/1 a=rtpmap:105 L16/8000/1 a=ptime:20 a=maxptime:240 </pre> |

Comments:

The SDP offer from Table A.10.2 has been modified by adding RTP Payload Types 101 (G.722), 102 (linear 16 bit PCM with 16 kHz sampling frequency), 103 (A-law PCM), 104 (μ -law PCM) and 105 (linear 16 bit PCM with 8 kHz sampling frequency).

NOTE: The sampling frequency for G.722 is 16 kHz but has been set to 8 kHz in the SDP because G.722 was (erroneously) assigned this value in the original version of the RTP A/V profile. Hence, one need to use "8000" for backwards compatibility reasons, see also [10].

The lines "`a=tcap:1 RTP/AVPF`" and "`a=pcfg:1 t=1`" are removed because the MTSI MGW does not support SDPCapNeg (in this example).

To allow for end-to-end adaptation for AMR and AMR-WB, the MTSI MGW keeps `a=maxptime:240`.

If the remote network supports AMR-WB or AMR, then the received SDP answer should contain at least one RTP Payload Type for AMR-WB or AMR but there may also be one or more RTP Payload Types for non-AMR codecs. In this case, the MTSI MGW does not need to perform transcoding and can remove the non-AMR RTP Payload Types before forwarding the SDP answer to the MTSI client in terminal.

If the SDP answer contains no AMR-WB or AMR RTP Payload Type then the MTSI MGW needs to perform transcoding to and from the format indicated by the remote network.

A.10.2.3 SDP modified by MGW when adding inter-working formats only when the original SDP offer was rejected

In this example, the MTSI MGW either forwards the original SDP offer that was received from the MTSI client in terminal to the remote network or it is not involved in the session setup at all until it is concluded that the same codecs

are not supported in the different networks. In this latter case, the MTSI MGW is invoked only if the remote network rejects the SDP offer.

In this case, when the MTSI MGW sends the (new) SDP offer to the remote network it knows that the AMR (and AMR-WB) codecs are not supported by the remote network because the original SDP offer was rejected. It is therefore unnecessary to include these codecs in the (new) SDP offer. The SDP offers could look like the examples included in Table A.10.5 (narrow-band speech only) and Table A.10.6 (wide-band and narrow-band speech).

The remote client may also suggest codecs and configurations when it rejects the SDP offer. Existence of such information can, of course, be used to increase the likelihood that the session setup will be successful. These SDP examples are however designed for the case when no such information is available from the remote network.

Table A.10.5: New SDP offer for narrow-band speech sent by the MTSI MGW to the remote network

| New SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 99 100 101 a=rtpmap:99 PCMA/8000/1 a=rtpmap:100 PCMU/8000/1 a=rtpmap:101 L16/8000/1 a=ptime:20 a=maxptime:80</pre> |

Comments:

The new SDP offer includes RTP Payload Types 99 (A-law PCM), 100 (μ -law PCM) and 101 (linear 16 bit PCM with 8 kHz sampling frequency).

In this case, the `maxptime` is set to 80, if the MTSI MGW does not support redundancy.

Table A.10.6: New SDP offer for narrow-band and wide-band speech sent by the MTSI MGW to the remote network

| New SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 101 102 103 104 105 a=rtpmap:101 G722/8000/1 a=rtpmap:102 L16/16000/1 a=rtpmap:103 PCMA/8000/1 a=rtpmap:104 PCMU/8000/1 a=rtpmap:105 L16/8000/1 a=ptime:20 a=maxptime:80</pre> |

Comments:

The new SDP offer includes RTP Payload Types 101 (G.722), 102 (linear 16 bit PCM with 16 kHz sampling frequency), 103 (A-law PCM), 104 (μ -law PCM) and 105 (linear 16 bit PCM with 8 kHz sampling frequency).

NOTE: The sampling frequency for G.722 is 16 kHz but has been set to 8 kHz in the SDP because G.722 was (erroneously) assigned this value in the original version of the RTP A/V profile. Hence, one need to use "8000" for backwards compatibility reasons, see also [10].

In this case, the `maxptime` is set to 80, if the MTSI MGW does not support redundancy.

A.11 Adding or removing a video component to/from an on-going video call session

The MTSI client in a terminal can add, remove and modify the media components during an ongoing MTSI session. This clause describes the SDP offer in the initial SIP INVITE message, see Table A.11.1, and the SDP in the subsequent re-INVITE or UPDATE message for adding and removing a video stream to/from the ongoing MTSI video call session, see Table A.11.2 and Table A.11.3, respectively. Corresponding SDP answers in the SIP 200/OK responses are also described.

The initial video call session contains one video component and one speech component. During the session, the MTSI client in terminal A adds a uni-directional video component (such as one video clip) to the ongoing video call session. The SDP content attribute "a=content:main" and "a=content:alt" are used to label the main and alternative video components respectively [81].

This example does not show how to use the content attribute in combination with the grouping attribute, nor does it show how to use the content attribute in combination with the synchronization attribute defined in Clause 6.2.6.

Table A.11.1: SDP offer/answer for setting up a video telephony session

| SDP offer from MTSI client in terminal A to B in SIP INVITE message |
|--|
| <pre> a=tcap:1 RTP/AVPF m=audio 49150 RTP/AVP 96 a=pcfg:1 t=1 b=AS:29 b=RS:0 b=RR:2000 a=rtpmap:96 AMR/8000/1 a=fmtp:96 mode-change-capability=2; max-red=220 aptime:20 a=maxptime:240 m=video 54320 RTP/AVP 99 a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbbr a=extmap:4 urn:3gpp:video-orientation </pre> |
| SDP answer from MTSI client in terminal B to A in 200/OK RESPONSE message |
| <pre> m=audio 49152 RTP/AVPF 96 a=acfg:1 t=1 b=AS:29 b=RS:0 b=RR:2000 a=rtpmap:96 AMR/8000/1 a=fmtp:96 mode-change-capability=2; max-red=220 aptime:20 a=maxptime:240 m=video 54320 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbbr a=extmap:4 urn:3gpp:video-orientation </pre> |

Table A.11.2: Second SDP offer/answer for adding one more video component

| SDP offer from MTSI client in terminal A to B in SIP UPDATE/Re-INVITE message |
|--|
| <pre> a=tcap:1 RTP/AVPF m=audio 49150 RTP/AVP 96 a=pcfg:1 t=1 b=AS:29 b=RS:0 b=RR:2000 a=rtpmap:96 AMR/8000/1 a=fmtp:96 mode-change-capability=2; max-red=220 aptime:20 a=maxptime:240 m=video 54320 RTP/AVP 99 i=Main video a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=content:main a=extmap:4 urn:3gpp:video-orientation m=video 43200 RTP/AVP 100 i=Alternative video a=pcfg:1 t=1 b=AS:128 b=RS:0 b=RR:2500 a=rtpmap:100 H264/90000 a=fmtp:100 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=content:alt a=sendonly a=extmap:4 urn:3gpp:video-orientation </pre> |
| SDP answer from MTSI client in terminal B to A in 200/OK RESPONSE to UPDATE/Re-INVITE message |
| <pre> m=audio 49152 RTP/AVPF 96 a=acfg:1 t=1 b=AS:29 b=RS:0 b=RR:2000 a=rtpmap:96 AMR/8000/1 a=fmtp:96 mode-change-capability=2; max-red=220 aptime:20 a=maxptime:240 m=video 54320 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 </pre> |

```

b=RS:0
b=RR:2500
a=rtpmap:99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA==
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=content:main
a=extmap:4 urn:3gpp:video-orientation
m=video 43200 RTP/AVPF 100
a=acfg:1 t=1
b=AS:128
b=RS:0
b=RR:2500
a=rtpmap:100 H264/90000
a=fmtp:100 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA==
a=content:alt
a=recvonly
a=extmap:4 urn:3gpp:video-orientation

```

Table A.11.3: Second SDP offer/answer for removing the video component

| SDP offer from MTSI client in terminal A to B in SIP INVITE message |
|---|
| <pre> m=audio 49150 RTP/AVPF 96 b=AS:29 b=RS:0 b=RR:2000 a=rtpmap:96 AMR/8000/1 a=fmtp:96 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 m=video 0 RTP/AVP 99 </pre> |
| SDP answer from MTSI client in terminal B to A in 200/OK RESPONSE message |
| <pre> m=audio 49152 RTP/AVPF 96 b=AS:29 b=RS:0 b=RR:2000 a=rtpmap:96 AMR/8000/1 a=fmtp:96 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 m=video 0 RTP/AVP 99 </pre> |

A.12 SDP examples when using ECN

A.12.1 SDP examples when using ECN for speech

A.12.1.1 With RTP/AVP and zero RTCP bandwidth

The following SDP offer and SDP answer are likely when both MTSI clients in terminals use ECN for speech.

This SDP example is based on the SDP example found in Table A.3.0 except that bandwidth information for the media has been added, zero RTCP bandwidth has been negotiated, and AVPF is not offered.

Table A.12.1.1: SDP example

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 98 b=AS:30 b=RS:0 b=RR:0 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=ecn-capable-rtp: leap ect=0 aptime:20 a=maxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVP 99 b=AS:29 b=RS:0 b=RR:0 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=ecn-capable-rtp: leap ect=0 aptime:20 a=maxptime:240</pre> |

Comments:

The SDP offer includes the SDP attribute 'ecn-capable-rtp' to indicate that ECN is supported. The SDP offer further includes the parameters: 'leap' to indicate that the leap-of-faith initiation method is to be used; and 'ect=0' to request that the other endpoint sets the ECN bits to ECT(0). The SDP offer does not include the "rtcp-fb" attribute for negotiating use of the RTCP AVPF ECN feedback messages [84]. This results in RTP CMR [28] being used as the application specific feedback for ECN-triggered adaptation. The SDP offer also proposes to not use RTCP for the session.

The SDP answer is configured in the same way as in the offer to indicate that the ECN usage and its configuration is agreeable to be used in the session.

A.12.1.2 With RTP/AVPF and non-zero RTCP bandwidth

This SDP example is based on the SDP example found in Table A.3.0 except that bandwidth information for the media has been added. The negotiation of Reduced-Size RTCP is added together with the ECN negotiation. Non-zero RTCP bandwidth and AVPF have also been negotiated.

Table A.12.1.2: SDP example

| SDP offer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:30 b=RS:0 b=RR:2000 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=ecn-capable-rtp: leap ect=0 a=rtcp-rsize a=ptime:20 a=maxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVPF 99 a=acfg:1 t=1 b=AS:29 b=RS:0 b=RR:2000 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=ecn-capable-rtp: leap ect=0 a=rtcp-rsize a=ptime:20 a=maxptime:240</pre> |

Comments:

The SDP offer includes the SDP attribute 'ecn-capable-rtp' to indicate that ECN is supported. The SDP offer further includes the parameters: 'leap' to indicate that the leap-of-faith initiation method is to be used; and 'ect=0' to request that the other endpoint sets the ECN bits to ECT(0). The SDP offer does not include the "rtcp-fb" attribute for negotiating use of the RTCP AVPF ECN feedback messages [84]. This results in RTCP-APP CMR and reduced-size RTCP being used as the application specific feedback for ECN-related adaptation.

The SDP answer is configured in the same way as in the offer to indicate that the ECN usage and its configuration is agreeable to be used in the session.

A.12.1.3 With RTCP ECN feedback messages and RTCP XR ECN summary reports for inter-working with non-MTSI clients

The following SDP offer and SDP answer are possible when an MTSI client is inter-working with non-MTSI clients and when the MTSI client supports RTCP AVPF ECN feedback messages and RTCP XR ECN summary reports.

Table A.12.1.3: SDP example

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:30 b=RS:0 b=RR:2000 a=rtpmap:97 AMR/8000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=ecn-capable-rtp: leap ect=0 a=rtcp-fb:* nack ecn a=rtcp-xr:ecn-sum a=rtcp-rsize a=ptime:20 a=maxptime:240 </pre> |
| SDP answer |
| <pre> m=audio 49152 RTP/AVPF 99 a=acfg:1 t=1 b=AS:29 b=RS:0 b=RR:2000 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=ecn-capable-rtp: leap ect=0 a=rtcp-fb:* nack ecn a=rtcp-xr:ecn-sum a=rtcp-rsize a=ptime:20 a=maxptime:240 </pre> |

Comments:

The SDP offer is similar to the offer in Table A.12.2. The line "a=rtcp-fb:* nack ecn" is included to indicate that the RTCP AVPF ECN feedback messages can be used by all payload types for speech. The line "a=rtcp-xr:ecn-sum" is included to indicate that the RTCP XR ECN summary reports can also be used.

Since the offering MTSI client supports the RTCP AVPF ECN feedback messages and RTCP XR ECN summary reports there is no need to insert any media gateway in the path to solve inter-working.

A.12.2 SDP examples when using ECN for video in RTP

A.12.2.1 Without RTCP AVPF ECN feedback messages and RTCP XR ECN summary reports

The following SDP offer and SDP answer are likely when both MTSI clients in terminals use ECN for video and TMMBR for rate adaptation.

This SDP example is the same as the SDP example found in Tables A.4.4b and A.4.4c, except that the negotiation for ECN has been added.

Table A.12.2.1: Example SDP offer for H.264 video with ECN

| SDP offer |
|---|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0;profile-level-id=42e00a; \ sprop-parameter-sets=J0LgCpWgsToB/UA=,KM4Gag== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm tmmbr a=rtcp-fb:* ccm fir a=ecn-capable-rtp: leap ect=0 a=extmap:4 urn:3gpp:video-orientation </pre> |
| SDP answer |
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0;profile-level-id=42e00a; \ sprop-parameter-sets=J0LgCpWgsToB/UA=,KM4Gag== a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm tmmbr a=rtcp-fb:* ccm fir a=ecn-capable-rtp: leap ect=0 a=extmap:4 urn:3gpp:video-orientation </pre> |

Comments:

The SDP offer includes the SDP attribute 'ecn-capable-rtp' to indicate that ECN is supported. The SDP offer further includes the parameters: 'leap' to indicate that the leap-of-faith initiation method is to be used; and 'ect=0' to request that the other endpoint sets the ECN bits to ECT(0).

The SDP offer also includes an offer for AVPF to enable sending adaptation requests without following the normal rules for RTCP transmission intervals. TMMBR is also offered to indicate that this can be used for rate adaptation.

The SDP answer is configured in the same way as in the offer to indicate that the ECN usage and its configuration is agreeable to be used in the session.

A.12.2.2 With RTCP AVPF ECN feedback messages and RTCP XR ECN summary reports for inter-working with non-MTSI clients

The following SDP offer and SDP answer are possible when an MTSI client is inter-working with non-MTSI clients not supporting TMMBR and when the MTSI client supports RTCP AVPF ECN feedback messages and RTCP XR ECN summary reports.

This SDP example is the same as the SDP example found in Tables A.4.4b and A.4.4c, except that the negotiation for ECN has been added.

Table A.12.2.2: Example SDP offer for H.264 video with ECN

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0;profile-level-id=42e00a; \ sprop-parameter-sets=J0LgCpWgsToB/UA=,KM4Gag== a=ecn-capable-rtp: leap ect=0 a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm tmmbr a=rtcp-fb:* ccm fir a=rtcp-fb:* nack ecn a=rtcp-xr:ecn-sum a=extmap:4 urn:3gpp:video-orientation </pre> |
| SDP answer (non-MTSI client) |
| <pre> m=video 49154 RTP/AVPF 99 a=acfg:1 t=1 b=AS:315 b=RS:0 b=RR:2500 a=rtpmap:99 H264/90000 a=fmtp:99 packetization-mode=0;profile-level-id=42e00a; \ sprop-parameter-sets=J0LgCpWgsToB/UA=,KM4Gag== a=ecn-capable-rtp: leap ect=0 a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* nack ecn a=rtcp-xr:ecn-sum a=extmap:4 urn:3gpp:video-orientation </pre> |

Comments:

The SDP offer is similar to the offer in Table A.12.2.1. The line "a=rtcp-fb:* nack ecn" is included to indicate that the RTCP AVPF ECN feedback messages can be used all payload types for video. The line "a=rtcp-xr:ecn-sum" is included to indicate that the RTCP XR ECN summary reports can also be used.

The answering client does not support TMMBR and full intra requests and therefore removes these attribute lines when creating the SDP answer.

Since the offering MTSI client supports the RTCP AVPF ECN feedback messages and RTCP XR ECN summary reports there is no need to insert any media gateway in the path to provide inter-working.

A.13 SDP examples for MTSI client in terminal using fixed access

A.13.1 General

This clause includes SDP examples that may be applicable to an MTSI client in terminal using fixed access. The SDP examples in Annex A.13.2 to A.13.6 show SDPs including PCM, G.729, G.722 and G.729.1. Examples of SDP offers for the AMR and AMR-WB codecs are described in Annex A.1 and the corresponding examples of SDP answers are found in Annex A.3. SDP examples for EVRC, EVRC-B and EVRC-WB are found in [97].

Examples of SDP offer and answer for video are described in Annex A.4.

An example for an SDP offer for real-time text is described in Annex A.5.

These examples also include bandwidth information which is calculated assuming IPv6 and 20 ms packetization.

A.13.2 SDP examples for PCM

Table A.13.1 shows an example for the SDP offer and answer negotiation for PCM. The SDP offer uses the static payload type numbers that are defined in RFC 3551 [10] for PCM, i.e. payload type number 0 for μ -law PCM and payload type number 8 for A-law PCM, see also Clause 18.4.3. Since static payload type numbers are used, as shown on the m= line, then there is no need for adding any a=rtptime attribute lines. The answerer chooses to accept A-law PCM and therefore sends an SDP answer with RTP payload type number 8 on the m= line.

Table A.13.1: SDP example for PCM

| SDP offer |
|--|
| m=audio 49152 RTP/AVP 0 8 b=AS:88 a=ptime:20 a=maxptime:240 |
| SDP answer |
| m=audio 49152 RTP/AVP 8 b=AS:88 a=ptime:20 a=maxptime:240 |

Comments:

The SDPs further describe that the clients prefer to receive speech with 20 ms packetization (ptime is set to 20) but up to 240 ms packetization is allowed.

Table A.13.2 shows an example for how the PCM codec can be negotiated using dynamic payload type numbers. In this case, payload type number 96 is used for μ -law PCM and payload type number 97 is used for A-law PCM. The answerer chooses to accept μ -law PCM.

Table A.13.2: SDP example for PCM

| SDP offer |
|--|
| m=audio 49152 RTP/AVP 96 97 b=AS:88 a=rtptime:96 PCMU/8000/1 a=rtptime:97 PCMA/8000/1 a=ptime:20 a=maxptime:240 |

| SDP answer |
|---|
| <pre>m=audio 49152 RTP/AVP 96 b=AS:88 a=rtpmap:96 PCMU/8000/1 a=ptime:20 a=maxptime:240</pre> |

Comments:

This example is included here to show that it is possible to use dynamic payload type numbers also for codecs for which static payload type numbers have been defined. It is however preferable to use static payload type numbers, see Clause 18.4.3.

A.13.3 SDP example for G.722

Table A.13.3 shows an example for how G.722 can be negotiated using the static payload type number (9) defined in RFC 3551 [10], see also Clause 18.4.3.

Table A.13.3: SDP example for G.722

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 9 b=AS:88 a=rtpmap:9 G722/8000/1 a=ptime:20 a=maxptime:240</pre> |
| SDP answer |
| <pre>m=audio 49152 RTP/AVP 9 b=AS:88 a=rtpmap:9 G722/8000/1 a=ptime:20 a=maxptime:240</pre> |

Comments:

The G.722 codec uses an RTP clock rate of 8 kHz even though G.722 is a wideband speech codec that uses a sampling frequency of 16 kHz. This means that the RTP Time Stamp is sampled with 8 kHz.

The SDPs further describe that the clients prefer to receive speech with 20 ms packetization (ptime is set to 20) but up to 240 ms packetization is allowed.

A.13.4 SDP example for EVS, AMR-WB, G.722, AMR, PCM and DTMF

Table A.13.4 shows an example where an MTSI client in terminal using fixed access has been developed to support fixed-mobile interworking without the need for transcoding in a media gateway. It therefore supports the G.722 and PCM codecs that are normally used in fixed networks. In addition, it also supports the AMR-WB and AMR codecs in the same way as an MTSI client in terminal using mobile access would do. The SDP offer includes all these codecs as well as DTMF.

Table A.13.4: SDP example for EVS, AMR-WB, G.722, AMR, PCM and DTMF

| SDP offer |
|--|
| <pre> m=audio 49152 RTP/AVP 96 97 98 9 99 100 8 0 105 106 b=AS:89 b=RS:0 b=RR:4000 a=rtpmap:96 EVS/16000/1 a=fmtp:96 br=64; bw=swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:9 G722/8000/1 a=rtpmap:0 PCMU/8000/1 a=rtpmap:8 PCMA/8000/1 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 a=rtpmap:106 telephone-event/8000 a=fmtp:106 0-15 a=ptime:20 a=maxptime:240 </pre> |

Comments:

The wideband codecs (AMR-WB and G.722) are listed as preferred over the narrowband codecs (AMR and PCM). This ensures that a wideband service will be set up whenever possible.

The AMR and AMR-WB codecs are listed here as preferred over the PCM and G.722 codecs, respectively, because of their lower bitrate and also because of their bitrate adaptation capabilities.

The SDP offer includes DTMF with both 8 kHz and 16 kHz RTP clock rate since there are codecs with both clock rates in the offer. An answerer is expected to accept the DTMF variant that has the same clock rate as for the accepted codec. This means that when G.722 is accepted then DTMF with 8 kHz clock rate should also be accepted, even though G.722 is a wideband speech codec. This is because G.722 uses 8 kHz clock rate, see RFC3551 [10].

Since the clock rate of EVS is set to 16 kHz, regardless of the bandwidth in the session, DTMF with 16 kHz RTP clock rate should be accepted when EVS is accepted.

A.14 SDP offers and answers for speech sessions with EVS

These examples show SDP offers and answers for speech sessions where EVS is negotiated. These SDP offer and answer examples are designed to highlight the respective area that is being described and should therefore not be considered as complete SDP offers and answers.

A.14.1 SDP offers initiated by MTSI client in terminal

The SDP offers below can be used by MTSI client in terminal, depending on the access technology or the number of audio channels.

A.14.1.1 Unknown access technology

When the access technology is unknown to MTSI client in terminal, the SDP offer below can be used to initiate a speech session. In this example, RTP Payload Type 97 is defined for EVS, and two sets of RTP Payload Types, 98 and 99, and 100 and 101 are defined for AMR-WB and AMR respectively.

Table A.14.1: SDP example

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:145 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/1 a=fmtp:97 max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 a=maxptime:240 </pre> |

Comments:

Since the MTSI client in terminal is not aware of the access technology it uses, all bit-rates of EVS are offered in the session.

The MTSI client in terminal supports all bandwidths, up to fullband.

Regardless of the bandwidth used in the session, clock rate of EVS shall be set to 16 kHz.

Media level b=AS is computed for the highest bit-rate of EVS, 128 kbps, with IPv4 and Header-full payload format, which is greater than the b=AS values of other RTP Payload Types.

A.14.1.2 EGPRS

When the access technology is EGPRS, the SDP offer below can be used to initiate a speech session. In this example, RTP Payload Type 97 is defined for EVS, and two sets of RTP Payload Types, 98 and 99, and 100 and 101 are defined for AMR-WB and AMR respectively.

Table A.14.2: SDP example

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:33 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-wb; max-red=200 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=200 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=200; octet-align=1 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=200 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=200; octet-align=1 aptime:40 a=maxptime:240 </pre> |

Comments:

It is assumed that the modulation and coding scheme (MCS) of EGPRS used in this session is MCS-7 [132] or higher, which supports at least 44.8 kbps. The bit-rate available for data will be reduced further from the overhead for RLC and MAC headers.

All bit-rates of EVS from 5.9 (SC-VBR) to 24.4 kbps are offered in the session.

The MTSI client in terminal supports narrowband and wideband.

Media level b=AS is computed for 24.4 kbps of EVS with Header-full payload format, or for 23.85 which results in a b=AS value of 33 kbps. MCS lower than MCS-7 would necessitate the use of mode-set parameter for AMR-WB as MCS-6 supports only 29.6 kbps. However, higher MCS values would leave lower overhead for channel coding.

A.14.1.3 NR/E-UTRAN/HSPA

When the access technology is E-UTRAN or HSPA, the SDP offer below can be used to initiate a speech session. In this example, RTP Payload Type 97 is defined for EVS, and two sets of RTP Payload Types, 98 and 99, and 100 and 101 are defined for AMR-WB and AMR respectively.

Table A.14.3: SDP offer

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:42 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 amaxptime:240 </pre> |

Comments:

It is assumed that 42 kbps is reserved for speech by the radio access technology.

All bit-rates of EVS from 5.9 (SC-VBR) to 24.4 kbps are offered in the session.

The MTSI client in terminal supports all bandwidths, up to super-wideband.

Media level b=AS is computed for 24.4 kbps of EVS with Header-full payload format.

A.14.1.4 Dual-mono

When dual-mono is offered, the SDP offer below can be used to initiate a speech session. In this example in Table A.14.4a, RTP Payload Types 97 and 98 are defined for EVS, and two sets of RTP Payload Types, 99 and 100, and 101 and 102 are defined for AMR-WB and AMR respectively.

Table A.14.4a: SDP offer

| SDP offer |
|--|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 102 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:50 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/2 a=fmtp:97 br=16.4; bw=swb; max-red=220 a=rtpmap:98 EVS/16000/1 a=fmtp:98 br=5.9-24.4; bw=nb-swb; ch-aw-recv=-1; max-red=220 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR-WB/16000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:101 AMR/8000/1 </pre> |

```

a=fmtp:101 mode-change-capability=2; max-red=220
a=rtpmap:102 AMR/8000/1
a=fmtp:102 mode-change-capability=2; max-red=220; octet-align=1
a=ptime:20
a=maxptime:240

```

Comments:

It is assumed that 50 kbps is reserved for speech by the radio access technology.

Dual-mono session consisting of two 16.4 kbps SWB channels is offered for the send and the receive directions.

In addition, all bit-rates of EVS from 5.9 (SC-VBR) to 24.4 kbps are offered in the session. Channel-aware mode is disabled in the session for the receiving direction.

Media level b=AS is computed for a dual-mono session including 16.4 kbps of EVS with IPv4 and Header-full payload format which results in a b=AS value of 50 kbps.

In the example in Table A.14.4b, RTP Payload Types 97 and 98 are defined for EVS, and two sets of RTP Payload Types, 99 and 100, and 101 and 102 are defined for AMR-WB and AMR respectively.

Table A.14.4b: SDP example

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 102 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:66 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/2 a=fmtp:97 br=13.2-24.4; bw=nb-swb; ch-aw-recv=3; max-red=220 a=rtpmap:98 EVS/16000/1 a=fmtp:98 br=13.2-24.4; bw=nb-swb; ch-aw-recv=3; max-red=220 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR-WB/16000/1 a=fmtp:100 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=220 a=rtpmap:102 AMR/8000/1 a=fmtp:102 mode-change-capability=2; max-red=220; octet-align=1 a=ptime:20 a=maxptime:240 </pre> |

Comments:

It is assumed that 66 kbps is reserved for speech by the radio access technology.

Dual-mono session consisting of two NB-SWB channels is offered for the send and the receive directions. All bit-rates of EVS from 13.2 to 24.4 kbps are offered in the session. Partial redundancy (channel-aware mode) is used at the start of the session for the receive direction.

Media level b=AS is computed for a dual-mono session including 24.4 kbps of EVS with IPv4, Header-full payload format which results in a b=AS value of 66 kbps.

A.14.2 SDP offers initiated by media gateway

The SDP offer below can be used by media gateway.

A.14.2.1 Network between MTSI using fixed access and MTSI modifying SDP offer to configure EVS AMR-WB IO mode

This example shows the SDP offer when the session is initiated from MTSI client in terminal using fixed access, which supports AMR (all modes) and AMR-WB (all modes). In addition, EVS, G.722, and PCM codecs are supported by the MTSI client in terminal.

The offers for AMR, AMR-WB and EVS codecs are changed by the network to include AMR {12.2, 7.4, 5.9, 4.75}, AMR-WB {12.65, 8.85, 6.60}, EVS AMR-WB IO {12.65, 8.85, 6.60}. For payload type 98, the network also changes the offer to start the session with EVS in EVS AMR-WB IO mode if it is agreed to use the codec.

For EVS, RTP Payload Types 97 and 98 are defined, for example, to initiate a speech session with another MTSI client in terminal using fixed access supporting a bit-rate of 64 kbps or radio access supporting all bit-rates from 5.9 to 24.4 kbps respectively.

The MTSI client in terminal does not include the evs-mode-switch parameter in the initial SDP offer, see Table 6.2a. The SDP is instead changed by the network.

Table A.14.5: SDP example

| SDP offer from MTSI client in terminal using fixed access |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 99 9 100 0 8 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:81 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=64; bw=wb; mode-change-capability=2; max-red=220 a=rtpmap:98 EVS/16000/1 a=fmtp:98 br=5.9-24.4; bw=nb-wb; mode-change-capability=2; max-red=220 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220 a=rtpmap:9 G722/8000/1 a=rtpmap:0 PCMU/8000/1 a=rtpmap:8 PCMA/8000/1 aptime:20 amaxptime:240 </pre> |
| SDP offer modified by network |
| <pre> m=audio 49152 RTP/AVP 97 98 99 9 100 0 8 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:81 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=64; bw=wb; mode-set=0,1,2; mode-change-capability=2; max-red=220 a=rtpmap:98 EVS/16000/1 a=fmtp:98 br=5.9-24.4; bw=nb-wb; evs-mode-switch=1; mode-set=0,1,2; mode-change-capability=2; \ max-red=220 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-set=0,1,2; max-red=0 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-set=0,2,4,7; max-red=0 a=rtpmap:9 G722/8000/1 a=rtpmap:0 PCMU/8000/1 a=rtpmap:8 PCMA/8000/1 aptime:20 amaxptime:80 </pre> |

Comments:

For EVS, narrowband and wideband are supported for Payload Type 98 while only wideband is supported for Payload Type 97.

If Payload Type 97 is negotiated, EVS Primary mode will be used at the start or update of the session, at 64 kbps wideband. EVS AMR-WB IO mode can be used in the middle of session, which can be switched by the sender, or the receiver with CMR.

If Payload Type 98 is negotiated, EVS AMR-WB IO mode will be used at the start or update of the session, at 6.60, 8.85, or 12.65 kbps.

A.14.3 SDP answers from MTSI client in terminal

The SDP answers below can be used by MTSI client in terminal, depending on access technology or service policy. It is assumed that SDP offers such as Tables A.14.1, A.14.2, A.14.3, A.14.4, or A.14.5 are received.

A.14.3.1 SDP answer from MTSI client in terminal when narrowband speech is negotiated

In this example, the MTSI client in terminal includes only narrowband speech in the SDP answer.

Table A.14.6: SDP example

| SDP answer |
|--|
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:30 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/1 a=fmtp:97 br=5.9-13.2; bw=nb; mode-set=0,1,2; max-red=220 aptime:20 a=maxptime:240</pre> |

Comments:

The SDP answer contains all bit-rates from 5.9 to 13.2 kbps, with IPv4 for the send and the receive directions.

Media level b=AS is computed for 13.2 kbps of EVS Primary mode, or 12.65 kbps of EVS AMR-WB IO mode, with Header-full payload format, either of which results in 30 kbps.

A.14.3.2 SDP answer from MTSI client in terminal when up to wideband speech is negotiated

In this example, the MTSI client in terminal includes narrowband and wideband speech in the SDP answer.

Table A.14.7: SDP example

| SDP answer |
|---|
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:49 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/1 a=fmtp:97 br=7.2-32; bw=nb-wb; max-red=220 aptime:20 a=maxptime:240</pre> |

Comments:

The SDP answer contains all bit-rates from 7.2 to 32 kbps, with IPv4 for the send and the receive directions.

As neither br-send nor br-recv of the SDP answer includes 5.9 kbps, source controlled variable bit-rate (SC-VBR) coding is not used for the session.

A.14.3.3 SDP answer from MTSI client in terminal when only wideband speech is negotiated

In this example, the MTSI client in terminal includes only wideband speech in the SDP answer.

Table A.14.8: SDP example

| SDP answer |
|--|
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:49 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/1 a=fmtp:97 br=9.6-32; bw=wb; mode-set=0,1,2; hf-only=1; max-red=220 a=ptime:20 a=maxptime:240</pre> |

Comments:

The SDP answer contains all bit-rates from 9.6 to 32 kbps, with IPv4 for the send and the receive directions.

In EVS AMR-WB IO mode, only 6.60, 8.85, and 12.65 kbps are used.

Only Header-full format is used in the session.

A.14.3.4 SDP answer from MTSI client in terminal when up to super-wideband speech is negotiated

In this example, the MTSI client in terminal includes narrowband, wideband, and super-wideband speech in the SDP answer.

Table A.14.9: SDP example

| SDP answer |
|---|
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:65 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/1 a=fmtp:97 br-send=8-48; br-recv=32-48; bw-send=nb-swb; bw-recv=swb; max-red=220 a=ptime:20 a=maxptime:240</pre> |

Comments:

The SDP answer contains bit-rates from 8 to 48 kbps for the send direction, and bit-rates from 32 to 48 kbps for the receive direction, with IPv4.

The SDP answer contains bandwidths from narrowband to super-wideband for the sending direction, and only super-wideband for the receiving direction.

A.14.3.5 SDP answer from MTSI client in terminal when only super-wideband speech is negotiated

In this example, the MTSI client in terminal includes only super-wideband speech in the SDP answer.

Table A.14.10: SDP example

| SDP answer |
|---|
| <pre>m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:65</pre> |

```

b=RS:0
b=RR:2000
a=rtptime:97 EVS/16000/1
a=fmtp:97 br=16.4-48; bw=swb; cmr=-1; max-red=220
a=ptime:20
a=maxptime:240

```

Comments:

The SDP answer contains bit-rates from 16.4 to 48 kbps, with IPv4 for the send and the receive directions.

CMR is not used in this session.

A.14.3.6 SDP answer from MTSI client in terminal using WLAN

This example shows the SDP answer when the MTSI client in terminal is using WLAN as the access technology.

Table A.14.11: SDP example

| SDP answer |
|---|
| <pre> m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:37 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/1 a=fmtp:97 br=13.2-32; ch-aw-recv=3; max-red=160 a=ptime:80 a=maxptime:240 </pre> |

Comments:

The SDP answer contains all bit-rates from 13.2 to 32 kbps, with IPv4 for the send and the receive directions. Channel-aware mode with offset 3 is enabled for the receiving direction.

A.14.3.7 SDP answer from MTSI client in terminal supporting dual-mono

This example shows the SDP answer when the MTSI client in terminal supports dual-mono.

Table A.14.12: SDP example

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:50 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/2 a=fmtp:97 br=16.4; bw=swb; ch-send=2; ch-recv=2; max-red=220 a=ptime:20 a=maxptime:240 </pre> |

Comments:

The SDP answer contains a dual-mono session consisting of two 16.4 kbps SWB channels, for the send and the receive directions, with IPv4.

A.14.3.8 SDP answer from MTSI client in terminal supporting dual-mono for send direction

This example shows the SDP answer when the MTSI client in terminal supports dual-mono only for the send direction.

Table A.14.13: SDP example

| SDP answer |
|---|
| <pre> m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:42 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/2 a=fmtp:97 br-send=16.4; br-recv=24.4; bw-send=swb; bw-recv=nb-swb; ch-send=2; ch-recv=1; \ max-red=220 a=ptime:20 a=maxptime:240 </pre> |

Comments:

The SDP answer contains a dual-mono session consisting of two 16.4 kbps SWB channels for the send direction, and a mono session consisting of 24.4 kbps for the receive direction, with IPv4.

In the usage of the channels parameter, "/n," n represents the number of audio channels, see [125].

'ch-send=2' indicates that the answerer will include two audio channels (dual-mono) in the RTP packets in the sending direction, see Clause 7.5.2.1.9. 'ch-recv=1' indicates that the answerer expects that only one audio channel will be included in the RTP packets in the receiving direction.

A.14.3.9 SDP answer from MTSI client in terminal when SC-VBR is negotiated

In this example, the MTSI client in terminal includes narrowband and wideband speech in the SDP answer. This SDP answer is used to negotiate the lowest possible bit-rate of EVS Primary, for example, due to limited network capacity.

Table A.14.13a: SDP example

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:25 b=RS:0 b=RR:2000 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9; bw=nb-wb; mode-set=0; max-red=220 a=ptime:20 a=maxptime:240 </pre> |

Comments:

Media level b=AS of SC-VBR is computed for the b=AS of its highest component bit-rate, 8 kbps. Therefore br=5.9, 5.9-7.2, and 5.9-8 result in the same b=AS value of 25, with IPv4.

If frame aggregation is not used, no codec mode of EVS AMR-WB IO other than 6.6 kbps can be used at this b=AS value, whether mode-set is included or not,

A.14.4 SDP answers from MTSI client in terminal using fixed access

These examples show the SDP answers when the MTSI client in terminal is using fixed access.

A.14.4.1 SDP answer from MTSI client in terminal using fixed access when narrowband speech is negotiated

In this example, the MTSI client in terminal includes only narrowband speech in the SDP answer.

Table A.14.14: SDP example

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:30 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/1 a=fmtp:97 br=13.2; bw=nb; dtx=0; mode-set=0,1,2; max-red=220 aptime:20 amaxptime:240 </pre> |

Comments:

The SDP answer contains only 13.2 kbps.

DTX is disabled in the session.

Media level b=AS with IPv4 is computed for 13.2 kbps of EVS Primary mode, or 12.65 kbps of EVS AMR-WB IO mode, with Header-full payload format, either of which results in 30 kbps.

A.14.4.2 SDP answer from MTSI client in terminal using fixed access when only wideband speech is negotiated

In this example, the MTSI client in terminal includes only wideband speech in the SDP answer.

Table A.14.15: SDP example

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:81 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/1 a=fmtp:97 br=64; bw=wb; dtx=0; max-red=220 aptime:20 amaxptime:240 </pre> |

Comments:

The SDP answer contains only 64 kbps.

DTX is disabled in the session.

A.14.4.3 SDP answer from MTSI client in terminal using fixed access when only super-wideband speech is negotiated

In this example, the MTSI client in terminal includes only super-wideband speech in the SDP answer.

Table A.14.16: SDP example

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVPF 97 a=acfg:1 t=1 b=AS:113 b=RS:0 b=RR:2000 a=rtptime:97 EVS/16000/1 a=fmtp:97 br=96; bw=swb; dtx=0; max-red=220 aptime:20 amaxptime:240 </pre> |

Comments:

The SDP answer contains only 96 kbps.

DTX is disabled in the session.

A.15 SDP offers and answers with ANBR capability signaling

Table A.15.1 demonstrates an example SDP offer with Access Network Bitrate Recommendation (ANBR) capability signalling defined in clause 6.2.9 for speech. The offer for ANBR capability signaling is indicated in the last line via the SDP attribute 'anbr'.

Table A.15.1: Example SDP offer with ANBR capability signalling for speech

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 98 99 100 101 b=AS:42 b=RS:0 b=RR:2000 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=220; octet-align=1 aptime:20 a=maxptime:240 a=anbr</pre> |

An example SDP answer is shown in Table A.15.2, where the ANBR capability signalling is also supported by the answerer, as indicated by the last line.

Table A.15.2: Example SDP answer with ANBR capability signalling for speech

| SDP answer |
|---|
| <pre>m=audio 49152 RTP/AVPF 97 b=AS:30 b=RS:0 b=RR:2000 a=acfg:1 t=1 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-13.2; bw=nb-wb; mode-set=0,1,2; max-red=220 aptime:20 a=maxptime:240 a=anbr</pre> |

Table A.15.3 demonstrates another example SDP offer with ANBR capability signalling, this time for video.

Table A.15.3: Example SDP offer with ANBR capability signalling for video

| SDP offer |
|---|
| <pre>m=video 49154 RTP/AVP 98 97 100 99 b=AS:690 b=RS:0 b=RR:5000 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:100 H264/90000 a=fmtp:100 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KAHpWgNQ9oB/U=,aM46gA==</pre> |

```

a=imageattr:100 send [x=848,y=480] recv [x=848,y=480]
a=rtpmap:99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e01f; \
    sprop-parameter-sets=Z0KADZWgUH6AflA=,aM46gA==
a=imageattr:99 send [x=320,y=240] recv [x=320,y=240]
a=rtpmap:98 H265/90000
a=fmtp:98 profile-id=1; level-id=93; \
    sprop-vps=QAEMAf//AWAAAAMAgAAAAwAAAwBaLAUg; \
    sprop-sps=QgEBAWAAAAMAgAAAAwAAAwBaoAaiAeFlLktIvQB3CAQQ; \
    sprop-pps=RAHAcYDZIA==
a=imageattr:98 send [x=848,y=480] recv [x=848,y=480]
a=rtpmap:97 H265/90000
a=fmtp:97 profile-id=1; level-id=93; \
    sprop-vps=QAEMAf//AWAAAAMAgAAAAwAAAwA8LAUg; \
    sprop-sps=QgEBAWAAAAMAgAAAAwAAAwA8oAoIDxZS5LSL0AdwgEE=; \
    sprop-pps=RAHAcYDZIA==
a=imageattr:97 send [x=320,y=240] recv [x=320,y=240]
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=extmap:4 urn:3gpp:video-orientation
a=anbr

```

The corresponding example SDP answer is shown in Table A.15.4, where the ANBR capability signalling is also supported by the answerer, as indicated by the last line.

Table A.15.4: Example SDP answer with ANBR capability signalling for video

| SDP answer |
|---|
| <pre> m=video 49156 RTP/AVPF 98 b=AS:690 b=RS:0 b=RR:5000 a=acfg:1 t=1 a=rtpmap:98 H265/90000 a=fmtp:98 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAAwAAAwBaLAUg; \ sprop-sps=QgEBAWAAAAMAgAAAAwAAAwBaoAaiAeFlLktIvQB3CAQQ; \ sprop-pps=RAHAcYDZIA== a=imageattr:98 send [x=848,y=480] recv [x=848,y=480] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation a=anbr </pre> |

A.16 SDP offers and answers with QoS hint signalling

Table A.16.1 demonstrates an example SDP offer with QoS hint signalling defined in clause 6.2.7.4. The offer for QoS hint signalling is indicated in the last line via the SDP attribute '3gpp-qos-hint'.

Table A.16.1: Example SDP offer with QoS hint signalling

| SDP offer |
|--|
| <pre> m=video 43200 RTP/AVP 100 b=AS:15000 b=RS:0 b=RR:2500 a=pcfg:1 t=1 a=rtpmap:100 H265/90000 a=fmtp:100 profile-id=1;level-id=153 a=imageattr:100 send [x=3840,y=2160] a=label:flus a=sendonly a=3gpp-qos-hint:loss=0.00001;latency=300 </pre> |

An example SDP answer is shown in Table A.16.2, where the QoS hint signalling is also supported by the answerer, as indicated by the last line.

Table A.16.2: Example SDP answer with QoS hint signalling

| SDP answer |
|--|
| <pre>m=video 43200 RTP/AVPF 100 b=AS:15000 b=RS:0 b=RR:2500 a=acfg:1 t=1 a=rtptime:100 H265/90000 a=fmtp:100 profile-id=1;level-id=153 a=imageattr:100 recv [x=3840,y=2160] a=label:flus a=recvonly a=3gpp-qos-hint:loss=0.00001;latency=300</pre> |

Table A.16.3 demonstrates an example SDP offer with QoS hint signalling defined in clause 6.2.7.4, using explicit split of the end-to-end values. The offer suggests itself to use less than half of the end-to-end loss, but more than half of the end-to-end latency in the SDP attribute '3gpp-qos-hint'.

Table A.16.3: Example SDP offer with QoS hint signalling and explicit split

| SDP offer |
|--|
| <pre>m=video 43200 RTP/AVP 100 b=AS:15000 b=RS:0 b=RR:2500 a=pcfg:1 t=1 a=rtptime:100 H265/90000 a=fmtp:100 profile-id=1;level-id=153 a=imageattr:100 send [x=3840,y=2160] a=label:flus a=sendonly a=3gpp-qos-hint:loss=0.00002/local:0.000005;latency=600/local:400</pre> |

Table A.16.4 demonstrates another example SDP answer where the QoS hint signalling is also supported by the answerer, but where neither the desired QoS hint end-to-end values nor the QoS hint split values from the SDP offer can be supported and are changed in the SDP answer.

Table A.16.4: Example SDP answer with QoS hint signalling changing QoS hint values and split

| SDP answer |
|--|
| <pre>m=video 43200 RTP/AVPF 100 b=AS:15000 b=RS:0 b=RR:2500 a=acfg:1 t=1 a=rtptime:100 H265/90000 a=fmtp:100 profile-id=1;level-id=153 a=imageattr:100 recv [x=3840,y=2160] a=label:flus a=recvonly a=3gpp-qos-hint:loss=0.1;latency=500/local:100</pre> |

The resulting loss hint is 0.05% for both SDP offerer and SDP answerer (half of the provided end-to-end value), which is a significant relaxation in loss rate compared to the offer in Table A.16.3 that might not be sufficient to provide an acceptable user experience. The resulting latency hint is 400 ms (500-100) for the SDP offerer and 100 ms for the SDP answerer, which is stricter end-to-end than the offer in Table A.16.3, but the answerer takes on the stricter latency and leaves the offerer part of the split from the offer (400 ms) unmodified.

A.17 SDP offers and answers with data channel capability signalling

The ellipsis ("...") in the examples in this clause is not part of the SDP but indicates possible presence of other media descriptions in addition to the ones shown in the examples.

Table A.17.1 demonstrates an example SDP offer with data channel capability signalling for the "bootstrap" data channel defined in clause 6.2.10. The offering part is an ICE Lite agent, indicated by "a=ice-lite" on SDP session level (i.e., before first m= line), and thus only offers host candidates, in this example a single host candidate aligned with address information on the corresponding m= and c= lines.

Table A.17.1: Example SDP offer with data channel capability signalling

| SDP offer |
|---|
| <pre>a=ice-options:ice2 a=ice-lite ... m=application 52718 UDP/DTLS/SCTP webrtc-datachannel c=IN IP4 192.0.2.156 b=AS:500 a=candidate:1 1 UDP 2130706431 192.0.2.156 52718 typ host a=ice-ufraq:8hhY a=ice-pwd:asd88fgpdd777uzjYhagZg a=max-message-size:1024 a=sctp-port:5000 a=setup:actpass a=fingerprint:SHA-1 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB a=tls-id: abc3de65cddef001be82 a=dcmmap:0 subprotocol="http"</pre> |

An example SDP answer is shown in Table A.17.2, where the data channel capability signalling from Table A.17.1 is also supported and accepted by the answerer, as indicated by the non-zero port on the m= line. The answering part is an ICE Lite agent, indicated by "a=ice-lite" on SDP session level, and only supports ICE according to the predecessor ICE specification to [184] as indicated by no "a=ice-options:ice2" being included on SDP session level.

Table A.17.2: Example SDP answer with data channel capability

| SDP answer |
|--|
| <pre>a=ice-lite ... m=application 52718 UDP/DTLS/SCTP webrtc-datachannel c=IN IP4 192.0.2.1 b=AS:500 a=candidate:1 1 UDP 2130706431 192.0.2.1 52718 typ host a=ice-ufraq:9uB6 a=ice-pwd:YH75Fviy6338Vbrhr1p8Yh a=max-message-size:1024 a=sctp-port:5002 a=setup:passive a=fingerprint:SHA-1 5B:AD:67:B1:3E:82:AC:3B:90:02:B1:DF:12:5D:CA:6B:3F:E5:54:FA a=tls-id: dcb3ae65cddef0532d42 a=dcmmap:0 subprotocol="http"</pre> |

Table A.17.3 demonstrates an example SDP offer with multiple possible data channel application sources for the "bootstrap" data channel defined in Table 6.2.10.1-2. In this example, the offering part supports full ICE, indicated by no "a=ice-lite" on SDP session level.

Table A.17.3: Example SDP offer with multiple data channel application sources

| SDP offer |
|---|
| <pre>a=ice-options:ice2 ... m=application 52718 UDP/DTLS/SCTP webrtc-datachannel c=IN IP6 fe80::6676:baff:fe9c:ee4a b=AS:500 a=candidate:1 1 UDP 2130706431 fe80::6676:baff:fe9c:ee4a 52718 typ host a=ice-ufraq:8hhY</pre> |


```

a=ice-pwd:asd88fgpdd777uzjYhagZg
a=max-message-size:1024
a=sctp-port:5000
a=setup:actpass
a=fingerprint:SHA-1 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id: abc3de65cddef001be82
a=dcmap:0 subprotocol="http"
a=dcmap:10 subprotocol="http"
a=dcmap:100 subprotocol="http"
a=dcmap:110 subprotocol="http"

```

An example SDP answer is shown in Table A.17.4, where only one of the the data channel application sources from the offer in Table A.17.3 is accepted by the answerer, removing the other a=dcmap lines.

Figure 6.2.10.1-3 in clause 6.2.10.1 may be used as illustration to this example, in which case UE A in that Figure would send the offer in Table A.17.3, and UE B would send the answer in Table A.17.4.

In this SDP answer, the answerer (UE B) only accepts stream ID 110 to receive the data channel application from the offerer (UE A), but UE B has rejected to use any other data channel application provider.

Table A.17.4: Example UE SDP answer choosing a single data channel application source

| SDP answer |
|--|
| <pre> a=ice-options:ice2 a=ice-lite ... m=application 52718 UDP/DTLS/SCTP webrtc-datachannel c=IN IP4 192.0.2.1 b=AS:500 a=candidate:1 1 UDP 2130706431 192.0.2.1 52718 typ host a=ice-ufrag:9uB6 a=ice-pwd:YH75Fviy6338Vbrhr1p8Yh a=max-message-size:1024 a=sctp-port:5002 a=setup:passive a=fingerprint:SHA-1 5B:AD:67:B1:3E:82:AC:3B:90:02:B1:DF:12:5D:CA:6B:3F:E5:54:FA a=tls-id: dcb3ae65cddef0532d42 a=dcmap:110 subprotocol="http" </pre> |

Figure 6.2.10.1-3 in clause 6.2.10.1 may be used as illustration also to the example in Table A.17.5, in which case UE A in Figure 6.2.10.1-3 would send the offer in Table A.17.3, and the SDP answer sent back to UE A from the network would be the one in Table A.17.5.

In the SDP answer in Table A.17.5 sent from UE A's (local) network, it is accepting stream ID 10 that would be used by UE A to receive its own, chosen data channel application, corresponding to the data channel application sent to UE B in stream ID 110 based on the SDP answer in Table A.17.4 such that both UEs can use the same application. That application is however received through different stream IDs for UE A and UE B, as shown in Figure 6.2.10.1-3.

Table A.17.5: Example network SDP answer choosing a single data channel application source

| SDP answer |
|--|
| <pre> a=ice-options:ice2 a=ice-lite ... m=application 52718 UDP/DTLS/SCTP webrtc-datachannel c=IN IP4 192.0.2.1 b=AS:500 a=candidate:1 1 UDP 2130706431 192.0.2.1 52718 typ host a=ice-ufrag:9uB6 a=ice-pwd:YH75Fviy6338Vbrhr1p8Yh a=max-message-size:1024 a=sctp-port:5010 a=setup:active a=fingerprint:SHA-1 BC:8A:99:A0:E3:28:CA:B3:09:20:1B:FD:21:D5:AC:B6:F3:5E:45:AF a=tls-id: cd3bea56dced0f35d224 a=dcmap:10 subprotocol="http" </pre> |

Table A.17.6 demonstrates an example SDP (re-)offer that adds two non-bootstrap data channel streams used by the data channel application in the bootstrap data channel in Table A.17.5. The data channel application streams (two in this example) desire specific loss and latency characteristics indicated by the "a=3gpp-qos-hint" line (see also Annex A.16) and are offered as a separate m= line due to having different QoS requirements and different destination (e.g. a peer UE) than the bootstrap data channel. The stream with ID 38754 has a strict latency requirement and data older than 150 ms will not be transmitted or re-transmitted. The stream with ID 7216 requires lower loss but can accept somewhat higher latency than stream ID 38754 and therefore allows at most 5 SCTP-level retransmissions.

Table A.17.6: Example SDP offer with data channel application streams

| SDP offer |
|---|
| <pre> c=IN IP4 192.0.2.156 a=ice-options:ice2 a=ice-lite ... m=application 52718 UDP/DTLS/SCTP webrtc-datachannel b=AS:500 a=candidate:1 1 UDP 2130706431 192.0.2.156 52718 typ host a=ice-ufrag:8hhY a=ice-pwd:asd88fgpdd777uzjYhagZg a=max-message-size:1024 a=sctp-port:5000 a=setup: actpass a=fingerprint:SHA-1 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB a=tls-id: abc3de65cddef001be82 a=dcmap:10 subprotocol="http" m=application 52720 UDP/DTLS/SCTP webrtc-datachannel b=AS:1000 a=candidate:1 1 UDP 2130706431 192.0.2.156 52720 typ host a=ice-ufrag:9uB6 a=ice-pwd: YH75Fviy6338Vbrhrlp8Yh a=max-message-size:1024 a=sctp-port:5000 a=setup: actpass a=fingerprint:SHA-1 BC:8A:99:A0:E3:28:CA:B3:09:20:1B:FD:21:D5:AC:B6:F3:5E:45:AF a=tls-id: cd3bea56dced0f35d224 a=dcmap:38754 max-time=150;label="low latency" a=dcmap:7216 max-retr=5;label="low loss" a=3gpp-qos-hint:loss=0.01;latency=100 </pre> |

A.18 SDP offers and answers for ITT4RT

Table A.18.1 shows an example of an SDP offer for an ITT4RT session with a 360 video, 2 overlay streams, and a scene description.

Table A.18.1: Example SDP offer with scene description signalling

| SDP offer |
|---|
| <pre> V=0 o=ITT4RT 3413526809 0 IN IP4 server.example.com s=Example of using AS in MTSI c=IN IP4 aaa.bbb.ccc.ddd b=AS:345 t=0 0 a=tcap:1 RTP/AVPF a=itt4rt_group: 1 2 3 m=video 49154 RTP/AVP 99 a=pcfg:1 t=1 b=AS:315 b=RS:0 b=RR:5000 a=rtmpmap:99 H264/90000 a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=3gpp_360video: Stereo a=mid:1 m=video 49154 RTP/AVP 99 </pre> |

```
a=pcfg:1 t=1
b=AS:315
b=RS:0
b=RR:5000
a=rtpmap:99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA==
a=mid:2
a=3gpp_overlay:2 1 0,0,0,0,0,0,0,0,0,0

m=video 49154 RTP/AVP 99
a=pcfg:1 t=1
b=AS:315
b=RS:0
b=RR:5000
a=rtpmap:99 H264/90000
a=fmtp:99 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA==
a=mid:3
a=3gpp_overlay:3 1 0,0,0,0,0,0,0,0,0,0

m=application 52718 UDP/DTLS/SCTP webrtc-datachannel
b=AS:500
a=sctp-port:5002
a=max-message-size:1024
a=fingerprint:SHA-1 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id: abc3de65cddef001be82
a=dcmap:110 subprotocol="mpeg-sd"
a=mid:4
```

Annex B (informative): Examples of adaptation scenarios

B.1 Video bitrate adaptation

It is recommended in clauses 7.3.3 and 10.3 that a video sender adapts its video output rate based on RTCP reports and TMMBR messages. The following examples illustrate the usage:

EXAMPLE 1 – Handover to a different cell:

1. A video session is established at 100kbps. 5kbps is allocated for RTCP and trr-int is set to 500 ms. This allows an MTSI client in terminal to send regular RTCP reports with an average 500 ms interval consuming less than 5 kbps for RTCP. At the same time it allows the MTSI client in terminal to send an early RTCP packet and then send the next one already after 800 ms instead of after 1 000 ms.
2. The receiver is now subject to a reduced bandwidth, e.g. 60 kbps, due to handover to a different cell. The network indicates the reduced bandwidth to the receiver. The receiver generates a TMMBR message to inform the sender of the new maximum bitrate, 60 kbps.
3. The sender receives the TMMBR message, adjusts its output bitrate and sends a TMMBN message back.
4. The receiver sends a SIP UPDATE message to the sender indicating 60 kbps
5. The receiver travels into an area with full radio coverage. A new bandwidth of 100 kbps is negotiated with the network. It sends a SIP UPDATE message for 100 kbps.
6. The sender receives the SIP UPDATE message, and adjusts its output bitrate.

EXAMPLE 2 – Bad coverage or congestion:

1. A video session is established at 100kbps. 5kbps is allocated for RTCP and trr-int is set to 500 ms. This allows an MTSI client in terminal to send regular RTCP reports with an average 500 ms interval consuming less than 5 kbps for RTCP. At the same time it allows the MTSI client in terminal to send an early RTCP packet and then send the next one already after 800 ms instead of after 1 000 ms.
2. The receiver detects congestion and estimates a preferred video transmission rate of e.g. 60 kbps. The receiver generates a TMMBR message to inform the sender of the new maximum bitrate, 60 kbps.
3. The sender receives the TMMBR message, adjusts its output bitrate and sends a TMMBN message back.

Annex C (informative): Example adaptation mechanisms for speech

C.1 Example of feedback and adaptation for speech and video

C.1.1 Introduction

This annex gives the outline of possible example adaptation implementations that make use of adaptation signalling for speech as described in section 10.2. Several different adaptation implementations are possible and the examples shown in this section are not to be seen as a set of different adaptive schemes excluding other designs. Implementers are free to use these examples or to use any other adaptation algorithms. The examples are based on measuring the packet loss rate (PLR) but Annex C.1.3.1 describes how the measured frame loss rate (FLR) can be used instead of the PLR. A real implementation is free to use other adaptation triggers. The purpose of the section is to show a few different examples of how receiver state machines can be used both to control the signalling but also to control the signalling requests. Notice that the MTSI clients can have different implementations of the adaptation state machines.

The annex is divided into three sections:

- Signalling considerations - Implementation considerations on the signalling mechanism; the signalling state machine.
- Adaptation state machines - Three different examples of adaptation state machines either using the full set of adaptation dimensions or a subset thereof.
- Other issues and solutions - Default actions and lower layer triggers.

In this annex, a *media receiver* is the receiving end of the media flow, hence the *request sender* of any adaptation request. A *media sender* is the sending entity of the media, hence the *request receiver* of the adaptation request. The three different adaptation mechanisms available; bit-rate, packet-rate and error resilience, represents different ways to adapt to current transport characteristics:

- Bit-rate adaptation. Reducing the bit-rate is in all examples shown in this section the first action done whenever a measurement indicating that action is needed to further optimize the session quality. A bit-rate reduction will reduce the utilization of the network resources to transmit the data. In the radio case, this would reduce the required transmission power and free resources either for more data or added channel coding. It is reasonable to assume, also consistent with a proper behaviour on IP networks, that a reduction of bit-rate is a valid first measure to take whenever the transport characteristics indicate that the current settings of the session do not provide an optimized session quality.
- Packet-rate adaptation. In some of the examples, packet-rate adaptation is a second measure available to further adapt to the transport characteristics. A reduction of packet rate will in some cases improve the session quality, e.g. in transmission channels including WLAN. Further, a reduction of packet rate will also reduce the protocol overhead since more data is encapsulated into each RTP packet. Although robust header compression (RoHC) can reduce the protocol overhead over the wireless link, the core network will still see the full header and for speech data, it consists of a considerable part of the data transmitted. Hence, packet-rate adaptation serves as a second step in reducing the total bit-rate needed for the session.
- Error resilience. The last adaptive measure in these examples is the use of error resilience measures, or explicitly, application level redundancy. Application level redundancy does not reduce the amount of bits needed to be transmitted but instead transmit the data in a more robust way. Application level redundancy should only be seen as a last measure when no other adaptation action has succeeded in optimizing the session quality sufficiently well. For most normal use cases, application level redundancy is not foreseen to be used, rather it serves as the last resort when the session quality is severely jeopardized.

C.1.2 Signalling state considerations

The control of the adaptation signalling can by itself be characterized as a state machine. The implementation of the state machine is in the decoder and each MTSI client has its own implementation. The decoder sends requests as described in clause 10.2 to the encoder in the other end.

The requests that are transmitted can be queued up in a send buffer to be transmitted the next time an RTCP-APP packet is to be sent. Hence, a sender might receive one, two or all three receiver requests at the same time. It should not expect any specific order of the requests. A receiver shall not send multiple requests of the same type in the same RTCP-APP packet. Transmission of the requests should preferably be done immediately using the AVPF early mode but in some cases it may be justified to delay the transmission a limited time or until the next DTX period in order to minimize disturbance on the RTP stream, in the latter case monitoring of the RTP stream described below must take the additional delay into account.

To summarize:

- A request can be sent immediately (alone in one RTCP-APP packet) but the subsequent RTCP-APP packet must follow the transmission rules for RTCP.
- RTCP-APP packets may be delayed until the next DTX period.

Reception of the transmitted RTCP-APP packets is not guaranteed. Similar to the RTP packets, the RTCP packets might be lost due to link losses. Monitoring that the adaptation requests are followed can to be done by means of inspection of the received RTP stream.

For various reasons the requests might not be followed even though they received successfully by the other end. This behaviour can be seen in the following ways:

- Request completely ignored: An example is a request for 1 frame/packet which might be rejected as the MTSI client decides that the default mode of operation 2 frames/packet or more and a frame aggregation reduction compared to the default state is not allowed.
- Request partially followed: An example here is when no redundancy is received and a request for 100 % redundancy with 1 extra frame offset is made which may be realized by the media sender as 100 % redundancy with no extra offset. Another example is when a request for 5.9 kbps codec rate is sent and it is realized as e.g. 6.7 kbps codec rate. Table C.1 displays how the requests and realizations are grouped. E.g. it can be seen (if $N_{init} = 1$) that a request for 3 frames per packets realized as 2 frames per packet is considered to be fulfilled.

Table C.1: Distinction of different settings for frame aggregation, redundancy and codec mode settings

| Codec rate | Frame aggregation | Redundancy |
|--------------------------|-------------------------------------|--|
| Highest rate in mode set | N_{init} frame per packet | No redundancy |
| All other codec rates | $\geq N_{init}+1$ frames per packet | ≥ 100 % redundancy , arbitrary offset |

In table C.1 above N_{init} is 1 in most cases which corresponds to 1 frame per packet. In certain cases N_{init} might have another value, one such example is E-GPRS access where N_{init} may be 2. N_{init} is given by theptime SDP attribute.

NOTE: Special care in the monitoring should be taken when DTX is used as DTX SID update packets are normally not aggregated or transmitted redundant. Important is also that it takes at least one roundtrip before the effect of a request is seen in the RTP flow, if transmission of RTCP is delayed due to e.g. bandwidth requirements this extra delay must also be taken into account in the monitoring.

If the requests are not followed as requested, the request should not be repeated infinitely as it will increase the total bit-rate without clear benefit. In order to avoid such behaviour the following recommendations apply:

- Partially fulfilled requests should be considered as obeyed.
- If a new request is not fulfilled within $T_RESPONSE$ ms, the request is repeated again with a delay between trials of $2 * T_RESPONSE$ ms. If the three attempts have been made without sender action, it should be assumed that the request cannot be fulfilled. In this case, the adaptation state machine will stay in the previous state or in a state that matches the current properties (codec mode, redundancy, frame aggregation). Any potential mismatch

between define states in the adaptation state machine and the current properties of the media stream should be resolved by the request sender.

- The default mode of operation for a MTSI client if the RTCP bandwidth for the session is greater than zero is that the requests received should be followed. Ignoring requests should be avoided as much as possible. However, it is required that any signalling requests are aligned with the agreed session parameters in the SDP.

In some cases the adaptation state machine may go out-of-synch with the received RTP stream. Such cases may occur if e.g. the other MTSI client makes a reset. These special cases can be sensed, e.g. through a detection of a large gap in timestamp and/or sequence number. The state machine should then reset to the default state and start over again.

The signalling state machine has three states according to table C.2.

Table C.2: Signalling state machine states

| State | Description |
|-------|--|
| T1 | Idle state: This is the default state of the signalling state machine. The signalling state should always return here after a state transition and when it has been detected that the media sender has followed the request, either completely or partially. The signalling state machine remains in this state as long as the selected adaptation is "stable", i.e. as long as the adaptation measures are appropriate for the current operating conditions. When it has been detected that the operating conditions has changed so much that the current adaptation measures are no longer appropriate then the adaptation function triggers a request signalling and the signalling state machine goes to state T2. |
| T2 | In this state, the received RTP stream is monitored to verify that the properties of a given adaptation state (redundancy, frame aggregation and codec mode) are detected in the received RTP stream. If necessary, some of the requests are repeated maximum 3 times. If any of the properties is considered to be not fulfilled, the signalling state machine enters state T3. |
| T3 | In this state, the properties of the RTP stream (redundancy, frame aggregation and codec rate) is reverted back to the properties of the last successful state and a new state transition is tested in T2, or alternatively the adaptation state is set to the state that matches the current properties (codec mode, redundancy, frame aggregation). |

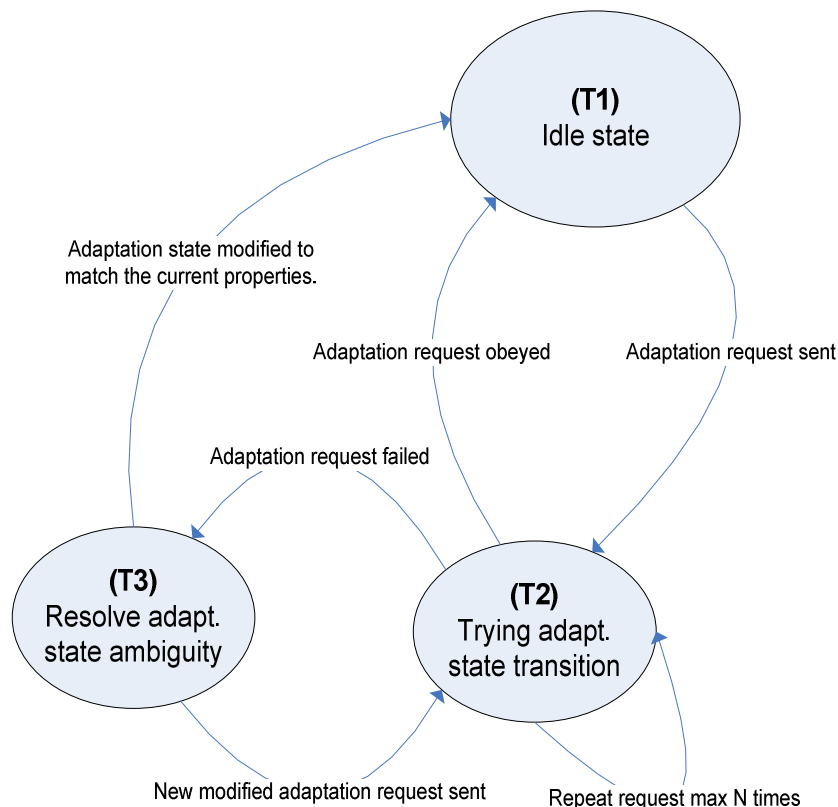


Figure C.1: Signalling state machine, implemented in order to ensure safe adaptation state transitions

C.1.3 Adaptation state machine implementations

C.1.3.1 General

The example adaptation state machines shown in this section are different realizations of the control algorithm for the adaptation requests. Note that this does not include how the actual signalling should be done but how various triggers will result in the transmission of different requests.

The example adaptation state machines make use of the signalling state machine outlined in clause B.2. Common to all adaptation state machines is that it is possible to implement all versions in the same code and just exclude appropriate states depending on desired mode of operation. All examples can transit between a number of states (denoted S1...S4). In these examples, it is assumed that the codec is AMR-NB and that it uses two coding rates (AMR 12.2 and AMR 5.9). However, this is not a limitation of the adaptation mechanism by itself. It is only the scenario used in these examples.

Since the purpose of the adaptation mechanism is to improve the quality of the session, any adaptation signalling is based upon some trigger; either a received indication or a measurement. In the case of a measurement trigger, it is important to gather reliable statistics. This requires a measurement period which is sufficiently long to give a reliable estimation of the channel quality but also sufficiently short to enable fast adaptation. For typical MTSI scenarios on 3GPP accesses, a measurement period in the order of 100 packets is recommended. Further, in order to have an adaptation control which is reliable and stable, a hangover period is needed after a new state has been entered (typically 100 to 200 packets). An even longer hangover period is suitable when transiting from an error resilient state or a reduced rate into the default, normal state. In the below examples, it is assumed that the metric used in the adaptation is the packet loss rate measured on the application layer. It is possible to use other metrics such as lower layer channel quality metrics.

NOTE: Mode change requests must follow the rules outlined in clause 5.2.1.

The example solution is designed based on the following assumptions:

- When the packet loss rate increases, the adaptation should:
 - First try with a lower codec mode rate, i.e. bit-rate back off.
 - If this does not improve the situation, then one should try with packet rate back-off by increasing the frame aggregation.
 - If none of these methods help, then application layer redundancy should be added to save the session.
- When the packet loss rate increases, one should try to increase the bit rate in a "safe" manner. This is done by probing for higher bit rates by adding redundancy.
- The downwards adaptation, towards lower rates and redundancy, should be fast while the upwards adaptation should be slow.
- Hysteresis should be used to avoid oscillating behaviour between two states.

A description of the different states and what trigger the transition into the respective state is given in table C.3.

Table C.3: Adaptation state machine states and their meaning

| State | Description |
|-------|--|
| S1 | Default/normal state: Good channel conditions. This state has the properties: <ul style="list-style-type: none"> - Codec rate: Highest mode in mode set. - Frame aggregation: Equal to the ptime value in the agreed session parameters. - Redundancy: 0%. |
| S2 | In this state the encoding bit-rate and the packet rate is reduced. The state is divided into 2 sub states (S2a and S2b). In state S2a the codec rate is reduced and in state S2b the packet rate is also reduced (the frame aggregation is increased). State S2a may also involve a gradual decrease of the codec-rate in order to be in agreement with the session parameters. If no restrictions are in place regarding mode changes (i.e. such as only allowing changing to a neighbouring mode), it changes bit-rate to the target reduced bit-rate directly. If restrictions are in place, several mode changes might be needed. This state has the properties: <ul style="list-style-type: none"> - Codec rate: Any codec rate except the highest rate in mode set, preferably a codec rate that is roughly half the highest rate. - Frame aggregation: <ul style="list-style-type: none"> - S2a: Equal to the ptime value in the agreed session parameters. - S2b: ptime+N*20ms where N > 1, limited by max-ptime. - Redundancy: 0%. |
| S3 | This is an interim state where the total bit-rate and packet rate is roughly equal to state S1. 100% redundancy is used with a lower codec mode than S1. This is done to probe the channel band-width with a higher tolerance to packet loss to determine if it is possible to revert back to S1 without significantly increase the packet loss rate. This state has the properties: <ul style="list-style-type: none"> - Codec rate: Any codec rate except the highest rate in mode set, preferably a codec rate that is roughly half the highest rate, target total rate (with redundancy) should be roughly the same as in S1. - Frame aggregation: Equal to the ptime value in the agreed session parameters. - Redundancy: 100%. |
| S4 | In this state the encoding bit-rate is reduced (the same bit-rate as in S2) and redundancy is turned on. Optionally also the packet rate is kept the same as in state S2. This state has the properties: <ul style="list-style-type: none"> - Codec rate: Any codec rate except the highest rate in mode set, preferably a codec rate that is roughly half the highest rate. - Frame aggregation: Equal to the ptime value in the agreed session parameters. - Redundancy: 100%, possibly with offset. |

The parameters and other definitions controlling the behaviour of the adaptation state machine are described in table C.4. Example values are also shown, values which give good performance on a wide range of different channel conditions.

Table C.4: State transition definitions, thresholds and temporal adaptation control parameters

| Parameter | Value/meaning | Comment |
|-------------------|---|---|
| PLR_1 | 3 % | |
| PLR_2 | 1 % | |
| PLR_3 | 2 % | |
| PLR_4 | 10 % | |
| N_INHIBIT | 1 000 frames | A random value may be used to avoid large scale oscillation problems. |
| N_HOLD | 5 measurement periods | |
| T_RESPONSE | 500 ms | Estimated response time for a request to be fulfilled. |
| Packet loss burst | 2 or more packet losses in the last 20 packets. | |

As described in Annex C.1.1, the frame loss rate (FLR) can be used instead of the packet loss rate to trigger the adaptation. The benefit with using FLR is that this metric can (and should) include the late losses that occur if frames are received too late to be useful for decoding. Table C.4a shows thresholds that can be used for FLR if the FLR-triggered adaptation is used instead of the PLR-triggered adaptation.

Table C.4a: FLR thresholds when using the frame loss rate to control the adaptation

| Parameter | Value/meaning | Comment |
|------------------|---|----------------------------|
| FLR_1 | 3 % | Used instead of PLR_1 |
| FLR_2 | 1 % | Used instead of PLR_2 |
| FLR_3 | 2 % | Used instead of PLR_3 |
| FLR_4 | 3 % | Used instead of PLR_4 |
| Frame loss burst | 2 or more frame losses in the last 20 frames. | Replaces packet loss burst |

The adaptation state machines shown in Annex C.1.3.2, C.1.3.3 and C.1.3.4 can be used also for FLR-triggered adaptation by applying the following modifications:

- The media receiver needs to measure the frame loss rate instead of the packet loss rate. The frame loss rate includes late losses.
- The PLR thresholds need to be replaced with the corresponding FLR thresholds, as shown in Table C.4a.

The state machines are otherwise the same.

C.1.3.2 Adaptation state machine with four states

The first example utilizes all adaptation possibilities, both in terms of possible states and transitions between the states. Figure C.2 shows the layout of the adaptation state machine and the signalling used in the transitions between the states.

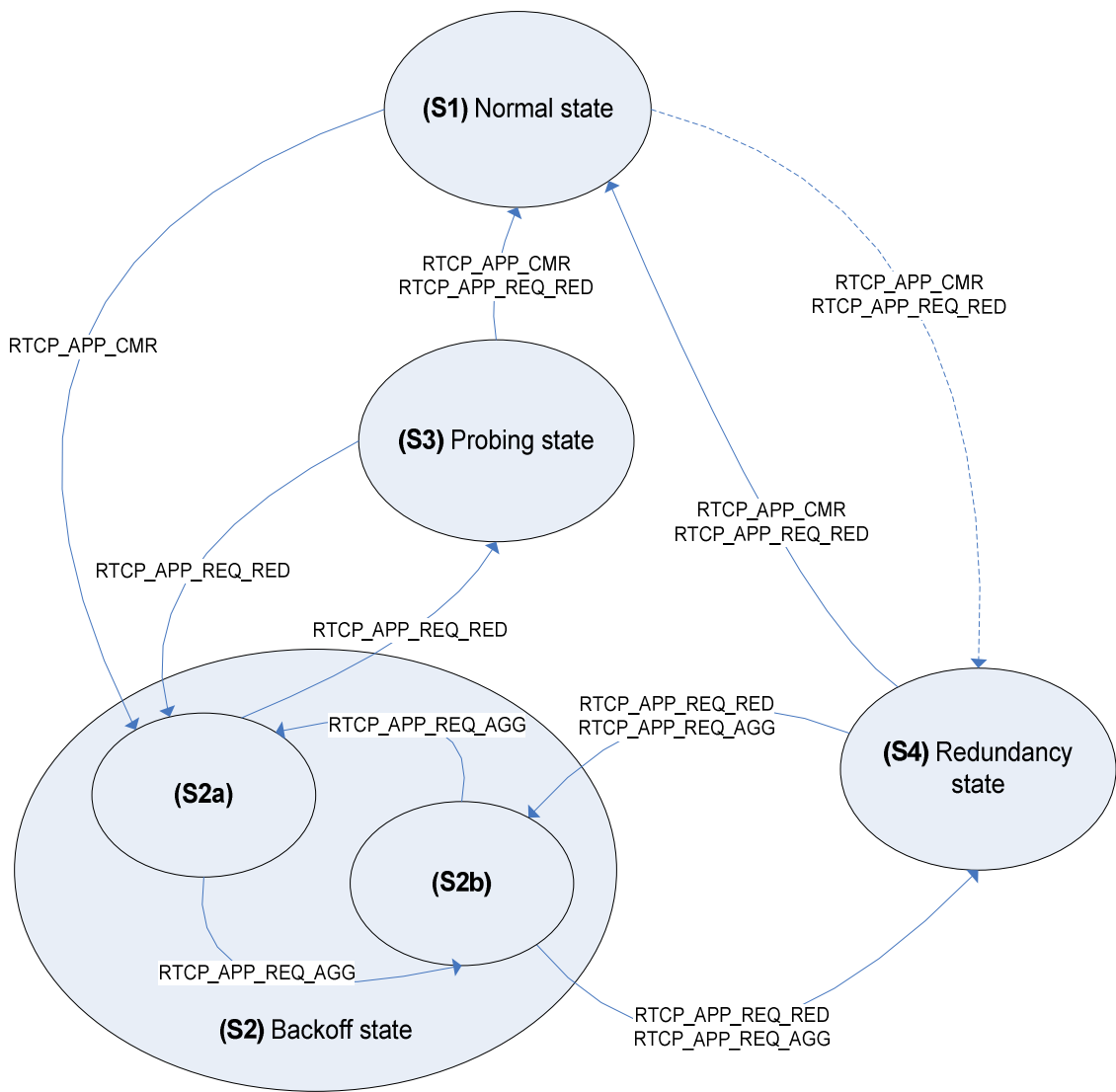


Figure C.2: State diagram for four-state adaptation state machine

State transitions:

Below are listed the possible state transitions and signalling that is involved. Note that the state can go from S1 to either S2 or state S4, this is explained below:

Table C.5: State transitions for four-state adaptation state machine

| State transition | Conditions and actions |
|------------------|--|
| S1 → S2a | Condition: Packet loss \geq PLR_1 or packet loss burst detected. A request to reduce the encoding bit-rate is sent using RTCP_APP_CMР, e.g. change mode from AMR 12.2 to AMR 5.9. |
| S2a → S2b | Condition: Packet loss \geq PLR_1. This state transition occurs if the packet loss is still high despite the reduction in codec rate. A request is sent to reduce the packet rate is reduced by means of an RTCP_APP_REQ_AGG message. |
| S2b → S2a | Condition: Packet loss \leq PLR_2 for N_HOLD consecutive measurement periods. This state transition involves an increase of the packet rate restoring it to the same value as in S1. The request transmitted is RTCP_APP_REQ_AGG. If the state transition S2b→S2a→S2b occurs in sequence, the state will be locked to S2b for N_INHIBIT frames to avoid state oscillation. |
| S2a → S3 | Condition: Packet loss \leq PLR_2 for N_HOLD consecutive measurement periods. A request to turn on 100% redundancy is transmitted by means of request RTCP_APP_REQ_RED. |
| S3 → S2a | Condition: Packet loss \geq PLR_3. Same actions as in transition from, S1→S2a. If the transition S2a→S3→S2a→S3→S2a occurs, the S3 is disabled for N_INHIBIT frames. |
| S3 → S1 | Condition: Packet loss \leq PLR_2 for N_HOLD consecutive measurement periods. A request to turn off redundancy is transmitted as RTCP_APP_REQ_RED. Encoding bit-rate is increased by means of RTCP_APP_CMР. |
| S2b → S4 | Condition: Packet loss \geq PLR_3. A request to turn on 100% redundancy is transmitted by means of request RTCP_APP_REQ_RED. The packet rate is restored to same value as in S1 using RTCP_APP_REQ_AGG. |
| S4 → S2b | Condition: 1. If the previous transition was S2b→S4 and packet loss \geq to 4*PLR@ S2b→S4 (packet loss considerably increased since transition to state S4). This is indicative of that the total bit-rate is too high and that it is probably better to transmit with a lower packet rate/bit-rate instead. This case might occur if the packet loss is high in S2a due to a congested link, a switch to redundant mode S4 will then increase the packet loss even more 2. If previous transition was S1→S4 and packet loss \geq PLR_4. This transition is made to test if a bitrate/packet rate reduction is better. |
| S4 → S1 | Condition: Packet loss $<$ PLR_3 for N_HOLD consecutive measurement periods. A request to turn off redundancy is transmitted using RTCP_APP_REQ_RED. Encoding bit-rate is requested to increase using RTCP_APP_CMР. |
| S1 → S4 | Condition: Packet loss \geq PLR_1 or packet loss burst detected AND the previous transition was S4→S1, otherwise the transition S1→S2a will occur. A request to turn on 100% redundancy is transmitted using RTCP_APP_REQ_RED. The encoding bit-rate is requested to be reduced (in the example from AMR 12.2 to AMR 5.9) using RTCP_APP_CMР. |

C.1.3.3 Adaptation state machine with four states (simplified version without frame aggregation)

This example is a simpler implementation with the frame aggregation removed.

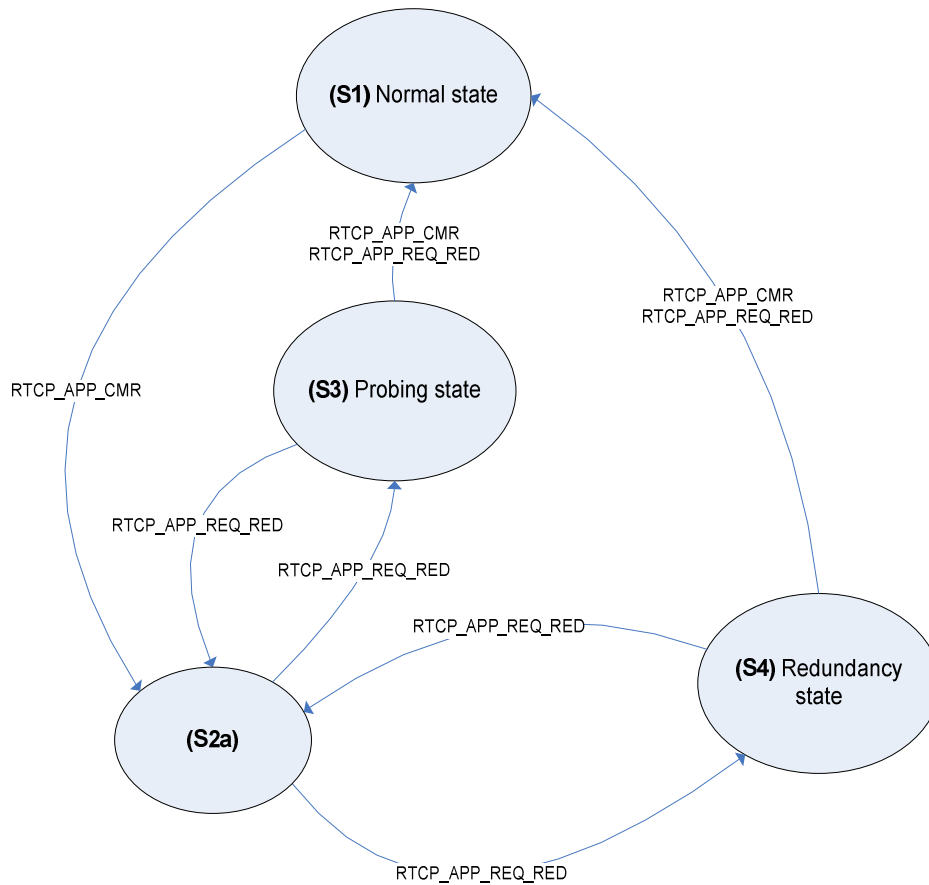


Figure C.3: State diagram for simplified four-state adaptation state machine

State transitions:

Below are listed the possible state transitions and signalling that is involved.

Table C.6: State transitions for simplified four-state adaptation state machine

| State transition | Conditions and actions |
|------------------|---|
| S1 → S2a | Condition: Packet loss \geq PLR_1 or packet loss burst detected. A request to reduce the encoding bit-rate is sent using RTCP_APP_CMV, e.g. change mode from AMR 12.2 to AMR 5.9. |
| S2a → S3 | Condition: Packet loss \leq PLR_2 for N_HOLD consecutive measurement periods. A request to turn on 100% redundancy is transmitted by means of request RTCP_APP_REQ_RED. |
| S3 → S2a | Condition: Packet loss \geq PLR_3. Same actions as in transition from, S1→S2a. If the transition S2a→S3→S2a→S3→S2a happens in sequence state S3 is disabled for N_INHIBIT frames. |
| S3 → S1 | Condition: Packet loss \leq PLR_2 for N_HOLD consecutive measurement periods. A request to turn off redundancy is transmitted as RTCP_APP_REQ_RED. Encoding bit-rate is increased by means of RTCP_APP_CMV. |
| S2a → S4 | Condition: Packet loss \geq PLR_3. A request to turn on 100% redundancy is transmitted by means of request RTCP_APP_REQ_RED. |
| S4 → S2a | Condition: Packet loss \geq to 4*PLR@ S2b→S4 (packet loss considerably increased since transition to state S4). This is indicative of that the total bit-rate is too high and that it is probably better to transmit with a lower packet rate/bit-rate instead. This case might occur if the packet loss is high in S2a due to a congested link, a switch to redundant mode S4 will then increase the packet loss even more. |
| S4 → S1 | Condition: Packet loss $<$ PLR_3 for N_HOLD consecutive measurement periods. A request to turn off redundancy is transmitted using RTCP_APP_REQ_RED. Encoding bit-rate is requested to increase using RTCP_APP_CMV. |

C.1.3.4 Adaptation state machine with two states

This example is an implementation with the redundant states removed.

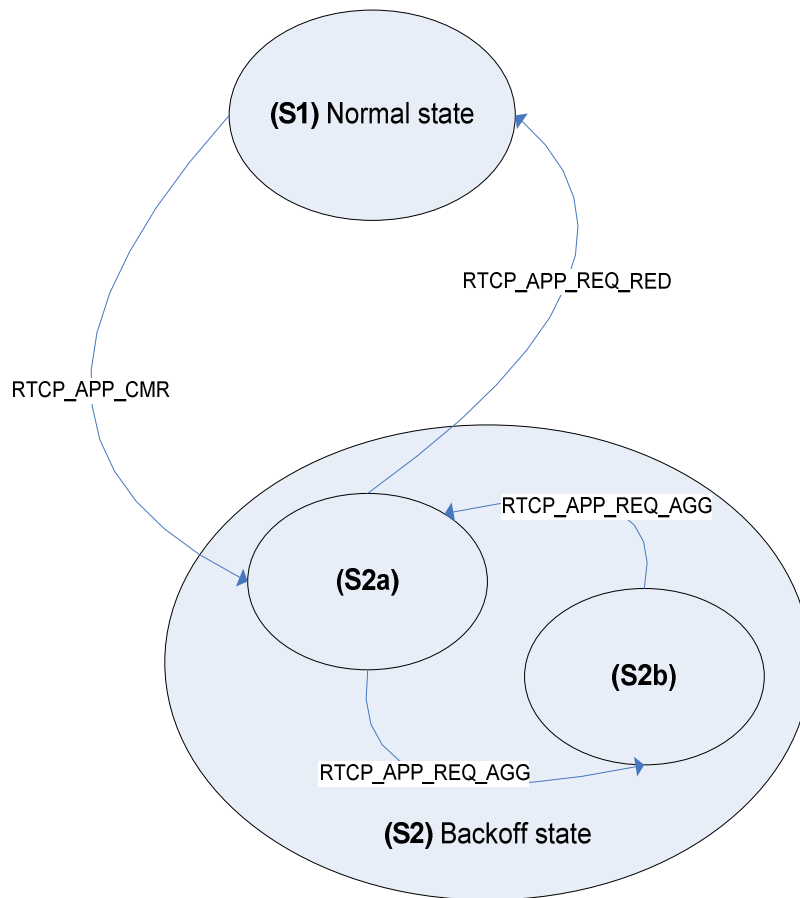


Figure C.4: State diagram for two-state adaptation state machine

State transitions:

Below are listed the possible state transitions and signalling that is involved.

Table C.7: State transitions for two-state adaptation state machine

| State transition | Conditions and actions |
|------------------|--|
| S1 → S2a | Condition: Packet loss ≥ PLR_1 or packet loss burst detected. A request to reduce the encoding bit-rate is sent using RTCP_APP_CMR, e.g. change mode from AMR 12.2 to AMR 5.9. A failed transition counter counts the number of consecutive switching attempts S2a→S1 that fails. In the number of failed attempts is two or more state S1 is inhibited for N_INHIBIT frames. A failed transition attempt occurs if the previous transition was S2a→S1 and the state transition immediately occurs back to S2a. |
| S2a → S2b | Condition: Packet loss ≥ PLR_1. This state transition occurs if the packet loss is still high despite the reduction in codec rate. A request to reduce the packet rate is reduced by means of an RTCP_APP_REQ_AGG message. |
| S2b → S2a | Condition: Packet loss ≤ PLR_2 for N_HOLD consecutive measurement periods. This state transition involves an increase of the packet rate. Also packet rate is restored to same value as in State (1) RTCP_APP_REQ_AGG. If the state transition S2b→S2a→S2b occurs in sequence, the state will be locked to S2b for N_INHIBIT frames. |
| S2a → S1 | Condition: Packet loss ≤ PLR_2 for N_HOLD consecutive measurement periods. Redundancy is turned on (100 %) by means of request RTCP_APP_REQ_RED. |

C.1.3.5 Adaptation when using ECN

This example shows how ECN may be used to trigger media bit-rate adaptation. ECN can be used in combination with other adaptation triggers, for example packet loss triggered adaptation schemes or frame loss rate triggered adaptation schemes, although this is not included in this example.

In this example, the ECN-triggered adaptation is configured using the set of parameters as described in Table C.8.

Table C.8: Configuration parameters used for the ECN triggered adaptation in this example

| Parameter | Description |
|---------------------|------------------------------------|
| ECN_min_rate | Used in accordance with Table 10.1 |
| ECN_congestion_wait | Used in accordance with Table 10.1 |

Figure C.5 shows how the codec rate changes over the session if there is no congestion and therefore no ECN-CE marking (and no packet losses). The codec modes that can be used during the session are negotiated at session setup. In this case it is assumed that the recommended four AMR {4.75, 5.9, 7.4 and 12.2 kbps} codec modes can be used in the session. It is further assumed that the highest codec mode allowed in the session is AMR 12.2 kbps and the ECN_min_rate corresponds to AMR 5.9 kbps, see Clause 10.2.0.

NOTE: ECN can also be used for AMR-WB in a corresponding way, with the difference that the highest codec mode and ECN_min_rate would be selected based on the AMR-WB codec mode rates.

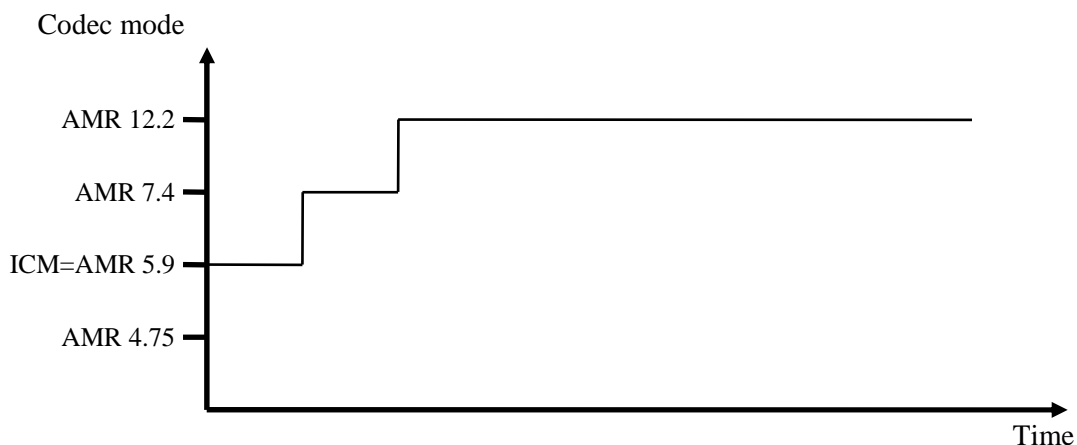


Figure C.5: Example of codec mode usage in a session

The session starts with the Initial Codec Mode (ICM), i.e. the AMR 5.9 kbps codec mode, see Clause 7.5.2.1.6. The receiving MTSI client evaluates the performance, for example by measuring the packet loss rate and detecting ECN-CE marked packets, and adapts the codec mode upwards (by sending adaptation requests backwards to the sender) in steps as long as no ECN-CE marked packets and no (or only marginal) performance problems are detected. In this case, this means that the MTSI client starts using the AMR 5.9 kbps mode, then switches to the AMR 7.4 kbps mode and then to the AMR 12.2 kbps mode.

The step-wise upswitching is used because the receiving MTSI client does not know whether the new and higher rate is sustainable or not. The transmission performance for each new rate needs to be verified over a time period before further upswitching can be attempted. If the new bit rate would prove to be not sustainable then the receiving MTSI client would switch back to the previously used rate or even a lower rate (not shown in these figures).

Figure C.6 shows how ECN-CE marked packets may trigger codec adaptation.

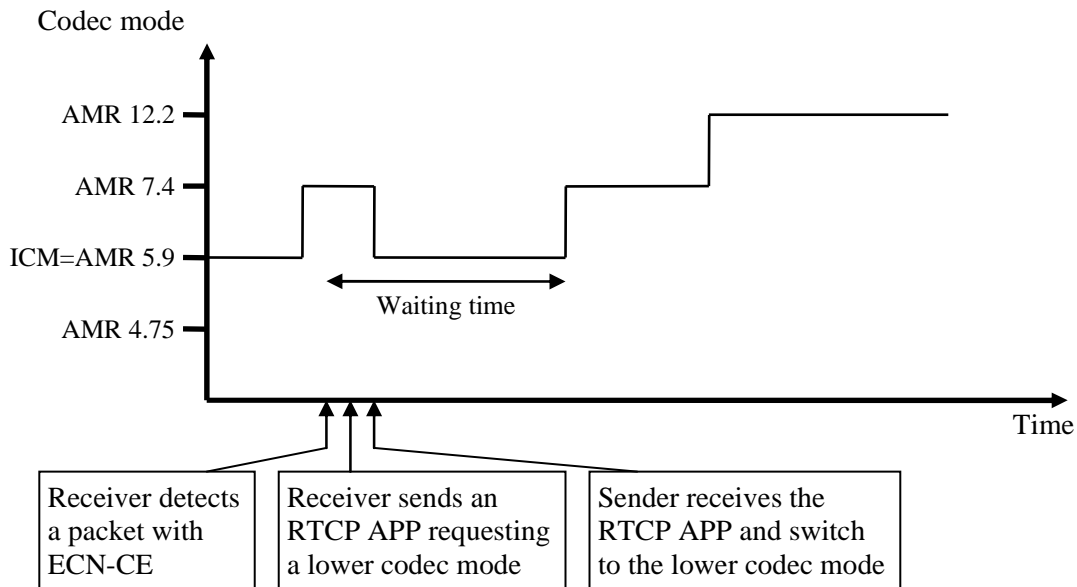


Figure C.6: Example of how ECN may trigger codec adaptation

Again, the session starts with ICM, i.e. the AMR 5.9 kbps codec mode. The MTSI client evaluates the performance, for example the packet loss rate and/or ECN-CE marked packets, and adapts the codec mode upwards if it is deemed possible to do so. During the upwards adaptation, the receiver detects in this example a congestion event since ECN-CE is set for at least one of the received IP packets. In this case a fast back-off strategy is used and the receiver therefore sends an adaptation request back to the sender using RTCP APP with a request to switch to a low codec mode, in this case to adapt to the AMR 5.9 kbps mode. The AVPF profile allows for sending an (one) RTCP packet without waiting for the normal RTCP transmission interval, even if a regular compound RTCP was recently transmitted. This gives a faster reaction to ECN-CE.

After the down-switch, a waiting time is used to prevent upswitch too soon after the congestion event since too early upswitch is likely to trigger further congestion. The receiver uses RTCP APP also for the adaptation requests for upswitch. It is beneficial to use the normal RTCP transmission rules, defined for the AVP profile, for the upswitch adaptation signalling because this enables using the AVPF transmission rules in case congestion would occur immediately after the upswitch.

The response to the ECN-CE marking, the waiting time and the upswitching after a congestion event is the same regardless of when the congestion occurs, which is shown in Figure C.7. This is because the MTSI client is evaluating if the bit rate is sustainable also after switching up to the high bit rate.

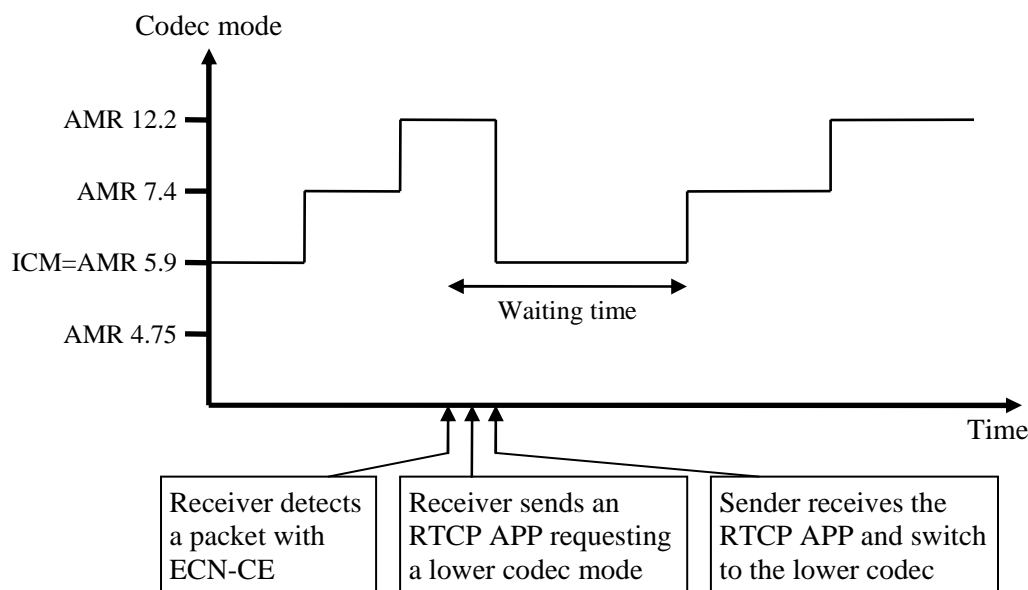


Figure C.7: Example of codec mode usage in a session

Figure C.7 also shows how the fast down-switch gives a rapid codec mode switch from the AMR 12.2 kbps to the AMR 5.9 kbps mode. The codec mode request (CMR) sent from the receiver may suggest a direct switch from the AMR 12.2 kbps mode to the AMR 5.9 kbps mode. However, if the MTSI client is inter-working with a CS GERAN then mode changes will be limited in the session setup to every other frame border and also to neighboring modes. The MTSI client obviously has to follow such rules for mode changes, if they are defined in the session setup. The sender may therefore be prevented from following the CMR directly and it may take a few frames until the target codec mode is reached.

C.2 Example criteria for video bit rate feedback and adaptation

C.2.1 Introduction

This annex gives the outline of possible example adaptation criteria that make use of adaptation signalling for video as described in section 10.3. Several different adaptation implementations are possible and the example criteria shown in this section are not to be seen as an adaptive scheme excluding other designs. Implementers are free to use these example criteria or to use any other adaptation algorithm as long as the requirements and recommendations specified in clause 10.3 are fulfilled. The description of the example criteria is split into two parts, one for the media sender side and one for the media receiver side.

C.2.2 Media sender side

The basic rate adaptation algorithm on the media sender side serves to combine the received RTCP TMMBR and RTCP Receiver Report or Sender Report in a way that makes adaptation possible in the presence of any or both of the aforementioned reports. One important aspect is that the TMMBR reports will serve as an upper limit on the permitted bitrate while RTCP Receiver or Sender Reports provide a means for temporary adjustments based on e.g. the packet loss rate in a given interval. Note that the actual bitrate limit will also depend on the bandwidth attribute in the SDP. Typically adjustments of the permitted bitrate due to TMMBR reports are less frequent than adjustments due to RTCP Receiver or Sender Reports. The media sender may use the following conditions to adjust its video transmission rate:

- 1 Upon receiving a TMMBR message the media sender sets its maximum transmission rate to the requested rate.

- 2 If the requested rate in the TMMBR message is greater than the current video transmission rate the media sender (gradually) increases its transmission rate to the requested rate in the TMMBR message.
- 3 Examining the RTCP Receiver Report and Sender Report information to determine whether finer adaptations can be made to the video transmission rate. For example, the media sender may reduce its video transmission rate in response to an increase in the packet loss rate.
- 4 Reducing the video transmission rate if the media sender determines that the local radio uplink throughput is decreasing, e.g. detecting congestion in the uplink transmission buffers or examining other indicators of uplink quality.
- 5 Other conditions

C.2.3 Media receiver side

The basic rate adaptation algorithm on the media receiver side may consist of both the well-established RFC 3550 RTCP RR and SR reporting (which is not described further) and the estimation and sending of TMMBR to the sender. Transmission of TMMBR reports is typically less frequent than RTCP Receiver Reports or Sender Reports.

The media receiver may use the following conditions to send a TMMBR message requesting a reduction in video transmission rate

- 1 The video packets are arriving too close to or too late for their scheduled playout
- 2 The receiver detects an unacceptably high packet loss rate
- 3 The receiver detects that the received bitrate has been reduced
- 4 Other conditions

The MTSI media receiver may use the following conditions to send a TMMBR requesting an increase in video transmission rate

- 1 The video packets are arriving earlier than needed for their scheduled playout
- 2 Other conditions

Care must be taken when sending consecutive TMMBR messages to accommodate the media sender's reaction to previously sent TMMBR messages. When doing this, the media receiver should account for delays in the transmission of TMMBR messages due to RTCP bandwidth requirements.

C.2.4 Video encoder bitrate adaptation, down-switch

As described in clause 10.3.4.2, the video encoder may not be able to immediately change the sending bitrate to the requested bitrate, especially if this also requires changing the frame rate and/or the video resolution. During this period, the generated bitrate is higher than the channel capacity which means that excessive bits (*excess_bits*) are generated and the corresponding RTP packets will be queuing up at the node where the congestion occurs. Figure C.8 gives a simplified schematic description of the encoder bitrate adaptation. The encoder bitrate is here shown with straight lines. In reality, the amount of data generated by the encoder will vary from frame to frame and the bitrate will then vary around the bitrate shown in this figure. These bitrate variations are not considered in this description but needs to be considered in the real implementation.

The encoder uses the bitrate of the previous channel capacity (*prev_rate*) as long as the sender is unaware of the reduced channel capacity. When the TMMBR message is received, the encoder starts reducing the bitrate towards the new channel capacity (*new_rate*). Since the bitrate indicated in the TMMBR message includes the IP/UDP/RTP overhead then this needs to be removed. The encoder bitrate is then reduced even further for a while to gain back the delay caused by the queue build-up.

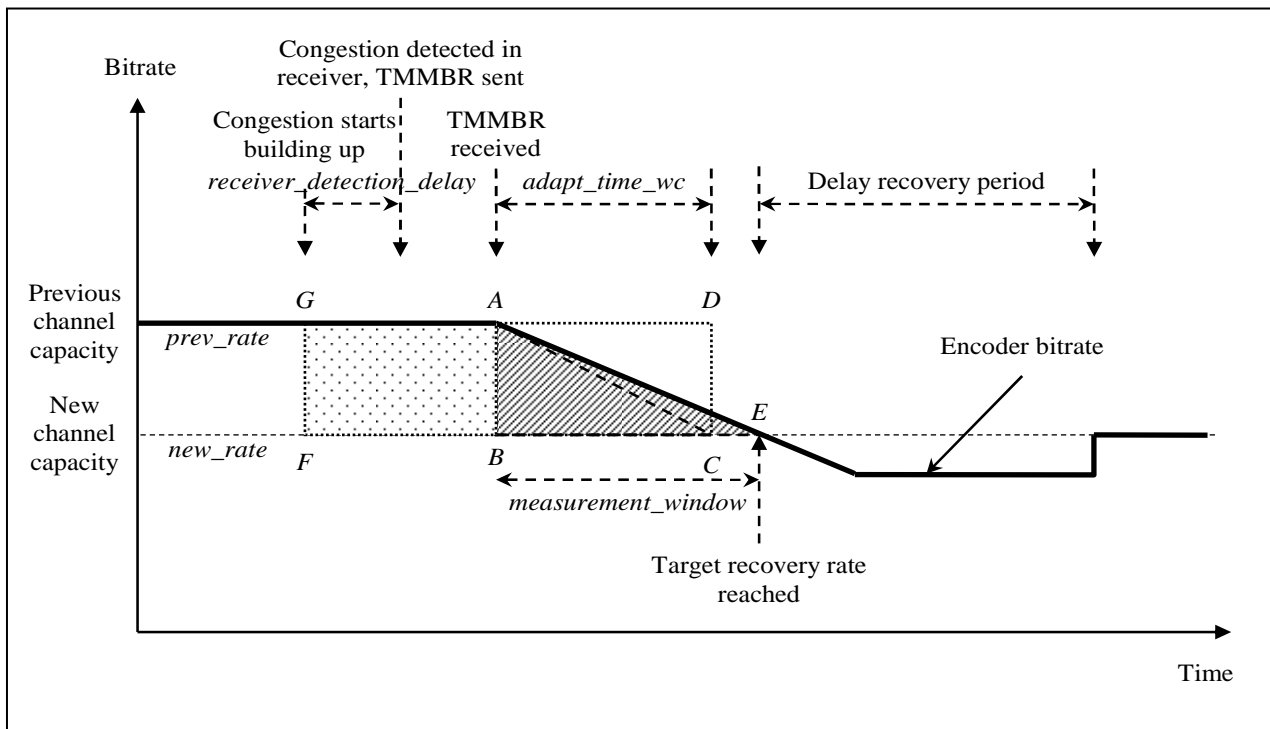


Figure C.8: Schematic figure of bitrate reduction in video encoder when the encoder cannot immediately switch to the requested bitrate

The upper limit requirement (*excess_bits_wc*) for how many excessive bits that is allowed to be generated during the adaptation time is defined by the Worst-Case Adaptation Algorithm, which corresponds to the rectangle *ABCD* while the recommendation ($0.5 * excess_bits_wc$) is defined by the triangle *ABE*.

The amount of excessive bits that are generated corresponds to the area of the triangle *ABE*. As can be seen in the figure, the measurement window (*measurement_window*) is here longer than the worst case adaptation time (*adapt_time_wc*) which is used for defining the requirement limit and the recommendation limit for the excessive bits. In this example, the encoder bitrate is not reduced fast enough to fulfil the recommendation but the requirement is fulfilled. This shows that it is allowed to use an adaptation time that is longer than the time used for defining the requirement and recommendation, as long as the amount of excessive bits does not exceed the limit.

After the encoder bitrate has been reduced to the new bitrate then the encoder needs to reduce the bitrate even further to form the beginning of the delay recovery period. The length of the delay recovery period depends on the amount of excessive bits that have been generated and how much lower the encoding bitrate is compared to the new channel capacity.

In reality, the queue starts to build up even before the sender has received the TMMBR message, which is here shown by the rectangle *ABFG*. The sender can estimate the excessive bits generated during this period from the RTCP Receiver Reports and can then compensate for this by extending the delay recovery period.

C.2.5 Video encoder bitrate adaptation, receiver-driven up-switch

When the channel is being under-utilized by the sender, it is likely that the delivery of video packets will occur before they actually need to be played out at the receiver. Therefore, the sender rate could be increased and introduce some additional delay without negatively affecting the system or the user experience.

The excess bits (*excess_bits*) that can be introduced into the transmission path can be computed as follows in the case that the channel bandwidth is equal to the average receiving rate measured at the receiver, i.e., the worst case with no spare channel bandwidth available:

$$excess_bits = rate_increase_step * (RTT + receiver_detection_delay) \quad (C.2.5-1)$$

where: $rate_increase_step$ is the bitrate with which the sending rate is increased;

RTT is the Round-Trip Time;

and: $receiver_detection_delay$ is the time needed to detect if congestion occurs, see Figure C.8.

The corresponding worst case excess delay ($excess_delay_wc$) due to $excess_bits$ equals:

$$excess_delay_wc = \frac{rate_increase_step * (RTT + receiver_detection_delay)}{avg_receiving_rate} \quad (C.2.5-2)$$

where: $avg_receiving_rate$ is the average throughput as measured by the receiver.

Therefore, if the receiver determines the amount of allowable excess delay ($excess_delay_allowable$) from the received video packets, it can calculate the amount of rate increase that would not congest the system as:

$$rate_increase_step = \frac{excess_delay_allowable * avg_receiving_rate}{(RTT + receiver_detection_delay)} \quad (C.2.5-3)$$

Since the one-way delay from sender to receiver is generally unknown to the receiver, it cannot use this to calculate the allowable excess delay. Instead the receiver measures the amount of time between when a packet arrives and when it is scheduled to be played out to determine how much additional delay is acceptable. This metric is actually more accurate from a user-experience perspective since this directly determines whether the video information in received packets can actually be displayed to the user without degradation.

The bitrate to request ($requested_rate$) in TMMBR is then:

$$requested_rate = prev_rate + rate_increase_step \quad (C.2.5-4)$$

Before sending the TMMBR message with the requested rate, the receiver needs to verify that the requested rate does not exceed the bitrate that was negotiated in SDP offer/answer.

C.2.6 Video encoder bitrate adaptation, recovery phase

The delay recovery rate ($delay_recovery_rate$) and delay recovery period ($delay_recovery_period$) can be determined as follows. Let

$$delay_recovery_rate = (1 - f_U) * new_rate \quad (C.2.6-1)$$

where: f_U ($0 < f_U < 1$) is the rate undershoot factor and may depend on the magnitude of the channel rate drop (ΔR) is:

$$\Delta R = prev_rate - new_rate \quad (C.2.6-2)$$

For example, if ΔR is large, then f_U may be proportionally large or if ΔR is small, then f_U may be proportionally small, for example:

$$f_U = \Delta R / prev_rate \quad (C.2.6-3)$$

Assuming that the bits during time period $\Delta T = timd(E) - timd(F)$ are queued up and are contributing to the delay, the delay recovery period ΔT_u can be computed as follows:

$$\Delta T_u = \Delta T (prev_rate - new_rate) / (f_U * new_rate) \quad (C.2.6-4)$$

A minimum bit rate requirement for the encoder may exist that applies to *delay_recovery_rate* and, therefore, also to f_U as follows:

$$\text{delay_recover_rate} \geq R_{min} \quad (\text{C.2.6-5})$$

and therefore:

$$f_U \leq 1 - (R_{min} / \text{new_rate}) \quad (\text{C.2.6-6})$$

with:

$$\text{new_rate} > R_{min} \quad (\text{C.2.6-7})$$

If during the delay recovery period a TMMBR message is received that carries a new rate value ($\hat{\text{new_rate}}$), and $\hat{\text{new_rate}}$ is significantly larger than new_rate , for example $\hat{\text{new_rate}} \geq 1.2 * \text{new_rate}$, then the delay recovery period may be shortened. Conversely, if $\hat{\text{new_rate}} < \text{new_rate}$ then an extended delay recovery period can be computed.

Annex D (informative): Reference delay computation algorithm

In this annex, the reference jitter management algorithm is described. It is written in pseudo code and is non-causal; hence non-implementable. The purpose of this algorithm is to define an "ideal" behaviour which all jitter buffers used in MTSI should strive to mimic. This buffer operates based on three input parameters:

- lookback factor to set the current target buffering depth;
- target late loss rate;
- maximum allowed time scaling percentage.

```
function ref_jb(channel, jb_adaptation_lookback, delay_delta_max, target_loss)
% channel          = file name of the channel
% lookback        = look back factor when estimating the max jitter
%                = buffer level [number of frames]
% delay_delta_max = max timescaling related modification (%) of the
%                = delay
% target_loss     = target late loss (%)
% example syntax:
% ref_jb('channel_1.dat', 200, 15, 0.5);

framelength = 20;
% this value sets the speech data in each RTP packet to 20 ms. For 2 speech
% frames/RTP packet the value would be 40 ms.
jitter_est_window=50;
% Sets the jitter estimation window in number of frames
delay_delta_max_ms = framelength*delay_delta_max*0.01;
% Sets the maximum allowed time scaling
tscale = 1;
% Scale factor of delay data
% In this case the files are assumed to be ascii files with one delay
% entry per line, the entries are in ms, a negative value denotes
% a packet loss.
x = load(channel);
x =x';
% remove packet losses
% remove initial startup empty frames
ix = find(x > 0);
x(1:ix(1)-1) = x(ix(1));
% remove packet losses (replace with nearby delay values)
ix = find(x < 0);
packet_loss = length(ix)/length(x)*100;
for n=1:length(ix)
    if (ix(n) > 1)
        x(ix(n)) = x(ix(n)-1);
    end;
end;
% convert timescale to ms
x = x*tscale;
L = length(x);
T = 1:L;
% estimate min and max TX delay, estimate a delta_delay
for n=1:L
    ix = [max(1, n-jitter_est_window):n];
    max_delay(n) = max(x(ix));
    min_delay(n) = min(x(ix));
    delta_delay(n) = max_delay(n)-min_delay(n);
end
% compute the target max jitter buffer level with some slow adaptation
% downwards, just to mimick how a jitter buffer might behave
for n=1:L
    ix = [max(1, n-jb_adaptation_lookback):n];
    jb(n) = max(delta_delay(ix));
    % The timescaling is not allowed to adjust the jitterbuffer target max level
    % too fast.
    if n == 1
        jb_ = jb(n);
    end
    delta = abs(jb_-jb(n));
    if delta < delay_delta_max_ms;
        jb_ = jb(n);
    end
end
```

```

else
    if (jb(n) < jb_)
        jb_ = jb_-delay_delta_max_ms;
    else
        jb_ = jb_+delay_delta_max_ms;
    end
    jb(n) = jb_;
end
% jitter buffer target max level can only assume an integer number of frames
jbq(n) = ceil(jb(n)/framelength)*framelength;
% compute estimated delay
del(n) = jbq(n)+min_delay(n);
end

if target_loss > 0
    % decrease the max jitter buffer level until a target late loss has been
    % reached.
    late_loss = length(find(del < x))/L*100.0;
    jbq_save = jbq; % as the max level is increased until the late loss > target one
    % must be able to revert back to the previous data
    while late_loss < target_loss
        jbq_save = jbq;
        jbq = min(max(jbq)-framelength,jbq);
        del = jbq+min_delay;
        late_loss = length(find(del < x))/L*100.0;
    end
    jbq = jbq_save;
    del = jbq+min_delay;
end

jdel = max(0,del-x);
%Calculate and plot the CDF of the reference buffer.
figure(1);plot(T,jbq,T,del,T,x);
[n,x] = hist(jdel,140); y = cumsum(n); y = y/max(y)*100;
figure(2);plot(x,y);axis([0 200 0 100]);ylabel('%');xlabel('ms');title('CDF of packet delay in JB');

```


Annex E (informative): QoS profiles

E.1 General

This annex contains examples with mappings of SDP parameters to QoS parameters [64] for MTSL.

The bitrates used in these QoS examples for MBR and GBR for MBR=GBR bearers and for MBR for MBR>GBR bearers are based on the highest bitrates possible with the codecs, profiles and levels defined in Clause 5.2. The bitrates used for GBR for MBR>GBR bearers are chosen to still give usable quality levels.

The 'a=bw-info' attributed defined in Clause 19 can be used to provide additional bandwidth information for the setting of MBR and GBR and also to align the resource allocation end-to-end, see also Clauses 6.2.5 and 6.2.7.

The bearer setup also depends on the outcome of the SDP offer-answer negotiation. The QoS profiles shown below assume that both end-points agree on using the codecs and bitrates as described in each respective example.

The QoS Class Identifier (QCI) [90] and 5G QoS Identifier (5QI) [176] are used to describe the packet forwarding treatment for different media types. The table below gives a few examples for how the QCI/5QI can be set for different media types.

Table E.0: Example mapping between media type and QCI.

| Media | Bearer type | QCI/5QI |
|----------------------------|---|--------------------------|
| Speech | Dedicated GBR bearer | 1 |
| Video (conversational) | Dedicated GBR bearer | 2 |
| Video (non-conversational) | Dedicated GBR bearer | 4 |
| | Dedicated non-GBR bearer or default non-GBR bearer | 6, 8 or 9 |
| Real-time text | Dedicated non-GBR bearer or default non-GBR bearer | 6, 8 or 9 |
| Data channel | Dedicated GBR or non-GBR, or default non-GBR bearer | 71, 72, 73, 74, 76, or 9 |

This mapping assumes that the QCIs are used as described in TS 23.203, Table 6.1.7, [90] for LTE access, and 5QIs as described in TS 23.501, Table 5.7.4-1 [176] for NR access.

E.2 Bi-directional speech (AMR12.2, IPv4, RTCP and MBR=GBR bearer)

The bitrate for AMR 12.2 including IP overhead (one AMR frame per RTP packet, using bandwidth efficient mode) is 28.8 kbps which is rounded up to 29 kbps. IPv4 is also assumed.

Table E.1: QoS mapping for bi-directional speech (AMR 12.2, IPv4, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 31 | The total bit-rate of AMR12.2 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for uplink (kbps) | 31 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 31 | The total bit-rate of AMR12.2 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for downlink (kbps) | 31 | The same as the guaranteed bitrate |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

E.3 Void

E.4 Bi-directional real-time text (3 kbps, IPv4 or IPv6, RTCP and MBR=GBR bearer)

Bi-directional text at 3 kbps all inclusive (text, IP overhead, RTCP).

Table E.3: QoS mapping for bi-directional real-time text (3 kbps, IPv4, RTCP and MBR=GBR bearer) when using a conversational class bearer

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $1 \cdot 10^{-3}$ | Text should have a higher level of protection than voice and video. |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 3.0 | An assumed total bit-rate of a real-time text service including headers and RTCP. |
| Maximum bitrate for uplink (kbps) | 3.0 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 3.0 | An assumed total bit-rate of a real-time text service including headers and RTCP. |
| Maximum bitrate for downlink (kbps) | 3.0 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | Subscribed value | Indicates the relative importance to other radio access bearers. It should be a lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

Using a conversational class bearer means that resources are reserved throughout the session. Depending on the intended usage of real-time text, it might not be the most resource efficient choice to use a conversational class bearer, especially if it is foreseen that the sessions will be long-lived while the actual text conversations will be rare and bursty. Table E.4 therefore shows an example with QoS mapping for using an interactive class bearer. It is recommended to use QCI 6, 8, or 9 [90] for T.140 real-time text unless the service policy decides to assign different QCI types.

Table E.4: QoS mapping for bi-directional real-time text (3 kbps, IPv4, RTCP) when using an interactive bearer

| Traffic class | Interactive class | Notes |
|-------------------------------|----------------------------|---|
| Delivery order | No | In sequence delivery is not required |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and voice transport delay and delay variation. |
| SDU error ratio | 10^{-4} | Text should have a higher level of protection than voice and video. |
| Maximum bitrate (kbps) | [Depending on UE category] | Should be set as high as the UE category can handle |
| Allocation/Retention priority | Subscribed value | Indicates the relative importance to otherradio access bearers. It should be a lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Traffic handling priority | 3 | |
| Signalling indication | 'No' | |

Table E.5: QoS mapping for bi-directional real-time text (3 kbps, IPv6, RTCP and MBR=GBR bearer) when using a conversational class bearer

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $1 \cdot 10^{-3}$ | Text should have a higher level of protection than voice and video. |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 4.0 | An assumed total bit-rate of a real-time text service including headers and RTCP. |
| Maximum bitrate for uplink (kbps) | 4.0 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 4.0 | An assumed total bit-rate of a real-time text service including headers and RTCP. |
| Maximum bitrate for downlink (kbps) | 4.0 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | Subscribed value | Indicates the relative importance to other radio access bearers. It should be a lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.5 Bi-directional speech (AMR-WB23.85, IPv4, RTCP and MBR=GBR bearer)

The bitrate for AMR-WB 23.85 including IP overhead (one AMR-WB frame per RTP packet, using bandwidth efficient mode) is 40.4 kbps which is rounded up to 41 kbps.

Table E.6: QoS mapping for bi-directional speech (AMR-WB 23.85, IPv4, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 44 | The total bit-rate of AMR-WB23.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for uplink (kbps) | 44 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 44 | The total bit-rate of AMR-WB23.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for downlink (kbps) | 44 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

E.6 Bi-directional video (H.264 AVC level 1.1, 192 kbps, IPv4, RTCP and MBR=GBR bearer)

The video bandwidth is assumed to be 192 kbps and the IPv4 overhead 10 kbps (assuming 15fps and 2 IP packets per frame), resulting in 202 kbps. The transfer delay for video is different from other media. The applicable H.264 profile level can be derived from the "profile-level-id" MIME parameter signalled within the SDP "a=fmtp" attribute. H.264 receivers can request to receive only a lower bandwidth than depicted in this example via the SDP "b:AS" parameter.

Table E.7: QoS mapping for bi-directional video (H.264 AVC level 1.1, 192 kbps, IPv4, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 208 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 10 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 208 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 208 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 10 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps) rounded up to nearest 8 kbps value. If uplink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 208 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.7 Bi-directional speech (AMR12.2, IPv6, RTCP and MBR=GBR bearer)

The bitrate for AMR 12.2 including IP overhead (one AMR frame per RTP packet, using bandwidth efficient mode) is 36.8 kbps which is rounded up to 37 kbps. IPv6 is also assumed.

Table E.8: QoS mapping for bi-directional speech (AMR 12.2, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 39 | The total bit-rate of AMR12.2 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for uplink (kbps) | 39 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 39 | The total bit-rate of AMR12.2 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for downlink (kbps) | 39 | The same as the guaranteed bitrate |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

E.8 Bi-directional speech (AMR-WB23.85, IPv6, RTCP and MBR=GBR bearer)

The bitrate for AMR-WB 23.85 including IP overhead (one AMR-WB frame per RTP packet, using bandwidth efficient mode) is 48.4 kbps which is rounded up to 49 kbps. IPv6 is also assumed.

Table E.9: QoS mapping for bi-directional speech (AMR-WB 23.85, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 52 | The total bit-rate of AMR-WB23.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for uplink (kbps) | 52 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 52 | The total bit-rate of AMR-WB23.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for downlink (kbps) | 52 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

E.9 Void

E.10 Bi-directional video (H.264 AVC level 1.1, 192 kbps, IPv6, RTCP and MBR=GBR bearer)

The video bandwidth is assumed to be 192 kbps and the IPv6 overhead 16 kbps (assuming 15fps and 2 IP packets per frame), resulting in 208 kbps. The transfer delay for video is different from other media. IPv6 is also assumed. The applicable H.264 profile level can be derived from the "profile-level-id" MIME parameter signalled within the SDP "a=fmtp" attribute. H.264 receivers can request to receive only a lower bandwidth than depicted in this example via the SDP "b:AS" parameter.

Table E.11: QoS mapping for bi-directional video (H.264 AVC level 1.1, 192 kbps, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 216 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 16 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 216 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 216 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 16 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps) rounded up to nearest 8 kbps value. If uplink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 216 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.11 Bi-directional speech (AMR, IPv4, RTCP and MBR>GBR bearer)

This QoS profile is defined for AMR (one AMR frame per RTP packet, bandwidth efficient mode) when AMR12.2 and AMR5.9 are used to define MBR and GBR for MBR>GBR bearers. IPv4 is also assumed.

The bitrate for AMR 12.2 including IP overhead is 28.8 kbps and the bitrate for AMR 5.9 including IP overhead is 22.4 kbps.

Table E.12: QoS mapping for bi-directional speech (AMR, IPv4, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 25 | The total bit-rate of AMR5.9 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for uplink (kbps) | 31 | The total bit-rate of AMR12.2 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Guaranteed bitrate for downlink (kbps) | 25 | The total bit-rate of AMR5.9 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for downlink (kbps) | 31 | The total bit-rate of AMR12.2 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

E.12 Bi-directional speech (AMR-WB, IPv4, RTCP and MBR>GBR bearer)

This QoS profile is defined for AMR-WB (one AMR-WB frame per RTP packet, bandwidth efficient mode) when AMR-WB23.85 and AMR-WB8.85 are used to define MBR and GBR for MBR>GBR bearers. IPv4 is also assumed.

The bitrate for AMR-WB23.85 including IP overhead is 40.4 kbps and the bitrate for AMR-WB8.85 including IP overhead is 25.6 kbps.

Table E.13: QoS mapping for bi-directional speech (AMR-WB, IPv4, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 28 | The total bit-rate of AMR-WB8.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for uplink (kbps) | 44 | The total bit-rate of AMRWB23.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Guaranteed bitrate for downlink (kbps) | 28 | The total bit-rate of AMR-WB8.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for downlink (kbps) | 44 | The total bit-rate of AMRWB23.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

E.13 Void

E.14 Bi-directional video (H.264 AVC level 1.1, IPv4, RTCP and MBR>GBR bearer)

The video bandwidths used for defining MBR and GBR are assumed to be 192 kbps and 64 kbps, respectively. The IPv4 overhead is 10 kbps (assuming 15fps and 2 IP packets per frame) for MBR and 5 kbps (assuming 15 fps and 1 IP packet per frame) for GBR, resulting in 202 kbps and 69 kbps, respectively. The transfer delay for video is different from other media. The applicable H.264 profile level can be derived from the "profile-level-id" MIME parameter signalled within the SDP "a=fmtp" attribute. H.264 receivers can request to receive only a lower bandwidth than depicted in this example via the SDP "b:AS" parameter.

Table E.15: QoS mapping for bi-directional video (H.264 AVC level 1.1, IPv4, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 72 | The total bit-rate of a video codec (running at 64 kbps) adding IP/UDP/RTP overhead (assumed to be 5 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. It is up to MTSI implementations or network policies to use higher GBR values. |
| Maximum bitrate for uplink (kbps) | 208 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 10 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. If downlink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Guaranteed bitrate for downlink (kbps) | 72 | The total bit-rate of a video codec (running at 64 kbps) adding IP/UDP/RTP overhead (assumed to be 5 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. It is up to MTSI implementations or network policies to use higher GBR values. |
| Maximum bitrate for downlink (kbps) | 208 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 10 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. If uplink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.15 Bi-directional speech (AMR, IPv6, RTCP and MBR>GBR bearer)

This QoS profile is defined for AMR (one AMR frame per RTP packet, bandwidth efficient mode) when AMR12.2 and AMR5.9 are used to define MBR and GBR for MBR>GBR bearers. IPv6 is also assumed.

The bitrate for AMR 12.2 including IP overhead is 36.8 kbps and the bitrate for AMR 5.9 including IP overhead is 30.4 kbps.

Table E.16: QoS mapping for bi-directional speech (AMR, IPv6, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 33 | The total bit-rate of AMR5.9 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for uplink (kbps) | 39 | The total bit-rate of AMR12.2 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Guaranteed bitrate for downlink (kbps) | 33 | The total bit-rate of AMR5.9 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for downlink (kbps) | 39 | The total bit-rate of AMR12.2 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

E.16 Bi-directional speech (AMR-WB, IPv6, RTCP and MBR>GBR bearer)

This QoS profile is defined for AMR-WB (one AMR-WB frame per RTP packet, bandwidth efficient mode) when AMR-WB23.85 and AMR-WB8.85 are used to define MBR and GBR for MBR>GBR bearers. IPv6 is also assumed.

The bitrate for AMR-WB 23.85 including IP overhead is 48.4 kbps and the bitrate for AMR-WB 8.85 including IP overhead is 33.6 kbps.

Table E.17: QoS mapping for bi-directional speech (AMR-WB, IPv6, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 36 | The total bit-rate of AMR-WB8.85 including IP/UDP/RTP overhead + 5 % for RTCP. |
| Maximum bitrate for uplink (kbps) | 52 | The total bit-rate of AMR-WB23.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Guaranteed bitrate for downlink (kbps) | 36 | The total bit-rate of AMR-WB8.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Maximum bitrate for downlink (kbps) | 52 | The total bit-rate of AMR-WB23.85 including IP/UDP/RTP overhead and 5 % for RTCP. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

E.17 Void

E.18 Bi-directional video (H.264 AVC level 1.1, IPv6, RTCP and MBR>GBR bearer)

The video bandwidths used for defining MBR and GBR are assumed to be 192 kbps and 64 kbps, respectively. The IPv6 overhead is 16 kbps (assuming 15fps and 2 IP packets per frame) for MBR and 8 kbps (assuming 15fps and 1 IP packets per frame) for GBR, resulting in 208 kbps and 72 kbps, respectively. The transfer delay for video is different from other media. The applicable H.264 profile level can be derived from the "profile-level-id" MIME parameter signalled within the SDP "a=fmtp" attribute. H.264 receivers can request to receive only a lower bandwidth than depicted in this example via the SDP "b:AS" parameter.

Table E.19: QoS mapping for bi-directional video (H.264 AVC level 1.1, IPv6, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 80 | The total bit-rate of a video codec (running at 64 kbps) adding IP/UDP/RTP overhead (assumed to be 8 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. It is up to MTSI implementations or network policies to use higher GBR values. |
| Maximum bitrate for uplink (kbps) | 216 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 16 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. If downlink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Guaranteed bitrate for downlink (kbps) | 80 | The total bit-rate of a video codec (running at 64 kbps) adding IP/UDP/RTP overhead (assumed to be 8 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. It is up to MTSI implementations or network policies to use higher GBR values. |
| Maximum bitrate for downlink (kbps) | 216 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 16 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. If uplink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.19 Bi-directional video (H.264 AVC level 1.2, 384 kbps, IPv4, RTCP and MBR=GBR bearer)

The video bandwidth is assumed to be 384 kbps and the IPv4 overhead 20 kbps (assuming 15fps and 4 IP packets per frame), resulting in 404 kbps. The transfer delay for video is different from other media. The applicable H.264 profile level can be derived from the "profile-level-id" MIME parameter signalled within the SDP "a=fmtp" attribute.

Table E.20: QoS mapping for bi-directional video (H.264 AVC level 1.2, 384 kbps, IPv4, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 408 | The total bit-rate of a video codec (running at 384 kbps) adding IP/UDP/RTP overhead (assumed to be 20 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 408 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 408 | The total bit-rate of a video codec (running at 384 kbps) adding IP/UDP/RTP overhead (assumed to be 20 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps) rounded up to nearest 8 kbps value. If uplink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 408 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.20 Bi-directional video (H.264 AVC level 1.2, 384 kbps, IPv6, RTCP and MBR=GBR bearer)

The video bandwidth is assumed to be 384 kbps and the IPv6 overhead 32 kbps (assuming 15fps and 4 IP packets per frame), resulting in 416 kbps. The transfer delay for video is different from other media. IPv6 is also assumed. The applicable H.264 profile level can be derived from the "profile-level-id" MIME parameter signalled within the SDP

"a=fmtp" attribute. H.264 receivers can request to receive only a lower bandwidth than depicted in this example via the SDP "b:AS" parameter.

Table E.21: QoS mapping for bi-directional video (H.264 AVC level 1.2, 384 kbps, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 424 | The total bit-rate of a video codec (running at 384 kbps) adding IP/UDP/RTP overhead (assumed to be 32 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 424 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 424 | The total bit-rate of a video codec (running at 384 kbps) adding IP/UDP/RTP overhead (assumed to be 32 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps) rounded up to nearest 8 kbps value. If uplink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 424 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.21 Bi-directional video (H.264 AVC level 1.2, IPv4, RTCP and MBR>GBR bearer)

The video bandwidths used for defining MBR and GBR are assumed to be 384 kbps and 192 kbps, respectively. The IPv4 overhead is 20 kbps (assuming 15fps and 4 IP packets per frame) for MBR and 10 kbps (assuming 15 fps and 2 IP packets per frame) for GBR, resulting in 404 kbps and 202 kbps, respectively. The transfer delay for video is different from other media. The applicable H.264 profile level can be derived from the "profile-level-id" MIME parameter signalled within the SDP "a=fmtp" attribute. H.264 receivers can request to receive only a lower bandwidth than depicted in this example via the SDP "b:AS" parameter.

Table E.22: QoS mapping for bi-directional video (H.264 AVC level 1.2, IPv4, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 208 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 10 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. It is up to MTSI implementations or network policies to use higher GBR values. |
| Maximum bitrate for uplink (kbps) | 408 | The total bit-rate of a video codec (running at 384 kbps) adding IP/UDP/RTP overhead (assumed to be 20 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. If downlink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Guaranteed bitrate for downlink (kbps) | 208 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 10 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. It is up to MTSI implementations or network policies to use higher GBR values. |
| Maximum bitrate for downlink (kbps) | 408 | The total bit-rate of a video codec (running at 384 kbps) adding IP/UDP/RTP overhead (assumed to be 20 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. If uplink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.22 Bi-directional video (H.264 AVC level 1.2, IPv6, RTCP and MBR>GBR bearer)

The video bandwidths used for defining MBR and GBR are assumed to be 384 kbps and 192 kbps, respectively. The IPv6 overhead is 32 kbps (assuming 15fps and 4 IP packets per frame) for MBR and 16 kbps (assuming 15fps and 2 IP packets per frame) for GBR, resulting in 416 kbps and 208 kbps, respectively. The transfer delay for video is different from other media. The applicable H.264 profile level can be derived from the "profile-level-id" MIME parameter signalled within the SDP "a=fmtp" attribute. H.264 receivers can request to receive only a lower bandwidth than depicted in this example via the SDP "b:AS" parameter.

Table E.23: QoS mapping for bi-directional video (H.264 AVC level 1.2, IPv6, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 216 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 16 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. It is up to MTSI implementations or network policies to use higher GBR values. |
| Maximum bitrate for uplink (kbps) | 424 | The total bit-rate of a video codec (running at 384 kbps) adding IP/UDP/RTP overhead (assumed to be 32 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. If downlink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Guaranteed bitrate for downlink (kbps) | 216 | The total bit-rate of a video codec (running at 192 kbps) adding IP/UDP/RTP overhead (assumed to be 16 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. It is up to MTSI implementations or network policies to use higher GBR values. |
| Maximum bitrate for downlink (kbps) | 424 | The total bit-rate of a video codec (running at 384 kbps) adding IP/UDP/RTP overhead (assumed to be 32 kbps) and RTCP (RS:0 and RR:5000 used in clause A.6 adds 2.5kbps). The value is then rounded up to nearest multiple of 8 kbps. If uplink SDP contains a lower b:AS bandwidth modifier value, this should be used instead. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.23 Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 500 kbps, IPv6, RTCP and MBR=GBR bearer)

The video bandwidth is assumed to be 500 kbps and the IPv6 overhead 36 kbps (assuming 25 fps, 3 IP packets per frame and IPv6), resulting in 540 kbps. Adding 5% for RTCP increases the bandwidth by 27 kbps. However, the RTCP

bandwidth is limited to max 14 kbps, see clause 7.3.1. Rounding up to the next higher integer multiple of 8 kbps gives 560 kbps.

Table E.24: QoS mapping for bi-directional video (H.265 (HEVC) level 3.1, 300 kbps, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 560 | The total bit-rate of a video codec (running at 500 kbps) adding IPv6/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 % but limited to max 14 kbps) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 560 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 560 | The total bit-rate of a video codec (running at 500 kbps) adding IPv6/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5% but limited to max 14 kbps) rounded up to nearest 8 kbps value. If uplink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 560 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.24 Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 500/40 kbps, IPv6, RTCP and MBR>GBR bearer)

The video bandwidths used for defining MBR and GBR are assumed to be 500 kbps and 40 kbps, respectively. The IPv6 overhead is 36 kbps (assuming 25 fps and 3 IP packets per frame) for MBR and 2.4 kbps (assuming QCIF, 5 fps and 1 IP packets per frame) for GBR, resulting in 540 kbps and 45 kbps, respectively. Adding 5% for RTCP increases the bandwidth by 27 kbps for both MBR and GBR. However, the RTCP bandwidth is limited to max 14 kbps, see clause 7.3.1. Rounding up to the nearest higher integer multiple of 8 kbps gives 560 kbps and 64 kbps, respectively.

Table E.25: QoS mapping for bi-directional video (H.265 (HEVC) level 3.1, 500/40 kbps, IPv6, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 64 | The total bit-rate of a video codec (running at 50 kbps) adding IP/UDP/RTP overhead (assumed to be 2.4 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 560 | The total bit-rate of a video codec (running at 500 kbps) adding IP/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Guaranteed bitrate for downlink (kbps) | 64 | The total bit-rate of a video codec (running at 40 kbps) adding IP/UDP/RTP overhead (assumed to be 2.4 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 560 | The total bit-rate of a video codec (running at 500 kbps) adding IP/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.25 Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 600 kbps, IPv6, RTCP and MBR=GBR bearer)

The video bandwidth is assumed to be 600 kbps and the IPv6 overhead 36 kbps (assuming 25 fps, 3 IP packets per frame and IPv6), resulting in 640 kbps. Adding 5% for RTCP increases the bandwidth by 32 kbps. However, the RTCP bandwidth is limited to max 14 kbps, see clause 7.3.1. Rounding up to the next higher integer multiple of 8 kbps gives 656 kbps.

Table E.26: QoS mapping for bi-directional video (H.265 (HEVC) level 3.1, 600 kbps, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 656 | The total bit-rate of a video codec (running at 600 kbps) adding IPv6/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 %) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 656 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 656 | The total bit-rate of a video codec (running at 600 kbps) adding IPv6/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5%) rounded up to nearest 8 kbps value. If uplink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 656 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.26 Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 600/40 kbps, IPv6, RTCP and MBR>GBR bearer)

The video bandwidths used for defining MBR and GBR are assumed to be 600 kbps and 40 kbps, respectively. The IPv6 overhead is 36 kbps (assuming 25 fps and 3 IP packets per frame) for MBR and 2.4 kbps (assuming QCIF, 5 fps and 1 IP packets per frame) for GBR, resulting in 640 kbps and 45 kbps, respectively. Adding 5% for RTCP increases the bandwidth by 32 kbps for both MBR and GBR. However, the RTCP bandwidth is limited to max 14 kbps, see clause 7.3.1. Rounding up to the nearest higher integer multiple of 8 kbps gives 656 kbps and 64 kbps, respectively.

Table E.27: QoS mapping for bi-directional video (H.265 (HEVC) level 3.1, 600/40 kbps, IPv6, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 64 | The total bit-rate of a video codec (running at 40 kbps) adding IP/UDP/RTP overhead (assumed to be 2.4 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 656 | The total bit-rate of a video codec (running at 600 kbps) adding IP/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Guaranteed bitrate for downlink (kbps) | 64 | The total bit-rate of a video codec (running at 40 kbps) adding IP/UDP/RTP overhead (assumed to be 2.4 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 656 | The total bit-rate of a video codec (running at 600 kbps) adding IP/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.27 Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 650 kbps, IPv6, RTCP and MBR=GBR bearer)

The video bandwidth is assumed to be 650 kbps and the IPv6 overhead 36 kbps (assuming 25 fps, 3 IP packets per frame and IPv6), resulting in 690 kbps. Adding 5% for RTCP increases the bandwidth by 34.5 kbps. However, the RTCP bandwidth is limited to max 14 kbps, see clause 7.3.1. Rounding up to the next higher integer multiple of 8 kbps gives 704 kbps.

Table E.28: QoS mapping for bi-directional video (H.265 (HEVC) level 3.1, 650 kbps, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 704 | The total bit-rate of a video codec (running at 650 kbps) adding IPv6/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 %) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 704 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 704 | The total bit-rate of a video codec (running at 650 kbps) adding IPv6/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5%) rounded up to nearest 8 kbps value. If uplink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 704 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.28 Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 650/40 kbps, IPv6, RTCP and MBR>GBR bearer)

The video bandwidths used for defining MBR and GBR are assumed to be 650 kbps and 40 kbps, respectively. The IPv6 overhead is 36 kbps (assuming 25 fps and 3 IP packets per frame) for MBR and 2.4 kbps (assuming QCIF, 5 fps and 1 IP packets per frame) for GBR, resulting in 690 kbps and 45 kbps, respectively. Adding 5% for RTCP increases the bandwidth by 34.5 kbps for both MBR and GBR. However, the RTCP bandwidth is limited to max 14 kbps, see clause 7.3.1. Rounding up to the nearest higher integer multiple of 8 kbps gives 704 kbps and 72 kbps, respectively.

Table E.29: QoS mapping for bi-directional video (H.265 (HEVC) level 3.1, 650/40 kbps, IPv6, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 64 | The total bit-rate of a video codec (running at 40 kbps) adding IP/UDP/RTP overhead (assumed to be 2.4 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 704 | The total bit-rate of a video codec (running at 650 kbps) adding IP/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Guaranteed bitrate for downlink (kbps) | 64 | The total bit-rate of a video codec (running at 40 kbps) adding IP/UDP/RTP overhead (assumed to be 2.4 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 704 | The total bit-rate of a video codec (running at 650 kbps) adding IP/UDP/RTP overhead (assumed to be 36 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.29 Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 750 kbps, IPv6, RTCP and MBR=GBR bearer)

The video bandwidth is assumed to be 750 kbps and the IPv6 overhead 48 kbps (assuming 25 fps, 4 IP packets per frame and IPv6), resulting in 800 kbps. Adding 5% for RTCP increases the bandwidth by 40 kbps. However, the RTCP bandwidth is limited to max 14 kbps, see clause 7.3.1. Rounding up to the next higher integer multiple of 8 kbps gives 816 kbps.

Table E.30: QoS mapping for bi-directional video (H.265 (HEVC) level 3.1, 750 kbps, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 816 | The total bit-rate of a video codec (running at 750 kbps) adding IPv6/UDP/RTP overhead (assumed to be 48 kbps) and RTCP (adds 5 %) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 816 | The same as the guaranteed bitrate. |
| Guaranteed bitrate for downlink (kbps) | 816 | The total bit-rate of a video codec (running at 750 kbps) adding IPv6/UDP/RTP overhead (assumed to be 48 kbps) and RTCP (adds 5%) rounded up to nearest 8 kbps value. If uplink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 816 | The same as the guaranteed bitrate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.30 Bi-directional video (H.265 (HEVC) Main profile, Main tier, level 3.1, 750/40 kbps, IPv6, RTCP and MBR>GBR bearer)

The video bandwidths used for defining MBR and GBR are assumed to be 750 kbps and 40 kbps, respectively. The IPv6 overhead is 48 kbps (assuming 25 fps and 4 IP packets per frame) for MBR and 2.4 kbps (assuming QCIF, 5 fps and 1 IP packets per frame) for GBR, resulting in 800 kbps and 45 kbps, respectively. Adding 5% for RTCP increases the bandwidth by 40 kbps for both MBR and GBR. However, the RTCP bandwidth is limited to max 14 kbps, see clause 7.3.1. Rounding up to the nearest higher integer multiple of 8 kbps gives 816 kbps and 72 kbps, respectively.

Table E.31: QoS mapping for bi-directional video (H.265 (HEVC) level 3.1, 750/40 kbps, IPv6, RTCP and MBR>GBR bearer)

| Traffic class | Conversational class | Notes |
|--|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets |
| Delivery of erroneous SDUs | No | |
| Residual BER | 10^{-5} | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $7 \cdot 10^{-3}$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for video services |
| Transfer delay (ms) | 170 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bitrate for uplink (kbps) | 64 | The total bit-rate of a video codec (running at 40 kbps) adding IP/UDP/RTP overhead (assumed to be 2.4 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for uplink (kbps) | 816 | The total bit-rate of a video codec (running at 750 kbps) adding IP/UDP/RTP overhead (assumed to be 48 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Guaranteed bitrate for downlink (kbps) | 64 | The total bit-rate of a video codec (running at 40 kbps) adding IP/UDP/RTP overhead (assumed to be 2.4 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Maximum bitrate for downlink (kbps) | 816 | The total bit-rate of a video codec (running at 750 kbps) adding IP/UDP/RTP overhead (assumed to be 48 kbps) and RTCP (adds 5 % of the session bandwidth) rounded up to nearest 8 kbps value. If downlink SDP contains a lower b=AS bandwidth modifier value, this should be used instead. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the same or next lower value to the priority of a Conversational bearer with source statistics descriptor 'speech'. |
| Source statistics descriptor | 'unknown' | |

E.31 Bi-directional speech (EVS 13.2, IPv4, RTCP and MBR=GBR bearer)

The bit-rate for EVS 13.2 including IP overhead (one EVS frame per RTP packet, using Header-Full format) is 30 kbps. IPv4 is assumed.

Table E.32: QoS mapping for bi-directional speech (EVS 13.2, IPv4, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|---|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets. |
| Delivery of erroneous SDUs | No | |
| Residual BER | $[10^{-5}]$ | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $[7 \cdot 10^{-3}]$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services. |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bit-rate for uplink (kbps) | 32 | The total bit-rate of EVS 13.2 including IP/UDP/RTP overhead and 2000 bps for RTCP. |
| Maximum bit-rate for uplink (kbps) | 32 | The same as the guaranteed bit-rate. |
| Guaranteed bit-rate for downlink (kbps) | 32 | The total bit-rate of EVS 13.2 including IP/UDP/RTP overhead and 2000 bps for RTCP. |
| Maximum bit-rate for downlink (kbps) | 32 | The same as the guaranteed bit-rate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

NOTE: Residual BER and SDU error ratio of EVS are for further study.

E.32 Bi-directional speech (EVS 24.4, IPv4, RTCP and MBR=GBR bearer)

The bit-rate for EVS 24.4 including IP overhead (one EVS frame per RTP packet, using Header-Full format) is 41.2 kbps which is rounded up to 42 kbps. IPv4 is assumed.

Table E.33: QoS mapping for bi-directional speech (EVS 24.4, IPv4, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|---|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets. |
| Delivery of erroneous SDUs | No | |
| Residual BER | $[10^{-5}]$ | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $[7 \cdot 10^{-3}]$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services. |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bit-rate for uplink (kbps) | 44 | The total bit-rate of EVS 24.4 including IP/UDP/RTP overhead and 2000 bps for RTCP. |
| Maximum bit-rate for uplink (kbps) | 44 | The same as the guaranteed bit-rate. |
| Guaranteed bit-rate for downlink (kbps) | 44 | The total bit-rate of EVS 24.4 including IP/UDP/RTP overhead and 2000 bps for RTCP. |
| Maximum bit-rate for downlink (kbps) | 44 | The same as the guaranteed bit-rate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

NOTE: Residual BER and SDU error ratio of EVS are for further study.

E.33 Bi-directional speech (EVS 13.2, IPv6, RTCP and MBR=GBR bearer)

The bit-rate for EVS 13.2 including IP overhead (one EVS frame per RTP packet, using Header-Full format) is 38 kbps. IPv6 is assumed.

Table E.34: QoS mapping for bi-directional speech (EVS 13.2, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|---|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets. |
| Delivery of erroneous SDUs | No | |
| Residual BER | $[10^{-5}]$ | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $[7 \cdot 10^{-3}]$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services. |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bit-rate for uplink (kbps) | 40 | The total bit-rate of EVS13.2 including IP/UDP/RTP overhead and 2000 bps for RTCP. |
| Maximum bit-rate for uplink (kbps) | 40 | The same as the guaranteed bit-rate. |
| Guaranteed bit-rate for downlink (kbps) | 40 | The total bit-rate of EVS13.2 including IP/UDP/RTP overhead and 2000 bps for RTCP. |
| Maximum bit-rate for downlink (kbps) | 40 | The same as the guaranteed bit-rate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

NOTE: Residual BER and SDU error ratio of EVS are for further study.

E.34 Bi-directional speech (EVS 24.4, IPv6, RTCP and MBR=GBR bearer)

The bit-rate for EVS 24.4 including IP overhead (one EVS frame per RTP packet, using Header-Full format) is 49.2 kbps which is rounded up to 50 kbps. IPv6 is assumed.

Table E.35: QoS mapping for bi-directional speech (EVS 24.4, IPv6, RTCP and MBR=GBR bearer)

| Traffic class | Conversational class | Notes |
|---|----------------------|--|
| Delivery order | No | The application should handle packet reordering. |
| Maximum SDU size (octets) | 1400 | Maximum size of IP packets. |
| Delivery of erroneous SDUs | No | |
| Residual BER | $[10^{-5}]$ | Reflects the desire to have a medium level of protection to achieve an acceptable compromise between packet loss rate and speech transport delay and delay variation. |
| SDU error ratio | $[7 \cdot 10^{-3}]$ | A packet loss rate of 0.7 % per wireless link is in general sufficient for speech services. |
| Transfer delay (ms) | 130 ms | Indicates maximum delay for 95 th percentile of the distribution of delay for all delivered SDUs between the UE and the PS domain during the lifetime of a bearer service. Permits the derivation of the RAN part of the total transfer delay for the radio access bearer. This attribute allows RAN to set transport formats and H-ARQ/ARQ parameters such as the discard timer. |
| Guaranteed bit-rate for uplink (kbps) | 52 | The total bit-rate of EVS 24.4 including IP/UDP/RTP overhead and 2000 bps for RTCP. |
| Maximum bit-rate for uplink (kbps) | 52 | The same as the guaranteed bit-rate. |
| Guaranteed bit-rate for downlink (kbps) | 52 | The total bit-rate of EVS 24.4 including IP/UDP/RTP overhead and 2000 bps for RTCP. |
| Maximum bit-rate for downlink (kbps) | 52 | The same as the guaranteed bit-rate. |
| Allocation/Retention priority | subscribed value | Indicates the relative importance to other radio access bearers. It should be the next lower value to the priority of the signalling bearer. |
| Source statistics descriptor | 'speech' | |

NOTE: Residual BER and SDU error ratio of EVS are for further study.

Annex F (Normative): Void

Annex G (Normative): DTMF events

G.1 General

Typically, "DTMF events" are triggered by an end-user (on either side of the speech call) by pressing some key of the user's terminal, with the goal to control services in the network or in an application server. This may happen throughout the duration of the speech call on both ends of the call.

It might happen that DTMF events are forwarded as telephone events to the (remote) MTSI client in terminal. In such a case, the MTSI client in terminal may convert the telephone events into audible DTMF tone-pairs. There is, however, no requirement in 3GPP for handling downlink telephone events.

This annex describes a method for sending DTMF events as telephone events within the same RTP media stream as the speech, i.e. "inband", using the same IP address and UDP port as the RTP for speech, but using a different RTP Payload Type number.

- MTSI clients in terminal offering speech communication shall support the below described method in the transmitting direction (uplink) and should support it in the receiving direction (downlink).
- MTSI media gateways offering speech communication shall support the below described method in both directions with respect to the MTSI client in terminal, the transmitting direction (downlink) and in the receiving direction (uplink). For MTSI media gateways, the described method applies only to the PS session between the MTSI gateway and the MTSI client in terminal.
- The use of the telephone-event codec between MTSI media gateways is recommended. Typically, DTMF events are also transported between MTSI media gateways as telephone events in both directions.

G.2 Encoding of DTMF events

DTMF events shall be encoded and transmitted in MTSI sessions as "telephone events" in RTP packets, using the selected "telephone-event" codec. DTMF events in this Annex refer to the DTMF Named Events described in Section 3.2, Table 3 in IETF RFC 4733 [61], i.e. events (0-9, *, # and A-D) which are encoded with event code values 0-9, 10, 11 and 12-15 respectively.

DTMF events are carried as part of the audio stream which can either be narrowband, wideband, super-wideband or fullband audio, i.e. use 8 kHz or 16 kHz RTP clock rate respectively. MTSI clients that in addition to narrowband also support wideband, super-wideband or fullband speech shall support telephone-event RTP clock rates matching all supported codecs. When switching between speech and DTMF, telephone events shall use the same RTP clock rate as the currently used codec for the speech signal in the same RTP stream.

The encoding of DTMF events includes specifying the duration time for the events, see IETF RFC 4733 [61]. Supporting long-lasting DTMF events, where the duration time exceeds the maximum duration time expressible by the duration field, is optional. If supported, long-lasting DTMF events shall be divided into segments, see IETF RFC 4733 [61]. To harmonize with legacy DTMF signalling, [62], [63], the tone duration of a DTMF event shall be at least 65 ms and the pause duration in-between two DTMF events shall be at least 65 ms. The duration of the DTMF event and the pause time to the next DTMF event, where applicable, should be selected such that it enables incrementing the RTP Time Stamp with an integer multiple of the number of timestamp units corresponding to the frame length of the speech codec used for the speech media before the DTMF event.

G.3 Session setup

The MTSI client(s) in terminal and the IMS network(s) shall support the telephone-event codec(s) for the transport of DTMF events during the whole MTSI session.

NOTE: They may in addition support the SIP INFO method (TS 24.229 [7]) for setup and modification of supplementary services, when no user plane is necessary. The SIP INFO method is, however, not to be used during MTSI sessions.

When sending an SDP offer, an MTSI client should indicate support of all DTMF Named Events 0 to 15 in the fmp parameter. As defined by IETF RFC 4733 [61] section 7.1.1, if this fmp parameter is not included, then the default applies: 0-15.

If the SDP offer includes a single audio codec then, this SDP offer shall also include the telephone-event codec with the same RTP clock rate as used for the offered audio codec, as described by Errata 3489 to IETF RFC 4733 [61]. If the SDP offer includes multiple audio codecs with different RTP clock rates, then this SDP offer shall also include the telephone-event codecs with different payload type numbers for these different RTP clock rates.

The answerer, which determines the Selected (audio) Codec(s), shall, if telephone-event was included in the SDP offer, include in the SDP answer the corresponding telephone-event codec that matches the highest RTP clock rate from the Selected (audio) Codec(s).

If the SDP offer or answer contains multiple m=audio lines, then the telephone-event shall only be included for the first m=audio line.

An example of SDP offer and answer from MTSI clients in terminals using 3GPP access is provided in Table G.3.1 when narrowband speech codec with RTP clock rate 8000 is offered.

Table G.3.1: SDP example for narrowband speech and DTMF

| SDP offer |
|---|
| <pre>m=audio 49152 RTP/AVPF 97 98 99 a=rtpmap:97 AMR/8000/1 a=fmp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 telephone-event/8000 a=fmp:99 0-15 aptime:20 amaxptime:240 a=sendrecv</pre> |
| SDP answer example |
| <pre>m=audio 49152 RTP/AVPF 97 99 a=rtpmap:97 AMR/8000/1 a=fmp:97 mode-change-capability=2; max-red=220 a=rtpmap:99 telephone-event/8000 a=fmp:99 0-15 aptime:20 amaxptime:240 a=sendrecv</pre> |

An example of SDP offer and answer from MTSI clients in terminals using 3GPP access is provided in Table G.3.2 when narrowband speech codecs with RTP clock rate 8000 and wideband and super-wideband speech codecs with RTP clock rate 16000 are offered.

Table G.3.2: SDP example for narrowband,wideband and super-wideband for both speech and DTMF

| SDP offer |
|--|
| <pre> m=audio 49152 RTP/AVPF 96 97 98 99 100 101 102 a=rtpmap:96 EVS/16000/1 a=fmtp:96 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:99 telephone-event/16000 a=fmtp:99 0-15 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:102 telephone-event/8000 a=fmtp:102 0-15 aptime:20 a=maxptime:240 a=sendrecv </pre> |
| SDP answer example |
| <pre> m=audio 49152 RTP/AVPF 96 99 a=rtpmap:96 EVS/16000/1 a=fmtp:96 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:99 telephone-event/16000 a=fmtp:99 0-15 aptime:20 a=maxptime:240 a=sendrecv </pre> |

NOTE 1: Wideband codec ITU-T G.722 uses an RTP clock rate of 8000, although its sampling frequency is 16000 Hz. An SDP offer or answer with G.722/8000 therefore must be combined with telephone-event/8000. The 3GPP EVS codec supports various sampling frequencies, up to 48000 Hz, but only one RTP clock rate of 16000 and the corresponding telephone-event/16000 is used.

NOTE 2: The a=sendrecv attribute applies to all RTP payload types within the same media stream. To comply with the transmission rules defined in clause G.4, SDP offers and SDP answers include audio codecs and telephone-event codec(s) for the same media stream in both directions. The consequence of this is that MTSI clients that want to send DTMF events in local uplink, also allow the remote MTSI client to send DTMF events in the reverse direction.

For MTSI clients in terminals, since support of DTMF events in the receiving direction is not mandatory, it is an implementation consideration to decide how to handle any received RTP packets containing telephone-event codec payload.

G.4 Data transport for DTMF events

For sending and receiving DTMF events via telephone events with RTP, the RTP payload format for "DTMF Named Events" of the telephone-event codec(s), IETF RFC 4733 [61], shall be supported by MTSI clients in terminal and MTSI media gateways.

Telephone events shall use the same RTP media stream as for speech, i.e. the same IP address, UDP port, RTP SSRC and RTP clock rate as the Selected (Speech) Codec. Thereby, RTP Sequence Number and RTP Time Stamp shall be

synchronized between RTP packets for speech and RTP packets for DTMF events. For example, by setting the initial random values the same and when switching from speech to DTMF, or vice versa, the RTP Sequence Number and RTP Time Stamp shall continue from the value that was used for the other audio media (speech or DTMF event).

The RTP Sequence Number for telephone events shall increment in the same way as for speech, i.e. by 1 for each transmitted RTP packet.

The RTP Time Stamp for telephone events should increment in the same way as for RTP speech packets or with an integer multiple. Example: if the RTP Time Stamp increments with 160 between RTP speech packets (Codec with RTP clock rate 8000 and 20 ms frame duration) then the RTP Time Stamp increment during DTMF events and when switching between speech and DTMF events should be 160 or an integer multiple of 160. The RTP Time Stamp should not increment with a smaller interval for DTMF events than for speech frames. The RTP Time Stamp shall use the same RTP clock rate as for the speech codec that is transmitted in the same RTP stream immediately before the start of the DTMF event(s).

NOTE 1: A DTMF event may be transmitted in several RTP packets even if the DTMF event has a shorter duration time than what is expressible by the duration field. In this case all RTP packets containing the same DTMF event within the same segment shall have the same RTP Time Stamp value according to IETF RFC 4733 [61].

Speech packets shall not be transmitted such that the resulting decoded speech would overlap in time with the audible representation of DTMF events, when DTMF events are transmitted in the same RTP media stream.

NOTE 2: Most DTMF events are not translated into DTMF tone-pairs, because they are used by the network or an application server (e.g. a voice mail box). For such cases, it would not matter, if DTMF events are overlapping with speech packets, but the non-overlap restriction is kept here for backward compatibility reasons. If an MTSI client receives speech and DTMF event packets, overlapping during a possible playout time, then the MTSI client optimally ignores the overlapping speech packets and plays out only the DTMF events correctly as DTMF tone-pairs, embedded into sufficient silence.

NOTE 3: If the RTP stream carrying the telephone events is paused on RTP level [156], the need to send telephone events may cause the MTSI client in terminal to leave the paused state.

Annex H (informative): Network Preference Management Object Device Description Framework

This Device Description Framework (DDF) is the standardized minimal set. A vendor can define its own DDF for the complete device. This DDF can include more features than this minimal standardized version. This MO is included in the zip-archive "3GPP MTSI MOs.zip" attached to the present document.

Annex I (informative): QoE Reporting Management Object Device Description Framework

This Device Description Framework (DDF) is the standardized minimal set. A vendor can define its own DDF for the complete device. This DDF can include more features than this minimal standardized version. This MO is included in the zip-archive "3GPP MTSI MOs.zip" attached to the present document.

Annex J (informative): Media Adaptation Management Object Device Description Framework

This Device Description Framework (DDF) is the standardized minimal set. A vendor can define its own DDF for the complete device. This DDF can include more features than this minimal standardized version. This MO is included in the zip-archive "3GPP MTSI MOs.zip" attached to the present document.

Annex K (informative): Computation of b=AS for AMR and AMR-WB

K.1 General

This annex contains examples of computing b=AS for AMR and AMR-WB when ptime=20 and when ptime=40. In these examples, it is assumed that no extra bandwidth is allocated for redundancy.

K.2 Procedure for computing the bandwidth

The bandwidth is calculated using the following procedure when no extra bandwidth is allocated for redundancy:

- 1) Calculate the size of the RTP payload, see below.
- 2) Calculate the size of the IP packets by taking the RTP payload size (in bytes) and adding the IP/UDP/RTP overhead: 20 bytes for IPv4; 40 bytes for IPv6; 8 bytes for UDP; 12 bytes for RTP.
- 3) Convert the IP packet size to bits.
- 4) Calculate the required bit rate (bps) given the packet size and the packet rate: 50 packets per second for 1 frame per packet; 25 packets per second for 2 frames per packet.
- 5) The b=AS bandwidth is then calculated by converting the required bit rate to kbps and rounding to the nearest higher integer value.

When redundancy is used then the RTP payload contains several frames, both non-redundant and redundant. For example, for 100% redundancy each RTP payload contains one non-redundant frame and one redundant frame, giving 2 frames per packet. The packet rate is however still 50 frames per packet.

If the SDP includes multiple codecs and/or configurations then the bandwidth is calculated for each configuration and the b=AS bandwidth is set to the highest of the bandwidths.

K.3 Computation of RTP payload size

When the b=AS bandwidth is computed it is assumed that the codec is using the highest allowed coded mode for each frame.

The RTP payload size for the bandwidth-efficient payload format mode and 1 frame/packet is calculated from the following components:

- 4 bits for the payload header (=CMR)
- 6 bits for the Table of Contents (ToC)
- N bits for the speech frame (size depends on codec mode)
- Padding bits at the end of the RTP payload to give an integer number of octets

The RTP payload size for the octet-aligned payload format mode and 1 frame/packet is calculated from the following components:

- 4 bits for the payload header (=CMR) + 4 bits padding
- 6 bits for the Table of Contents (ToC) + 2 bits padding
- N bits for the speech frame (size depends on codec mode) + padding bits to give an integer number of octets
- No padding in the end of the RTP payload is needed since each item is already an integer number of octets

The RTP payload size for the bandwidth-efficient payload format mode and 2 frames per packet is calculated from the following components:

- 4 bits for the payload header (=CMR)
- 6 bits for the Table of Contents (ToC) for speech frame 1
- 6 bits for the Table of Contents (ToC) for speech frame 2
- N1 bits for the speech frame 1 (size depends on codec mode)
- N2 bits for the speech frame 2 (size depends on codec mode)
- Padding bits at the end of the RTP payload to give an integer number of octets

The RTP payload size for the octet-aligned payload format mode and 2 frames per packet is calculated from the following components:

- 4 bits for the payload header (=CMR) + 4 bits padding
- 6 bits for the Table of Contents (ToC) for speech frame 1 + 2 bits padding
- 6 bits for the Table of Contents (ToC) for speech frame 2 + 2 bits padding
- N1 bits for the speech frame 1 (size depends on codec mode) + padding bits to give an integer number of octets
- N2 bits for the speech frame 2 (size depends on codec mode) + padding bits to give an integer number of octets
- No padding in the end of the RTP payload is needed since each item is already an integer number of octets

K.4 Detailed computation

The tables below give a detailed description of the bandwidth computation. The $b=AS$ bandwidth is not defined for SID. Since a SID is rarely encapsulated with a speech frame or another SID, Tables K.9-16 do not include the computation procedure for the comfort noise frame.

Table K.1: Computation of $b=AS$ for AMR (IPv4, ptime=20, bandwidth-efficient mode)

| Mode | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 | SID |
|--------------------------------|-------|-------|------|------|-------|-------|-------|-------|------|
| Bits per speech frame | 95 | 103 | 118 | 134 | 148 | 159 | 204 | 244 | 39 |
| Payload header and ToC | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| RTP payload (bits) | 105 | 113 | 128 | 144 | 158 | 169 | 214 | 254 | 49 |
| RTP payload (bytes) | 13.13 | 14.13 | 16 | 18 | 19.75 | 21.13 | 26.75 | 31.75 | 6.13 |
| Rounded-up RTP payload (bytes) | 14 | 15 | 16 | 18 | 20 | 22 | 27 | 32 | 7 |
| Rounded-up RTP payload (bits) | 112 | 120 | 128 | 144 | 160 | 176 | 216 | 256 | 56 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 20 ms | 432 | 440 | 448 | 464 | 480 | 496 | 536 | 576 | 376 |
| Total bit-rate (kbps) | 21.6 | 22 | 22.4 | 23.2 | 24 | 24.8 | 26.8 | 28.8 | 18.8 |
| AS | 22 | 22 | 23 | 24 | 24 | 25 | 27 | 29 | N/A |

Table K.2: Computation of b=AS for AMR (IPv6, ptime=20, bandwidth-efficient mode)

| Mode | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 | SID |
|--------------------------------|-------|-------|------|------|-------|-------|-------|-------|------|
| Bits per speech frame | 95 | 103 | 118 | 134 | 148 | 159 | 204 | 244 | 39 |
| Payload header and ToC | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| RTP payload (bits) | 105 | 113 | 128 | 144 | 158 | 169 | 214 | 254 | 49 |
| RTP payload (bytes) | 13.13 | 14.13 | 16 | 18 | 19.75 | 21.13 | 26.75 | 31.75 | 6.13 |
| Rounded-up RTP payload (bytes) | 14 | 15 | 16 | 18 | 20 | 22 | 27 | 32 | 7 |
| Rounded-up RTP payload (bits) | 112 | 120 | 128 | 144 | 160 | 176 | 216 | 256 | 56 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 20 ms | 592 | 600 | 608 | 624 | 640 | 656 | 696 | 736 | 536 |
| Total bit-rate (kbps) | 29.6 | 30 | 30.4 | 31.2 | 32 | 32.8 | 34.8 | 36.8 | 26.8 |
| AS | 30 | 30 | 31 | 32 | 32 | 33 | 35 | 37 | N/A |

Table K.3: Computation of b=AS for AMR (IPv4, ptime=20, octet-aligned mode)

| Mode | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 | SID |
|--------------------------------------|-------|-------|-------|-------|------|-------|------|------|------|
| Bits per speech frame | 95 | 103 | 118 | 134 | 148 | 159 | 204 | 244 | 39 |
| Speech frame size (bytes) | 11.88 | 12.88 | 14.75 | 16.75 | 18.5 | 19.88 | 25.5 | 30.5 | 4.88 |
| Rounded-up speech frame size (bytes) | 12 | 13 | 15 | 17 | 19 | 20 | 26 | 31 | 5 |
| Rounded-up speech frame size (bits) | 96 | 104 | 120 | 136 | 152 | 160 | 208 | 248 | 40 |
| Payload header and ToC | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 112 | 120 | 136 | 152 | 168 | 176 | 224 | 264 | 56 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 20 ms | 432 | 440 | 456 | 472 | 488 | 496 | 544 | 584 | 376 |
| Total bit-rate (kbps) | 21.6 | 22 | 22.8 | 23.6 | 24.4 | 24.8 | 27.2 | 29.2 | 18.8 |
| AS | 22 | 22 | 23 | 24 | 25 | 25 | 28 | 30 | N/A |

Table K.4: Computation of b=AS for AMR (IPv6, ptime=20, octet-aligned mode)

| Mode | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 | SID |
|--------------------------------------|-------|-------|-------|-------|------|-------|------|------|------|
| Bits per speech frame | 95 | 103 | 118 | 134 | 148 | 159 | 204 | 244 | 39 |
| Speech frame size (bytes) | 11.88 | 12.88 | 14.75 | 16.75 | 18.5 | 19.88 | 25.5 | 30.5 | 4.88 |
| Rounded-up speech frame size (bytes) | 12 | 13 | 15 | 17 | 19 | 20 | 26 | 31 | 5 |
| Rounded-up speech frame size (bits) | 96 | 104 | 120 | 136 | 152 | 160 | 208 | 248 | 40 |
| Payload header and ToC | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 112 | 120 | 136 | 152 | 168 | 176 | 224 | 264 | 56 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 20 ms | 592 | 600 | 616 | 632 | 648 | 656 | 704 | 744 | 536 |
| Total bit-rate (kbps) | 29.6 | 30 | 30.8 | 31.6 | 32.4 | 32.8 | 35.2 | 37.2 | 26.8 |
| AS | 30 | 30 | 31 | 32 | 33 | 33 | 36 | 38 | N/A |

Table K.5: Computation of b=AS for AMR-WB (IPv4, ptime=20, bandwidth-efficient mode)

| Mode | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 | SID |
|--------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|------|
| Bits per speech frame | 132 | 177 | 253 | 285 | 317 | 365 | 397 | 461 | 477 | 40 |
| Payload header and ToC | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| RTP payload (bits) | 142 | 187 | 263 | 295 | 327 | 375 | 407 | 471 | 487 | 50 |
| RTP payload (bytes) | 17.75 | 23.38 | 32.88 | 36.88 | 40.88 | 46.88 | 50.88 | 58.88 | 60.875 | 6.25 |
| Rounded-up RTP payload (bytes) | 18 | 24 | 33 | 37 | 41 | 47 | 51 | 59 | 61 | 7 |
| Rounded-up RTP payload (bits) | 144 | 192 | 264 | 296 | 328 | 376 | 408 | 472 | 488 | 56 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 20 ms | 464 | 512 | 584 | 616 | 648 | 696 | 728 | 792 | 808 | 376 |
| Total bit-rate (kbps) | 23.2 | 25.6 | 29.2 | 30.8 | 32.4 | 34.8 | 36.4 | 39.6 | 40.4 | 18.8 |
| AS | 24 | 26 | 30 | 31 | 33 | 35 | 37 | 40 | 41 | N/A |

Table K.6: Computation of b=AS for AMR-WB (IPv6, ptime=20, bandwidth-efficient mode)

| Mode | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 | SID |
|--------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|------|
| Bits per speech frame | 132 | 177 | 253 | 285 | 317 | 365 | 397 | 461 | 477 | 40 |
| Payload header and ToC | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| RTP payload (bits) | 142 | 187 | 263 | 295 | 327 | 375 | 407 | 471 | 487 | 50 |
| RTP payload (bytes) | 17.75 | 23.38 | 32.88 | 36.88 | 40.88 | 46.88 | 50.88 | 58.88 | 60.875 | 6.25 |
| Rounded-up RTP payload (bytes) | 18 | 24 | 33 | 37 | 41 | 47 | 51 | 59 | 61 | 7 |
| Rounded-up RTP payload (bits) | 144 | 192 | 264 | 296 | 328 | 376 | 408 | 472 | 488 | 56 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 20 ms | 624 | 672 | 744 | 776 | 808 | 856 | 888 | 952 | 968 | 536 |
| Total bit-rate (kbps) | 31.2 | 33.6 | 37.2 | 38.8 | 40.4 | 42.8 | 44.4 | 47.6 | 48.4 | 26.8 |
| AS | 32 | 34 | 38 | 39 | 41 | 43 | 45 | 48 | 49 | N/A |

Table K.7: Computation of b=AS for AMR-WB (IPv4, ptime=20, octet-aligned mode)

| Mode | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 | SID |
|--------------------------------------|------|-------|-------|-------|-------|-------|-------|-------|--------|------|
| Bits per speech frame | 132 | 177 | 253 | 285 | 317 | 365 | 397 | 461 | 477 | 40 |
| Speech frame size (bytes) | 16.5 | 22.13 | 31.63 | 35.63 | 39.63 | 45.63 | 49.63 | 57.63 | 59.625 | 5 |
| Rounded-up speech frame size (bytes) | 17 | 23 | 32 | 36 | 40 | 46 | 50 | 58 | 60 | 5 |
| Rounded-up speech frame size (bits) | 136 | 184 | 256 | 288 | 320 | 368 | 400 | 464 | 480 | 40 |
| Payload header and ToC | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 152 | 200 | 272 | 304 | 336 | 384 | 416 | 480 | 496 | 56 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 20 ms | 472 | 520 | 592 | 624 | 656 | 704 | 736 | 800 | 816 | 376 |
| Total bit-rate (kbps) | 23.6 | 26 | 29.6 | 31.2 | 32.8 | 35.2 | 36.8 | 40 | 40.8 | 18.8 |
| AS | 24 | 26 | 30 | 32 | 33 | 36 | 37 | 40 | 41 | N/A |

Table K.8: Computation of b=AS for AMR-WB (IPv6, ptime=20, octet-aligned mode)

| Mode | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 | SID |
|--------------------------------------|------|-------|-------|-------|-------|-------|-------|-------|--------|------|
| Bits per speech frame | 132 | 177 | 253 | 285 | 317 | 365 | 397 | 461 | 477 | 40 |
| Speech frame size (bytes) | 16.5 | 22.13 | 31.63 | 35.63 | 39.63 | 45.63 | 49.63 | 57.63 | 59.625 | 5 |
| Rounded-up speech frame size (bytes) | 17 | 23 | 32 | 36 | 40 | 46 | 50 | 58 | 60 | 5 |
| Rounded-up speech frame size (bits) | 136 | 184 | 256 | 288 | 320 | 368 | 400 | 464 | 480 | 40 |
| Payload header and ToC | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 152 | 200 | 272 | 304 | 336 | 384 | 416 | 480 | 496 | 56 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 20 ms | 632 | 680 | 752 | 784 | 816 | 864 | 896 | 960 | 976 | 536 |
| Total bit-rate (kbps) | 31.6 | 34 | 37.6 | 39.2 | 40.8 | 43.2 | 44.8 | 48 | 48.8 | 26.8 |
| AS | 32 | 34 | 38 | 40 | 41 | 44 | 45 | 48 | 49 | N/A |

Table K.9: Computation of b=AS for AMR (IPv4, ptime=40, bandwidth-efficient mode)

| Mode | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 |
|--------------------------------|-------|-------|------|------|------|-------|------|------|
| Bits per speech frame | 95 | 103 | 118 | 134 | 148 | 159 | 204 | 244 |
| Payload header and ToC | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 206 | 222 | 252 | 284 | 312 | 334 | 424 | 504 |
| RTP payload (bytes) | 25.75 | 27.75 | 31.5 | 35.5 | 39 | 41.75 | 53 | 63 |
| Rounded-up RTP payload (bytes) | 26 | 28 | 32 | 36 | 39 | 42 | 53 | 63 |
| Rounded-up RTP payload (bits) | 208 | 224 | 256 | 288 | 312 | 336 | 424 | 504 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 40 ms | 528 | 544 | 576 | 608 | 632 | 656 | 744 | 824 |
| Total bit-rate (kbps) | 13.2 | 13.6 | 14.4 | 15.2 | 15.8 | 16.4 | 18.6 | 20.6 |
| AS | 14 | 14 | 15 | 16 | 16 | 17 | 19 | 21 |

Table K.10: Computation of b=AS for AMR (IPv6, ptime=40, bandwidth-efficient mode)

| Mode | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 |
|--------------------------------|-------|-------|------|------|------|-------|------|------|
| Bits per speech frame | 95 | 103 | 118 | 134 | 148 | 159 | 204 | 244 |
| Payload header and ToC | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 206 | 222 | 252 | 284 | 312 | 334 | 424 | 504 |
| RTP payload (bytes) | 25.75 | 27.75 | 31.5 | 35.5 | 39 | 41.75 | 53 | 63 |
| Rounded-up RTP payload (bytes) | 26 | 28 | 32 | 36 | 39 | 42 | 53 | 63 |
| Rounded-up RTP payload (bits) | 208 | 224 | 256 | 288 | 312 | 336 | 424 | 504 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 40 ms | 688 | 704 | 736 | 768 | 792 | 816 | 904 | 984 |
| Total bit-rate (kbps) | 17.2 | 17.6 | 18.4 | 19.2 | 19.8 | 20.4 | 22.6 | 24.6 |
| AS | 18 | 18 | 19 | 20 | 20 | 21 | 23 | 25 |

Table K.11: Computation of b=AS for AMR (IPv4, ptime=40, octet-aligned mode)

| Mode | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 |
|--------------------------------------|------|------|------|------|------|------|------|------|
| Bits per speech frame | 95 | 103 | 118 | 134 | 148 | 159 | 204 | 244 |
| Speech frame size (bytes) | 24 | 26 | 30 | 34 | 38 | 40 | 52 | 62 |
| Rounded-up speech frame size (bytes) | 24 | 26 | 30 | 34 | 38 | 40 | 52 | 62 |
| Rounded-up speech frame size (bits) | 192 | 208 | 240 | 272 | 304 | 320 | 416 | 496 |
| Payload header and ToC | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 |
| RTP payload (bits) | 216 | 232 | 264 | 296 | 328 | 344 | 440 | 520 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 40 ms | 536 | 552 | 584 | 616 | 648 | 664 | 760 | 840 |
| Total bit-rate (kbps) | 13.4 | 13.8 | 14.6 | 15.4 | 16.2 | 16.6 | 19 | 21 |
| AS | 14 | 14 | 15 | 16 | 17 | 17 | 19 | 21 |

Table K.12: Computation of b=AS for AMR (IPv6, ptime=40, octet-aligned mode)

| Mode | 4.75 | 5.15 | 5.9 | 6.7 | 7.4 | 7.95 | 10.2 | 12.2 |
|--------------------------------------|------|------|------|------|------|------|------|------|
| Bits per speech frame | 95 | 103 | 118 | 134 | 148 | 159 | 204 | 244 |
| Speech frame size (bytes) | 24 | 26 | 30 | 34 | 38 | 40 | 52 | 62 |
| Rounded-up speech frame size (bytes) | 24 | 26 | 30 | 34 | 38 | 40 | 52 | 62 |
| Rounded-up speech frame size (bits) | 192 | 208 | 240 | 272 | 304 | 320 | 416 | 496 |
| Payload header and ToC | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 |
| RTP payload (bits) | 216 | 232 | 264 | 296 | 328 | 344 | 440 | 520 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 40 ms | 696 | 712 | 744 | 776 | 808 | 824 | 920 | 1000 |
| Total bit-rate (kbps) | 17.4 | 17.8 | 18.6 | 19.4 | 20.2 | 20.6 | 23 | 25 |
| AS | 18 | 18 | 19 | 20 | 21 | 21 | 23 | 25 |

Table K.13: Computation of b=AS for AMR-WB (IPv4, ptime=40, bandwidth-efficient mode)

| Mode | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 |
|--------------------------------|-----|-------|-------|-------|-------|-------|-------|-------|--------|
| Bits per speech frame | 132 | 177 | 253 | 285 | 317 | 365 | 397 | 461 | 477 |
| Payload header and ToC | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 280 | 370 | 522 | 586 | 650 | 746 | 810 | 938 | 970 |
| RTP payload (bytes) | 35 | 46.25 | 65.25 | 73.25 | 81.25 | 93.25 | 101.3 | 117.3 | 121.25 |
| Rounded-up RTP payload (bytes) | 35 | 47 | 66 | 74 | 82 | 94 | 102 | 118 | 122 |
| Rounded-up RTP payload (bits) | 280 | 376 | 528 | 592 | 656 | 752 | 816 | 944 | 976 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 40 ms | 600 | 696 | 848 | 912 | 976 | 1072 | 1136 | 1264 | 1296 |
| Total bit-rate (kbps) | 15 | 17.4 | 21.2 | 22.8 | 24.4 | 26.8 | 28.4 | 31.6 | 32.4 |
| AS | 15 | 18 | 22 | 23 | 25 | 27 | 29 | 32 | 33 |

Table K.14: Computation of b=AS for AMR-WB (IPv6, ptime=40, bandwidth-efficient mode)

| Mode | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 |
|--------------------------------|-----|-------|-------|-------|-------|-------|-------|-------|--------|
| Bits per speech frame | 132 | 177 | 253 | 285 | 317 | 365 | 397 | 461 | 477 |
| Payload header and ToC | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 280 | 370 | 522 | 586 | 650 | 746 | 810 | 938 | 970 |
| RTP payload (bytes) | 35 | 46.25 | 65.25 | 73.25 | 81.25 | 93.25 | 101.3 | 117.3 | 121.25 |
| Rounded-up RTP payload (bytes) | 35 | 47 | 66 | 74 | 82 | 94 | 102 | 118 | 122 |
| Rounded-up RTP payload (bits) | 280 | 376 | 528 | 592 | 656 | 752 | 816 | 944 | 976 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 40 ms | 760 | 856 | 1008 | 1072 | 1136 | 1232 | 1296 | 1424 | 1456 |
| Total bit-rate (kbps) | 19 | 21.4 | 25.2 | 26.8 | 28.4 | 30.8 | 32.4 | 35.6 | 36.4 |
| AS | 19 | 22 | 26 | 27 | 29 | 31 | 33 | 36 | 37 |

Table K.15: Computation of b=AS for AMR-WB (IPv4, ptime=40, octet-aligned mode)

| Mode | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 |
|--------------------------------------|------|------|-------|-------|-------|-------|-------|-------|-------|
| Bits per speech frame | 132 | 177 | 253 | 285 | 317 | 365 | 397 | 461 | 477 |
| Speech frame size (bytes) | 34 | 46 | 64 | 72 | 80 | 92 | 100 | 116 | 120 |
| Rounded-up speech frame size (bytes) | 34 | 46 | 64 | 72 | 80 | 92 | 100 | 116 | 120 |
| Rounded-up speech frame size (bits) | 272 | 368 | 512 | 576 | 640 | 736 | 800 | 928 | 960 |
| Payload header and ToC | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 |
| RTP payload (bits) | 296 | 392 | 536 | 600 | 664 | 760 | 824 | 952 | 984 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 40 ms | 616 | 712 | 856 | 920 | 984 | 1080 | 1144 | 1272 | 1304 |
| Total bit-rate (kbps) | 15.4 | 17.8 | 21.4 | 23 | 24.6 | 27 | 28.6 | 31.8 | 32.6 |
| AS | 16 | 18 | 22 | 23 | 25 | 27 | 29 | 32 | 33 |

Table K.16: Computation of b=AS for AMR-WB (IPv6, ptime=40, octet-aligned mode)

| Mode | 6.6 | 8.85 | 12.65 | 14.25 | 15.85 | 18.25 | 19.85 | 23.05 | 23.85 |
|--------------------------------------|------|------|-------|-------|-------|-------|-------|-------|-------|
| Bits per speech frame | 132 | 177 | 253 | 285 | 317 | 365 | 397 | 461 | 477 |
| Speech frame size (bytes) | 34 | 46 | 64 | 72 | 80 | 92 | 100 | 116 | 120 |
| Rounded-up speech frame size (bytes) | 34 | 46 | 64 | 72 | 80 | 92 | 100 | 116 | 120 |
| Rounded-up speech frame size (bits) | 272 | 368 | 512 | 576 | 640 | 736 | 800 | 928 | 960 |
| Payload header and ToC | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 |
| RTP payload (bits) | 296 | 392 | 536 | 600 | 664 | 760 | 824 | 952 | 984 |
| RTP header | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 40 ms | 776 | 872 | 1016 | 1080 | 1144 | 1240 | 1304 | 1432 | 1464 |
| Total bit-rate (kbps) | 19.4 | 21.8 | 25.4 | 27 | 28.6 | 31 | 32.6 | 35.8 | 36.6 |
| AS | 20 | 22 | 26 | 27 | 29 | 31 | 33 | 36 | 37 |

Annex L (Normative): Facsimile transmission

L.1 General

This Annex describes Facsimile over IP (FoIP) transmission in MTSI using UDPTL-based transmission, see ITU-T Recommendation T.38, [93].

FoIP is an optional capability for both MTSI client in terminals and MTSI MGWs. This Annex defines the minimum capabilities that need to be supported when FoIP is supported.

L.2 FoIP support in MTSI clients

L.2.1 FoIP support in MTSI client in terminal

An MTSI client in terminal supporting FoIP is typically either a facsimile gateway or a facsimile end-point and does not need to support both cases.

An MTSI client in terminal may support FoIP where the MTSI client in terminal is used as a facsimile gateway between an external Group 3 facsimile equipment and the IMS network. In this case, the MTSI client in terminal acts as an Internet Facsimile Protocol (IFP) peer, either as an emitting gateway or as a receiving gateway depending on whether the MTSI client in terminal initiates the Internet Facsimile Transfer (IFT) or whether it accepts the IFT, [93].

An MTSI client in terminal may support FoIP where the MTSI client in terminal is the end-point for the facsimile transmission. In this case, the facsimile transmission originates or terminates in the MTSI client in terminal.

An MTSI client in terminal supporting FoIP and used as a facsimile gateway shall support:

- input/output to Group 3 facsimile devices, [91]

NOTE: The interface used to connect the external facsimile device to the MTSI client in terminal is outside the scope of this specification.

An MTSI client in terminal supporting FoIP and used either as a facsimile gateway or as a facsimile end-point should support the recommended configuration defined in Clause L.2.3, Table L.1.

- encapsulating and decapsulating T.30 to/from Internet Facsimile Protocol (IFP) packets, [93];

L.2.2 FoIP support in MTSI MGW

An MTSI MGW may support FoIP where the MGW is used as a facsimile gateway between the IMS network and a Circuit Switched (CS) network, e.g. PSTN or CS GERAN, in order to connect to another Group 3 facsimile device.

An MTSI MGW supporting FoIP should support the recommended configuration defined in Clause L.2.3, Table L.1.

L.2.3 Recommended configuration

The recommended configuration for T.38 UDPTL-based FoIP is defined in Table L.1.

Table L.1: Recommended configuration for T.38 UDPTL-based FoIP

| SDP attributes | Value |
|---|---------------------------------|
| T38FaxVersion | 2 (or higher) (NOTE 1) |
| T38MaxBitRate | 14400 bps |
| T38FillBitRemoval | N/A (NOTE 2) |
| T38FaxTranscodingMMR | N/A (NOTE 2) |
| T38FaxTranscodingJBIG | N/A (NOTE 2) |
| T38FaxRateManagement | 'transferredTCF' |
| T38FaxMaxBuffer | 1800 bytes |
| T38FaxMaxDatagram | At least 150 bytes |
| T38FaxMaxIFP | 40 bytes (NOTE 3) |
| T38FaxUdpEC | 't38UDPRedundancy' |
| T38FaxUdpECDepth | 'minred:1', 'maxred:2' (NOTE 3) |
| T38FaxUdpFECMaxSpan | 3 (NOTE 3) |
| T38ModemType | 't38G3FaxOnly' (NOTE 3) |
| NOTE 1: Some SDP attributes listed here apply only to newer versions | |
| NOTE 2: Support not required | |
| NOTE 3: Only applicable when fax version 4 is supported | |
| NOTE 4: See ITU-T T.38, Annex D, Table D.1 for a complete description | |

It is recommended that the MTSI client supports sending and receiving facsimile with 200% redundancy when UDP redundancy is used, even if the SDP attributes and parameters ('T38FaxUdpECDepth' with 'minred' and 'maxred') are not supported. This allows for transmitting each IFP message three times, once as a primary message and twice as redundancy messages.

L.3 Session setup

L.3.1 Session setup for any MTSI client supporting facsimile transmission

An MTSI client supporting facsimile transmission shall support facsimile transmission in stand-alone sessions without any other media types.

NOTE: This does not prevent supporting facsimile transmission also in other session types, for example in speech+facsimile sessions, but this is not described here.

An MTSI client supporting facsimile versions (T38FaxVersion) higher than 0 shall be capable of downgrading the session to any lower facsimile version, if indicated by a received SDP message.

An MTSI client sending an SDP for a facsimile session shall include the following in the SDP (offer or answer):

- MIME media type and subtype names as defined in [94];
- bandwidth information, both on media level and session level;
- T38FaxVersion attribute, if the offered version is higher than 0;
- T38FaxRateManagement attribute, with the value according to the offered method.

Absence of the T38FaxVersion attribute indicates that only version 0 is supported.

SDP examples for facsimile calls can be found in clause L.7.

L.3.2 Session setup when the recommended profile is supported

When the MTSI client supports facsimile transmission according to the recommended profile in Annex L.2.3 and initiates a session for UDPTL-based facsimile transmission then:

- the following SDP lines shall be used in the SDP offer:

- b=AS with the bandwidth set to at least 46 kbps for IPv4 or 48 kbps for IPv6;
- T38FaxVersion attribute indicating at least version 2;
- T38FaxRateManagement attribute with value 'transferredTCF';
- the following SDP attributes should be included in the SDP offer:
 - T38MaxBitRate, the value should be set to 14400;
 - T38FaxMaxBuffer with value 1800;
 - T38FaxMaxDatagram with value 150;
 - T38FaxUdpEC with value 't38UDPRedundancy'.

Other SDP attributes defined in ITU-T T.38 Annex D may be included, if supported.

When the MTSI client supports facsimile transmission according to the recommended profile in Annex L.2.3 and accepts an offer for a session initiation for facsimile transmission then:

- the following SDP lines shall be included in the SDP answer:
 - T38FaxVersion attribute indicating at least version 2;
 - T38FaxRateManagement, the value shall be the same as in the SDP offer;
 - T38FaxUdpEC, the value to include depends both on what error correction schemes the MTSI client supports and what error correction schemes that are declared in the SDP offer;
 - b=AS, the value indicates the bandwidth needed for facsimile transmission and should be aligned with T38MaxBitRate (if included);
- and the following SDP attributes should be included:
 - T38MaxBitRate, the value should be set to 14400;
 - T38FaxMaxBuffer, the value indicates the receiver buffer size.

L.4 Data transport using UDP/IP

An MTSI client in terminal supporting facsimile transmission using UDP/IP shall support:

- encapsulating and decapsulating T.30 [92] into/from Internet Facsimile Protocol (IFP) packets [93];
- UDPTL-based transport format in ITU-T Recommendation T.38 [93], and:
- redundancy transmission of primary IFP packets, see ITU-T Recommendation T.38 Clause 9.1.4.1 [93].

An MTSI client in terminal supporting facsimile transmission using UDP/IP may support:

- the parity FEC scheme specified in ITU-T Recommendation T.38 Annex C [93].

An IFP packet may include either partial, single or multiple HDLC frames.

A T.38 packet may include both one IFP packet and one or more redundancy/FEC information packets.

L.5 CS GERAN inter-working

An MTSI MGW for CS GERAN inter-working and supporting facsimile transmission should support the recommended profile in Annex L.2.3.

L.6 PSTN inter-working

An MTSI MGW for PSTN inter-working and supporting facsimile transmission should support the recommended profile in Annex L.2.3.

L.7 SDP examples

L.7.1 Facsimile-only session

This example shows the media scope of the SDP offer and SDP answer for a facsimile-only session when the recommended configuration in Annex L.2.3 is offered by both end-points.

Table L.2: Example SDP offer for facsimile-only session

| SDP offer |
|--|
| <pre>m=image 49150 udptl t38 b=AS:46 a=T38FaxVersion:2 a=T38FaxRateManagement:transferredTCF a=T38MaxBitRate:14400 a=T38FaxUdpEC:t38UDPRedundancy a=T38FaxMaxBuffer:1800 a=T38FaxMaxDatagram:150</pre> |
| SDP answer |
| <pre>m=image 49154 udptl t38 b=AS:46 a=T38FaxVersion:2 a=T38FaxRateManagement:transferredTCF a=T38MaxBitRate:14400 a=T38FaxUdpEC:t38UDPRedundancy a=T38FaxMaxBuffer:1800 a=T38FaxMaxDatagram:150</pre> |

Comments:

The session bandwidth is set to 46 kbps based on the following calculation:

- the maximum fax bitrate is 14.4 kbps;
- it is recommended that the MTSI client supports 200% redundancy;
- 14.4 kbps and 150 bytes in each IP/UDP packet gives 12 packets per second;
- IPv4 and UDP gives 28 bytes overhead for each IP/UDP packet (IPv6 and UDP gives 48 bytes overhead).

This gives $3 \cdot 14400 + 12 \cdot 28 \cdot 8 = 45888$ bps, which is rounded up to 46 kbps for IPv4 (48 kbps for IPv6).

ITU-T Recommendation T.38 states that only 'T38FaxRateManagement' is mandatory to include in the SDP. There are however other reasons to include the other SDP lines:

- b=AS is included both because this is useful information both for resource allocation in network nodes and the remote end-point, and because it is required by the current specification, see Clause 6.2.5.1;
- the 'T38FaxVersion' attribute needs to be included to declare that version 2 is supported, since the other end-point would otherwise assume that only version 0 is supported;

- the 'T38MaxBitRate', 'T38FaxMaxBuffer' and 'T38FaxMaxDatagram' attributes are very useful to ensure that the remote end-point sends fax media within the limitations of the local end-point;
- The 'T38FaxUdpEc' attribute is very useful information for the remote end-point in case of bad channel conditions.

The SDP attributes 'T38FaxFillBitRemoval', 'T38FaxTranscodingMMR' and 'T38FaxTranscodingJBIG' are not included since the MTSI client is not required to support these options.

The SDP attributes 'T38FaxMaxIFP', 'T38FaxUdpECDepth' and 'T38FaxUdpFECMaxSpan' are not included since these attributes are not defined for T.38 fax version 2.

Annex M (informative): IANA registration information for SDP attributes

M.1 Introduction

This Annex provides the SDP attribute registration information that is referenced from the IANA registry at <http://www.iana.org/>.

M.2 3gpp_sync_info

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp_sync_info

Long-form Attribute Name in English:

3GPP Synchronization Information attribute

Type of Attribute

Media level and Session Level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute specifies whether media streams should be synchronized or not.

Appropriate Attribute Values for this Attribute:

The attribute is a value attribute. The defined values are "Sync" and "No Sync".

MUX Category for this Attribute:

NORMAL

M.3 3gpp_MaxRecvSDUSize

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp_MaxRecvSDUSize

Long-form Attribute Name in English:

3GPP Maximum Receive SDU Size attribute

Type of Attribute

Media level and Sesssion level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute indicates the maximum SDU size (in octets) of the application data (excluding RTP/UDP/IP headers) that can be transmitted to the receiver without segmentation.

Appropriate Attribute Values for this Attribute:

The attribute is a value attribute. The defined values are 1*5DIGIT; 0 to 65535.

MUX Category for this Attribute:

NORMAL

M.4 3gpp_mtsi_app_adapt

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp_mtsi_app_adapt

Long-form Attribute Name in English:

3GPP MTSI RTCP-APP Adaptation attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to negotiate which RTCP-APP request messages that can be used in a session.

Appropriate Attribute Values for this Attribute:

The attribute is a value attribute. The defined values are: "RedReq", "FrameAggReq", "AmrCmr", "EvsRateReq", "EvsBandwidthReq", "EvsParRedReq", "EvsIoModeReq", "EvsPrimaryModeReq".

MUX Category for this Attribute:

IDENTICAL-PER-PT

M.5 predefined_ROI

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

predefined_ROI

Long-form Attribute Name in English:

3GPP predefined video region-of-interest (ROI) attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to negotiate which pre-defined regions of interest can be requested in a video telephony session.

Appropriate Attribute Values for this Attribute:

See clause 6.2.3.4

MUX Category for this Attribute:

IDENTICAL-PER-PT

M.6 bw-info

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

bw-info

Long-form Attribute Name in English:

Additional bandwidth information attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to negotiate additional bandwidth information for sessions where the "b=" -line bandwidth provides insufficient information or no information at all.

Appropriate Attribute Values for this Attribute:

See clause 19.3.

Additional attribute values may be defined in the future.

MUX Category for this Attribute:

IDENTICAL-PER-PT

M.7 ccc_list

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

ccc_list

Long-form Attribute Name in English:

3GPP compact concurrent codec capabilities (CCC) attribute

Type of Attribute

Session level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to indicate the concurrent codec capabilities of a terminal in a compact representation.

Appropriate Attribute Values for this Attribute:

See clause S.5.7.2

MUX Category for this Attribute:

NORMAL

M.8 anbr

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

anbr

Long-form Attribute Name in English:

3GPP access network bitrate recommendation (ANBR) support attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to indicate the UE's ability to use ANBR as an adaptation trigger and also its ability to receive ANBR information from the access network.

Appropriate Attribute Values for this Attribute:

No values. See clause 6.2.9 for detailed usage.

MUX Category for this Attribute:

IDENTICAL

M.9 PLR_adapt

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

PLR_adapt

Long-form Attribute Name in English:

Packet Loss Rate Adaptation

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to describe the media receiver's ability adapt codec configurations based on packet loss rate.

Appropriate Attribute Values for this Attribute:

See TS 26.114 clauses W.1, W.2, and W.3 for ABNF and detailed usage.

MUX Category for this Attribute:

IDENTICAL

M.10 MAXimum-e2e-PLR

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

MAXimum-e2e-PLR

Long-form Attribute Name in English:

Maximum end-to-end PLR of the media receiver

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to describe the maximum tolerable packet loss rate for the media receiver and a means to negotiate how this loss rate can be distributed across different links.

Appropriate Attribute Values for this Attribute:

See TS 26.114 clauses W.4.2 and W.4.3 for ABNF and detailed usage.

MUX Category for this Attribute:

IDENTICAL

M.11 3gpp-qos-hint

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp-qos-hint

Long-form Attribute Name in English:

3GPP QoS hint attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to describe the UE's desired QoS properties from the local access network.

Appropriate Attribute Values for this Attribute:

See TS 26.114 clause 6.2.7.4 for ABNF and detailed usage.

MUX Category for this Attribute:

IDENTICAL

M.12 imageseq

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

imageseq

Long-form Attribute Name in English:

3GPP Image Sequence attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to negotiate the usage of still images in a session.

Appropriate Attribute Values for this Attribute:

See TS 26.114 clause 6.2.11 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

M.13 3gpp_360video

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp_360video

Long-form Attribute Name in English:

3GPP 360-degree Projected Video attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to negotiate the usage of a 360-degree projected video steam in a session .

Appropriate Attribute Values for this Attribute:

See TS 26.114 Annex Y.6.2 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

M.14 itt4rt_group

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

itt4rt_group

Long-form Attribute Name in English:

3GPP ITT4RT Group attribute

Type of Attribute

Session level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to group 360-degree media and overlay media using the mid attribute

Appropriate Attribute Values for this Attribute:

See TS 26.114 Annex Y.6.2.6 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

M.15 3gpp_overlay

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp_overlay

Long-form Attribute Name in English:

3GPP Overlay attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to indicate one or more parameters for configuring the rendering properties of an overlay.

Appropriate Attribute Values for this Attribute:

See TS 26.114 Annex Y.6.4.3 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

M.16 3gpp_360video_replacement

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp_360video_replacement

Long-form Attribute Name in English:

3GPP 360-degree Video Replacement attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to indicate that the content captured in the 360-degree video can be replaced.

Appropriate Attribute Values for this Attribute:

See TS 26.114 Annex Y.6.4.4 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

M.17 3gpp_fisheye

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp_fisheye

Long-form Attribute Name in English:

3GPP 360-degree Fisheye Video attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to indicate a 360-degree fisheye video stream.

Appropriate Attribute Values for this Attribute:

See TS 26.114 Annex Y.6.5.2 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

M.18 3gpp-camera-calibration

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

3gpp-camera-calibration

Long-form Attribute Name in English:

3GPP Camera Calibration attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used for signaling of camera calibration parameters for network-based stitching.

Appropriate Attribute Values for this Attribute:

See TS 26.114 Annex Y.6.6 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

M.19 stitch_group

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

stitch_group

Long-form Attribute Name in English:

3GPP Stitch Group attribute

Type of Attribute

Session level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to group the to-be-stitched 2D video captures using the mid attribute for network-based stitching.

Appropriate Attribute Values for this Attribute:

See TS 26.114 Annex Y.6.6 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

M.20 no_other_overlays

Contact name, email address, and telephone number:

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Attribute Name (as it will appear in SDP)

no_other_overlays

Long-form Attribute Name in English:

3GPP No Other Overlays attribute

Type of Attribute

Media level

Is Attribute Value subject to the Charset Attribute?

This Attribute is not dependent on charset.

Purpose of the attribute:

This attribute is used to indicate that the MRF shall not group the 360-degree media stream from that ITT4RT-Tx client with overlay media streams from other ITT4RT clients.

Appropriate Attribute Values for this Attribute:

See TS 26.114 Annex Y.6.8.2 for ABNF and detailed usage.

MUX Category for this Attribute:

NORMAL

Annex N (informative): Computation of b=AS for Video Codec

N.1 General

If an MTSI client includes any video codecs in the SDP offer or answer, procedures to compute b=AS are left to the discretion of the implementation.

N.2 Examples

Table N.x shows example values of b=AS in IPv4 and IPv6 for each source bit-rate and image size pair of H.264/AVC. These values are determined from encoding a variety of content with the codec targeting a particular source bit-rate and then measuring the fluctuations of the encoded bit-rate in order to determine the recommended b=AS values. As the source bit-rate increases, the relative differences of b=AS values in IPv4 and IPv6 decrease as the proportion of RTP/UDP/IP headers in the packet decreases, but the required margin for the fluctuations increases. Different source bit-rates and b=AS values can be used for the same image size depending on service policy, alternative codec configurations, and the amount of bit-rate variation introduced by the rate control algorithm implementation.

Table N.1: Example of b=AS values for H.264/AVC

| Image size | Source bit-rate | b=AS (IPv4) | b=AS (IPv6) | Notes |
|------------|-----------------|-------------|-------------|--|
| 176×144 | 48 | 64 | 66 | Maximum SDU size=1400 octets, average packet rate=14 packets/s, picture rate=7 pictures/s, average packets per picture=2 packets/picture, average header (RTP/UDP/IP) bit-rate=5/7 kbps (IPv4/IPv6) |
| 320×240 | 300 | 384 | 399 | Maximum SDU size=1400 octets, average packet rate=60 packets/s, picture rate=15 pictures/s, average packets per picture=4 packets/picture, average header (RTP/UDP/IP) bit-rate=20/32 kbps (IPv4/IPv6) |
| 640×480 | 512 | 639 | 653 | Maximum SDU size=1400 octets, average packet rate=60 packets/s, picture rate=15 pictures/s, average packets per picture=4 packets/picture, average header (RTP/UDP/IP) bit-rate=20/32 kbps (IPv4/IPv6) |

Annex O (informative): IANA registration information for RTP Header Extensions

O.1 Introduction

This Annex provides the RTP header extension registration information that is referenced from the IANA registry at <http://www.iana.org/>.

O.2 urn:3gpp:video-orientation

The desired extension naming URI:

urn:3gpp:video-orientation

A formal reference to the publicly available specification:

TS 26.114

A short phrase describing the function of the extension:

Coordination of video orientation (CVO) feature, see clause 6.2.3.3

Contact information for the organization or person making the registration

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

O.3 urn:3gpp:video-orientation:6

The desired extension naming URI:

urn:3gpp:video-orientation:6

A formal reference to the publicly available specification:

TS 26.114

A short phrase describing the function of the extension:

Higher granularity (6-bit) coordination of video orientation (CVO) feature, see clause 6.2.3.3

Contact information for the organization or person making the registration

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

O.4 urn:3gpp:roi-sent

The desired extension naming URI:

urn:3gpp:roi-sent

A formal reference to the publicly available specification:

TS 26.114

A short phrase describing the function of the extension:

Signalling of the arbitrary region-of-interest (ROI) information for the sent video, see clause 6.2.3.4

Contact information for the organization or person making the registration

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

O.5 urn:3gpp:predefined-roi-sent

The desired extension naming URI:

urn:3gpp:predefined-roi-sent

A formal reference to the publicly available specification:

TS 26.114

A short phrase describing the function of the extension:

Signalling of the predefined region-of-interest (ROI) information for the sent video, see clause 6.2.3.4

Contact information for the organization or person making the registration

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

O.6 urn:3gpp:audio-mixing-gain

The desired extension naming URI:

urn:3gpp:audio-mixing-gain

A formal reference to the publicly available specification:

TS 26.114

A short phrase describing the function of the extension:

Signalling of the audio mixing gain header extension for the sent audio, see clause Y.X.1

Contact information for the organization or person making the registration

3GPP Specifications Manager

3gppContact@etsi.org

+33 (0)492944200

Annex P (informative): Video packet loss handling operation principles and examples

P.1 General

This annex describes operation principles and provides examples to video packet loss handling scheme described in section 9.3. Several different video packet loss handling behaviours are possible at both sender and receiver ends for responding and reporting, respectively. Example criteria shown in this section are not to be seen as a scheme that excludes other designs. Implementers are free to use any packet loss algorithm as long as the requirements and recommendations specified in clause 9.3 are fulfilled.

P.2 Video error recovery

Efficient video error recovery requires error tracking capabilities at both the sender and the receiver side. Error detection and tracking is necessary on the receiver side for detecting the occurrence of the error as well as detecting the recovery from the error. On the sender side it is necessary for producing a recovery picture that would address the reported packet loss. Basically a receiver should be able to detect errors and report them to the sender in timely fashion. In return sender responds by sending recovery pictures or performing gradual decoder refresh (GDR).

An example of video error recovery is illustrated in Figure P.1 below using a NACK message.

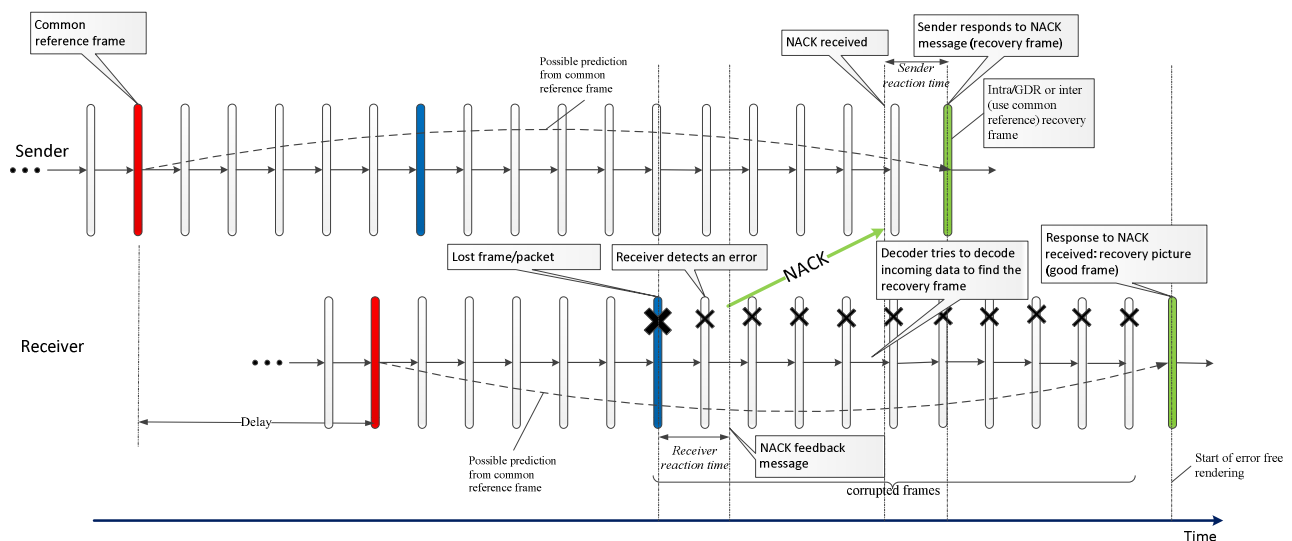


Figure P.1: Video error recovery using NACK feedback message.

In this example, the error correction is performed in the following steps:

- 1) Sender encodes a reference picture (blue) and transmits it. One or more of the packets belonging to this picture are lost.
- 2) Receiver detects lost packets belonging to the blue picture upon receiving packets belonging to the picture following the blue picture or the last packet (if received) of the blue picture, after de-jittering.
- 3) When the decoder tries to decode the picture following the blue picture and notices that a reference picture that it is referring to (i.e. the blue picture) is missing or has been partially received, and in response flags an error.
- 4) Upon seeing the error report from the decoder, the receiver issues a NACK message. The duration of time that elapses from the first detection of missing packets to the issuance of the feedback message is denoted as the receiver reaction time.

- 5) Sender receives the NACK message, feeds this information to the encoder, which responds by encoding the next picture either as an intra or inter picture. Alternatively the encoder can generate GDR over next N frames. In the inter-picture case, the encoder refers to a reference picture (red) that it assumes can be correctly decoded at the receiver side. The duration of time that elapses from receipt of the feedback message and sending of the recovery picture is denoted as the sender reaction time.
- 6) Sender sends the recovery picture or the GDR to the receiver.
- 7) Receiver's decoder continues to decode incoming pictures looking for the arrival of the recovery picture or full refresh from GDR. The receiver may opt not to render any incoming corrupted pictures while waiting for the arrival of the recovery picture or full refresh.

If the recovery picture does not arrive in response wait time duration (RWT) then the receiver should issue another NACK message to request error recovery and wait for recovery. If the recovery still does not occur within another RWT, then it starts issuing PLI messages to request IDR or GDR recovery. This is illustrated in Figure P.2 below.

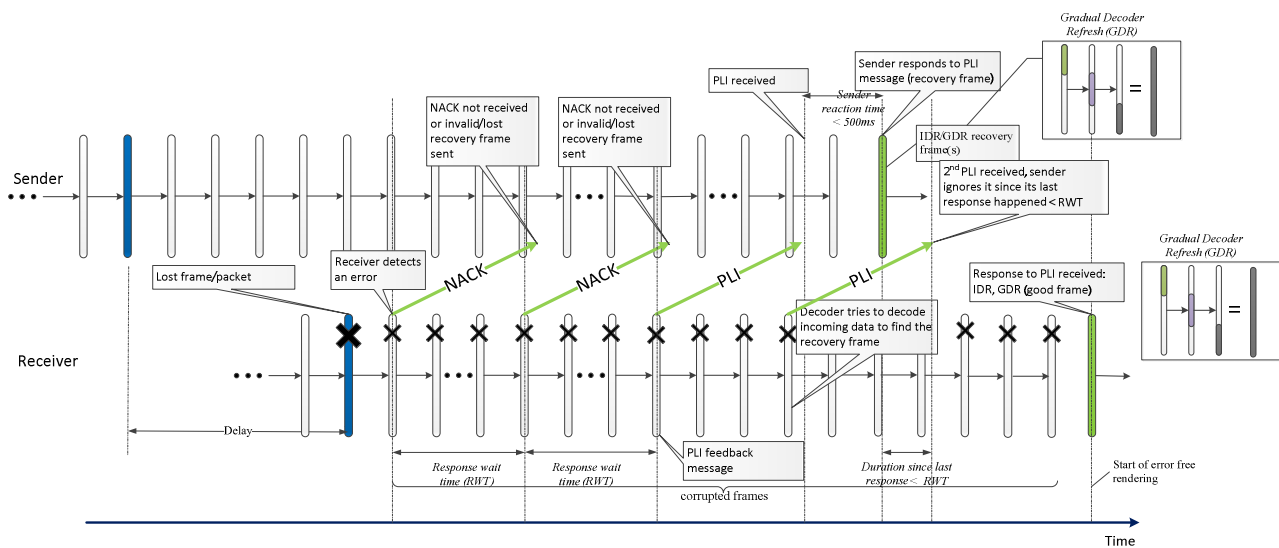


Figure P.2: Video error recovery using PLI feedback message.

PLI request becomes necessary when the likelihood of having a common reference frame for inter error recovery is diminished. In this example, the error correction is performed in the following steps:

- 1) Receiver issues a PLI message after waiting for two RWT duration for a recovery picture requested by NACK messages to arrive from the onset of the error.
- 2) Sender upon reception of the PLI message, encodes the next picture as IDR picture or starts a GDR.
- 3) Receiver receives the IDR picture or the GDR pictures resulting in full refresh.

In the above example, a second PLI is received by the sender within RWT interval. In this case, the sender ignores the second PLI since the receiver cannot detect the arrival of the first sent IDR/GDR within this time frame. The same principle applies to NACK messages as well. This would also apply to cases where the sender has sent a picture that could serve as a recovery picture (not triggered by a PLI/NACK message) prior to the reception of a PLI/NACK message within RWT duration. In this case the sender does not have to respond to the received PLI/NACK message as illustrated in Figure P.3 below.

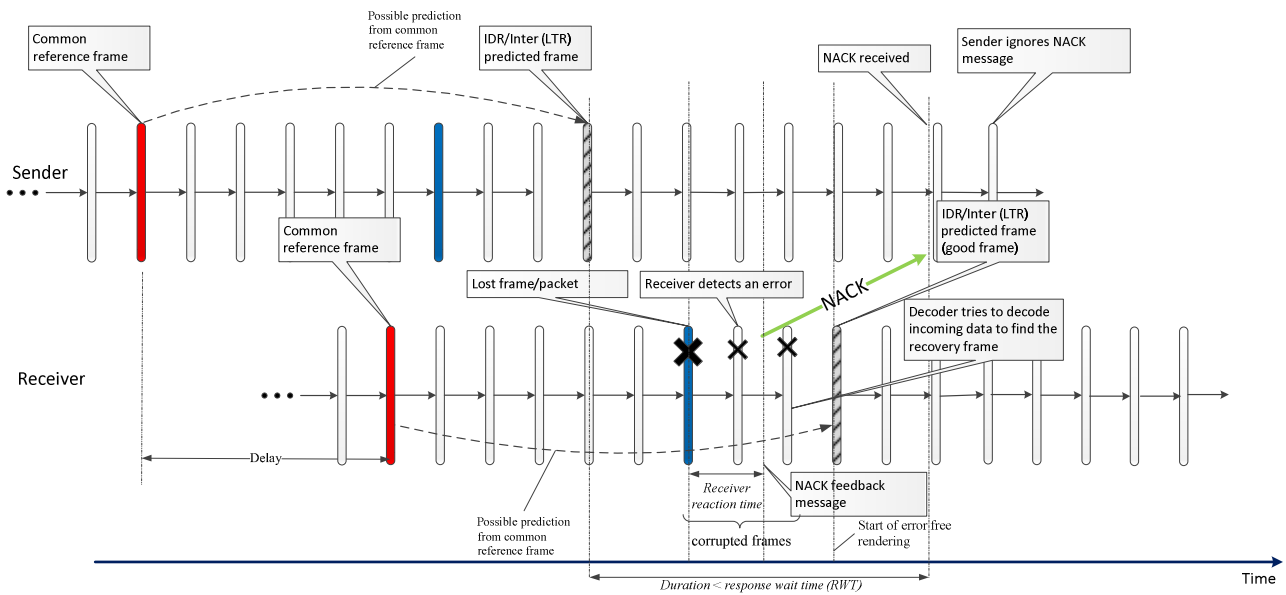


Figure P.3: Example case where sender does not have to respond to incoming NACK/PLI messages.

This case would apply to schemes where the sender periodically performs some form of periodic intra refresh or inter recovery (periodically predict from long term reference (LTR) pictures) as long as the period conforms to the timing restrictions defined in section 9.3.

P.3 RTP Retransmission

RTP retransmission offers retransmission of lost packets reported by NACK feedback. The receiver detects missing packets and requests retransmission of missing packets. The sender upon receiving the NACK message decides to take corrective action by retransmitting the reported missing packets to the receiver. If retransmitted packets arrive in time for rendering, then timely recovery is achieved.

An example of recovery from error is illustrated in Figure P.4 below using a NACK message.

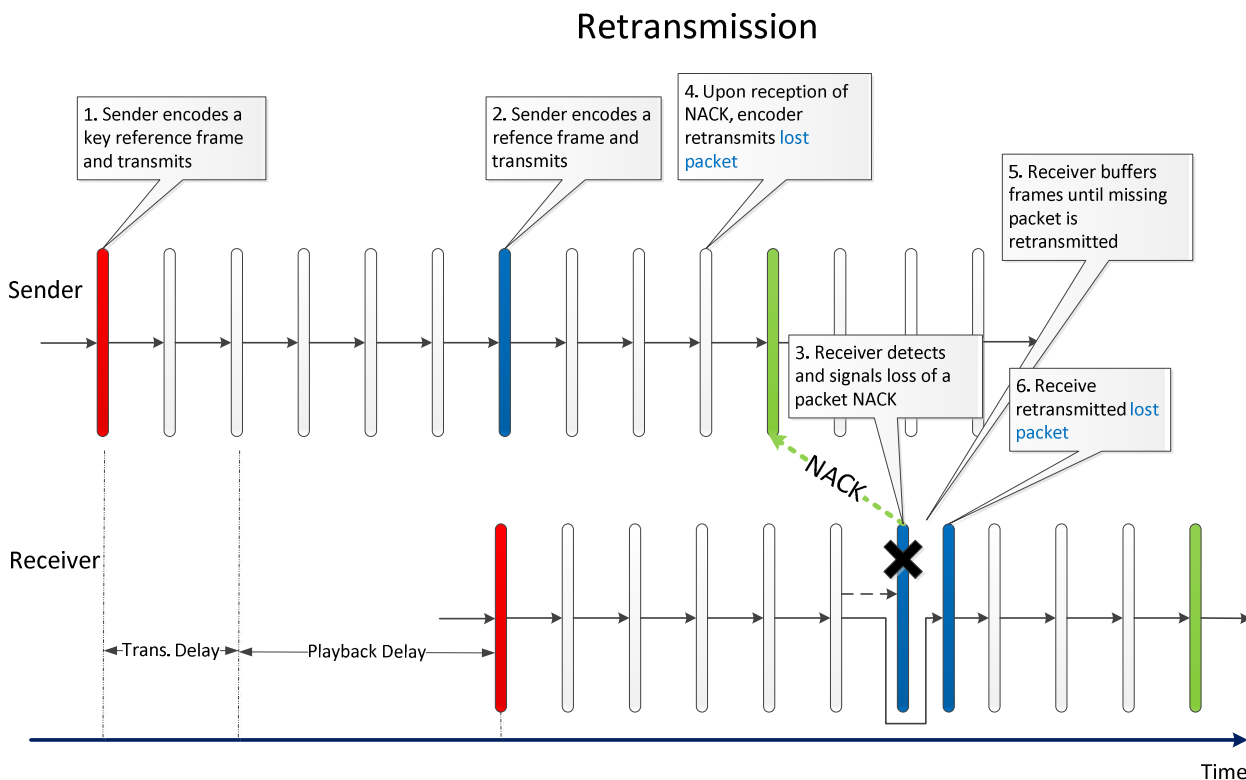


Figure P.4: Video error recovery using NACK feedback message and retransmission.

In this example, the error correction is performed in the following steps:

- 1) Sender encodes a reference picture (blue) and transmits it. Sender stores RTP packets corresponding to this frame in its buffers. One or more of the packets belonging to this picture are lost.
- 2) Receiver detects lost packets belonging to the blue picture upon receiving packets belonging to the picture following the blue picture or the last packet (if received) of the blue picture, after de-jittering.
- 3) The receiver issues a NACK message and pauses decoding while caching incoming packets.
- 4) Sender receives the NACK message, checks whether the requested packets are available in its cache. If they are, it retransmits the requested packets.
- 5) Receiver monitors the incoming packets to determine the arrival of the requested packets to resume decoding.
- 6) If packets arrive in time, rendering is not interrupted.

Annex Q (informative): Computation of b=AS for EVS

Q.1 General

This annex contains examples of computing b=AS for EVS Primary mode when ptime=20, and ptime=40. In these examples, it is assumed that no extra bandwidth is allocated for redundancy.

Q.2 Procedure for computing the bandwidth

The bandwidth is calculated using the following procedure when no extra bandwidth is allocated for redundancy:

- 1) Calculate the size of the RTP payload, see below.
- 2) Calculate the size of the IP packets by taking the RTP payload size (in bytes) and adding the IP/UDP/RTP overhead: 20 bytes for IPv4; 40 bytes for IPv6; 8 bytes for UDP; 12 bytes for RTP.
- 3) Convert the IP packet size to bits.
- 4) Calculate the required bit-rate (bps) given the packet size and the packet rate: 50 packets per second for 1 frame per packet; 25 packets per second for 2 frames per packet.
- 5) The b=AS bandwidth is then calculated by converting the required bit-rate to kbps and rounding to the nearest higher integer value.

If the SDP includes multiple codecs and/or configurations, the bandwidth is calculated for each configuration and the b=AS bandwidth is set to the highest of the bandwidths.

Q.3 Computation of RTP payload size

When the b=AS bandwidth is computed, it is assumed that the codec is using the highest allowed bit-rate for each frame.

The RTP payload size for the 2 bytes header-full payload format and 1 frame/packet is calculated from the following components:

- 8 bits for the codec mode request (CMR)
- 8 bits for the table of content (ToC)
- N bits for the speech frame (size depends on bit-rate)
- No padding in the end of the RTP payload is needed since each item is already an integer number of octets

The RTP payload size for the 2 bytes header-full payload format and 2 frames/packet is calculated from the following components:

- 8 bits for the codec mode request (CMR)
- 16 bits for the table of content (ToC)
- N bits for the speech frame (size depends on bit-rate)
- No padding in the end of the RTP payload is needed since each item is already an integer number of octets

Q.4 Detailed computation

The tables below give a detailed description of the bandwidth computation. The b=AS bandwidth is not defined for SID.

NOTE 1: The tables below apply for both the header-full format and the compact format of EVS Primary mode, as switching between the modes can occur during the session.

For each codec mode, b=AS value of EVS AMR-WB IO mode is the same as the b=AS value of AMR-WB in octet-aligned mode. See clause K.4.

NOTE 2: The tables in clause K.4 also apply for the compact format of EVS AMR-WB IO mode, as switching between the modes can occur during the session.

NOTE 3: In tables Q.1 and Q.2, the speech frame of 7.2 kbps in EVS Primary mode at ptime=20 is zero-padded by one byte to avoid the protected payload size of 8.0 kbps, but zero-padding by additional byte(s) is also allowed.

Table Q.1: Computation of b=AS for EVS Primary mode (IPv4, ptime=20)

| Mode | 7.2 | 8 | 9.6 | 13.2 | 16.4 | 24.4 | 32 | 48 | 64 | 96 | 128 | SID |
|---------------------------|------|------|------|------|------|------|------|------|------|-------|-------|------|
| Bits per speech frame | 144 | 160 | 192 | 264 | 328 | 488 | 640 | 960 | 1280 | 1920 | 2560 | 48 |
| Speech frame size (bytes) | 19 | 20 | 24 | 33 | 41 | 61 | 80 | 120 | 160 | 240 | 320 | 6 |
| CMR and ToC (bits) | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 168 | 176 | 208 | 280 | 344 | 504 | 656 | 976 | 1296 | 1936 | 2576 | 64 |
| RTP header (bits) | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header (bits) | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header (bits) | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 20 ms | 488 | 496 | 528 | 600 | 664 | 824 | 976 | 1296 | 1616 | 2256 | 2896 | 384 |
| Total bit-rate (kbps) | 24.4 | 24.8 | 26.4 | 30 | 33.2 | 41.2 | 48.8 | 64.8 | 80.8 | 112.8 | 144.8 | 19.2 |
| AS (kbps) | 25 | 25 | 27 | 30 | 34 | 42 | 49 | 65 | 81 | 113 | 145 | N/A |

Table Q.2: Computation of b=AS for EVS Primary mode (IPv6, ptime=20)

| Mode | 7.2 | 8 | 9.6 | 13.2 | 16.4 | 24.4 | 32 | 48 | 64 | 96 | 128 | SID |
|---------------------------|------|------|------|------|------|------|------|------|------|-------|-------|------|
| Bits per speech frame | 144 | 160 | 192 | 264 | 328 | 488 | 640 | 960 | 1280 | 1920 | 2560 | 48 |
| Speech frame size (bytes) | 19 | 20 | 24 | 33 | 41 | 61 | 80 | 120 | 160 | 240 | 320 | 6 |
| CMR and ToC (bits) | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| RTP payload (bits) | 168 | 176 | 208 | 280 | 344 | 504 | 656 | 976 | 1296 | 1936 | 2576 | 64 |
| RTP header (bits) | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header (bits) | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header (bits) | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 20 ms | 648 | 656 | 688 | 760 | 824 | 984 | 1136 | 1456 | 1776 | 2416 | 3056 | 544 |
| Total bit-rate (kbps) | 32.4 | 32.8 | 34.4 | 38 | 41.2 | 49.2 | 56.8 | 72.8 | 88.8 | 120.8 | 152.8 | 27.2 |
| AS (kbps) | 33 | 33 | 35 | 38 | 42 | 50 | 57 | 73 | 89 | 121 | 153 | N/A |

Table Q.3: Computation of b=AS for EVS Primary mode (IPv4, ptime=40)

| Mode | 7.2 | 8 | 9.6 | 13.2 | 16.4 | 24.4 | 32 | 48 | 64 | 96 | 128 |
|---------------------------|------|------|------|------|------|------|------|------|------|-------|-------|
| Bits per speech frame | 144 | 160 | 192 | 264 | 328 | 488 | 640 | 960 | 1280 | 1920 | 2560 |
| Speech frame size (bytes) | 18 | 20 | 24 | 33 | 41 | 61 | 80 | 120 | 160 | 240 | 320 |
| CMR and ToC (bits) | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 |
| RTP payload (bits) | 312 | 344 | 408 | 552 | 680 | 1000 | 1304 | 1944 | 2584 | 3864 | 5144 |
| RTP header (bits) | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header (bits) | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header (bits) | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 40 ms | 632 | 664 | 728 | 872 | 1000 | 1320 | 1624 | 2264 | 2904 | 4184 | 5464 |
| Total bit-rate (kbps) | 15.8 | 16.6 | 18.2 | 21.8 | 25 | 33 | 40.6 | 56.6 | 72.6 | 104.6 | 136.6 |
| AS (kbps) | 16 | 17 | 19 | 22 | 25 | 33 | 41 | 57 | 73 | 105 | 137 |

Table Q.4: Computation of b=AS for EVS Primary mode (IPv6, ptime=40)

| Mode | 7.2 | 8 | 9.6 | 13.2 | 16.4 | 24.4 | 32 | 48 | 64 | 96 | 128 |
|---------------------------|------|------|------|------|------|------|------|------|------|-------|-------|
| Bits per speech frame | 144 | 160 | 192 | 264 | 328 | 488 | 640 | 960 | 1280 | 1920 | 2560 |
| Speech frame size (bytes) | 18 | 20 | 24 | 33 | 41 | 61 | 80 | 120 | 160 | 240 | 320 |
| CMR and ToC (bits) | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 |
| RTP payload (bits) | 312 | 344 | 408 | 552 | 680 | 1000 | 1304 | 1944 | 2584 | 3864 | 5144 |
| RTP header (bits) | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header (bits) | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header (bits) | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 40 ms | 792 | 824 | 888 | 1032 | 1160 | 1480 | 1784 | 2424 | 3064 | 4344 | 5624 |
| Total bit-rate (kbps) | 19.8 | 20.6 | 22.2 | 25.8 | 29 | 37 | 44.6 | 60.6 | 76.6 | 108.6 | 140.6 |
| AS (kbps) | 20 | 21 | 23 | 26 | 29 | 37 | 45 | 61 | 77 | 109 | 141 |

Table Q.5: Computation of b=AS for EVS Primary mode (IPv4, ptime=20, dual-mono)

| Mode | 7.2 | 8 | 9.6 | 13.2 | 16.4 | 24.4 | 32 | 48 | 64 | 96 | 128 |
|---------------------------|------|------|------|------|------|------|------|-------|-------|-------|-------|
| Bits per speech frame | 144 | 160 | 192 | 264 | 328 | 488 | 640 | 960 | 1280 | 1920 | 2560 |
| Speech frame size (bytes) | 18 | 20 | 24 | 33 | 41 | 61 | 80 | 120 | 160 | 240 | 320 |
| CMR and ToC (bits) | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 |
| RTP payload (bits) | 312 | 344 | 408 | 552 | 680 | 1000 | 1304 | 1944 | 2584 | 3864 | 5144 |
| RTP header (bits) | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header (bits) | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv4 header (bits) | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 |
| Total bits per 20 ms | 632 | 664 | 728 | 872 | 1000 | 1320 | 1624 | 2264 | 2904 | 4184 | 5464 |
| Total bit-rate (kbps) | 31.6 | 33.2 | 36.4 | 43.6 | 50 | 66 | 81.2 | 113.2 | 145.2 | 209.2 | 273.2 |
| AS (kbps) | 32 | 34 | 37 | 44 | 50 | 66 | 82 | 114 | 146 | 210 | 274 |

Table Q.6: Computation of b=AS for EVS Primary mode (IPv6, ptime=20, dual-mono)

| Mode | 7.2 | 8 | 9.6 | 13.2 | 16.4 | 24.4 | 32 | 48 | 64 | 96 | 128 |
|---------------------------|------|------|------|------|------|------|------|-------|-------|-------|-------|
| Bits per speech frame | 144 | 160 | 192 | 264 | 328 | 488 | 640 | 960 | 1280 | 1920 | 2560 |
| Speech frame size (bytes) | 18 | 20 | 24 | 33 | 41 | 61 | 80 | 120 | 160 | 240 | 320 |
| CMR and ToC (bits) | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 |
| RTP payload (bits) | 312 | 344 | 408 | 552 | 680 | 1000 | 1304 | 1944 | 2584 | 3864 | 5144 |
| RTP header (bits) | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | 96 |
| UDP header (bits) | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 |
| IPv6 header (bits) | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 | 320 |
| Total bits per 20 ms | 792 | 824 | 888 | 1032 | 1160 | 1480 | 1784 | 2424 | 3064 | 4344 | 5624 |
| Total bit-rate (kbps) | 39.6 | 41.2 | 44.4 | 51.6 | 58 | 74 | 89.2 | 121.2 | 153.2 | 217.2 | 281.2 |
| AS (kbps) | 40 | 42 | 45 | 52 | 58 | 74 | 90 | 122 | 154 | 218 | 282 |

Annex R (informative): IANA registration information for RTCP Feedback Message Types

R.1 Video Region-of-Interest (ROI)

The RTCP feedback types for 'Arbitrary ROI' and 'Pre-defined ROI' is registered with IANA as follows:

Value name: 3gpp-roi-arbitrary

Long name: Video region-of-interest (ROI) arbitrarily selected by the endpoint

Mux-Category: IDENTICAL-PER-PT

Reference: TS 26.114.

Value name: 3gpp-roi-predefined

Long name: Video region-of-interest (ROI) pre-defined by the sender and selected by the endpoint

Mux-Category: IDENTICAL-PER-PT

Reference: TS 26.114.

The following value can be registered as one FMT value in the "FMT Values for PSFB Payload Types" registry <http://www.iana.org/assignments/rtp-parameters>:

Name: ROI

Long name: Video region-of-interest (ROI)

Value: 9

Reference: TS 26.114.

R.2 Delay Budget Information (DBI)

The RTCP feedback types for Delay Budget Information (DBI) is registered with IANA as follows:

Value name: 3gpp-delay-budget

Long name: Available or requested delay budget specified in milliseconds

Mux-Category: IDENTICAL-PER-PT

Reference: TS 26.114.

The following value can be registered as one FMT value in the "FMT Values for RTPFB Payload Types" registry <http://www.iana.org/assignments/rtp-parameters>:

Name: DBI

Long name: Delay Budget Information (DBI)

Value: 10

Reference: TS 26.114.

R.3 Viewport (VP)

The RTCP feedback types for 'Viewport' is registered with IANA as follows:

Value name: 3gpp-viewport

Long name: Viewport feedback for viewport-dependent 360-degree video

Mux-Category: IDENTICAL-PER-PT

Reference: 3GPP TS 26.114.

The following value is registered as one FMT value in the "FMT Values for PSFB Payload Types" registry
<http://www.iana.org/assignments/rtp-parameters>:

Name: VP

Long name: Viewport (VP)

Value: 11

Reference: 3GPP TS 26.114.

Annex S (normative): Multi-party Multimedia Conference Media Handling

S.1 General

This annex describes an extension to MTSI, which is optional to implement for an MTSI client. It contains descriptions of both mandatory and optional functionalities for the particular type of MTSI client that is called MSMTSI client, which is better suited for group audio/video communication than a regular MTSI client. The specifications in this Annex apply in addition to the rest of this specification, except when it is explicitly stated that the text here replaces other parts of this specification.

The intention with this extension is to avoid transcoding as far as possible in multiparty calls. For example, without this extension, video conferencing would need transcoding in the MRF to compose the different videos received from different senders into a single video, containing a main video and a number of thumbnails, which is then sent to each receiver. It may happen that the MRF needs to compose different video compositions to different receivers, which multiplies the need for transcoding and thus scales poorly with the size of the group. With this extension, the MRF can forward the necessary streams, without transcoding, to each MSMTSI client in terminal, which then composes the final image that is presented to the user.

Avoiding transcoding reduces the complexity in the MRF and reduces the end-to-end delay. Quality degradations caused by the transcoding can often also be avoided. The benefits with this extension are further described in TR 26.980 [152].

S.2 Video

S.2.1 Conversational video

An MSMTSI client in the terminal shall be capable of receiving and locally composing at least one main video and one or more video thumbnails. A "thumbnail" video is in this context defined as a receive-only video "m="-line that is not the first video "m="-line in the SDP, and that is also not identified with any "a=content:main" or "a=content:slides".

An MSMTSI client in terminal shall support receiving at least one thumbnail and may also support receiving any number of additional thumbnails, subject to MSMTSI client capability. An MSMTSI MRF shall support sending at least two thumbnails and may support sending any number of additional thumbnails, subject to MSMTSI MRF capability.

An MSMTSI client in terminal shall support sending at least one thumbnail-sized simulcast format of the main video, and may support sending also other simulcast formats. An MSMTSI MRF shall support receiving at least one thumbnail-sized simulcast format of the main video, and may support receiving also other simulcast formats.

S.2.2 Non-conversational (screenshare) video

An MSMTSI client may support sending and receiving screenshare video. The first picture of the screen sharing video an MSMTSI client sends after being granted the screenshare BFCP floor (see clause S.5.6) shall be random accessible, i.e., as if a FIR would have been received.

S.3 Audio

S.3.1 General

An MSMTSI client in terminal shall be capable of receiving and may be capable of sending multiple simultaneous audio RTP streams. The number of multiple audio streams received at the MSMTSI client may be different than the number of multiple audio streams sent from the same MSMTSI client.

Support for multiple audio streams in the direction from an MSMTSI MRF to an MSMTSI client in the terminal shall be interpreted as originating from different group call participants.

An MSMTSI client in terminal shall support local mixing of received audio streams, and may support use of spatial rendering tools, such as local Head-Related Transfer Function (HRTF), to perform audio panning and mixing of the multiple audio streams. Audio panning may enable the rendering device to choose to vary the audio levels of participants by adjusting the mixing gains.

Multi-stream audio is not to be confused with multichannel audio where multi-stream audio may include one or more of mono, stereo, or multichannel audio RTP streams originating from different group call participants.

S.3.2 De-jitter buffer

The functional requirements for jitter-buffer management of MSMTSI client in terminal shall meet the same minimum performance requirements per audio stream that are set for MTSI clients (Clause 8).

S.4 SIP

S.4.1 MSMTSI client in terminal

An MSMTSI client in terminal shall, when connecting to a remote party that included the "isFocus" tag in any of its SIP headers, subscribe to SIP conference events, as described in [147]. The MSMTSI client in terminal shall be able to receive and handle conference event notifications, resulting from the subscription. The MSMTSI client in terminal shall be capable of parsing all conference event information elements, shall be capable of receiving partial conference event information, and shall be capable of presenting conference event information to the user of the MSMTSI client in terminal, either automatically or on explicit user request. The MSMTSI client in terminal shall be capable of presenting at least the values of the conference event information elements indicated in **bold** below to the user of the MSMTSI client in the terminal, and shall be capable to use the conference event information elements indicated in *italics* below for endpoint and media identification, as defined by [147] and [148] (here also indicating relevant parts of the XML document hierarchy, for information):

- <conference-info>
 - <conference-description>
 - **<display-text>**
 - **<subject>**
 - <users>
 - <user>
 - **<display-text>**
 - <associated-aors>
 - <entry>
 - <uri>
 - <endpoint>
 - <media>
 - <src-id>

Values for other conference event information elements may optionally be made available to the user of the MSMTSI client in the terminal.

S.4.2 MSMTSI MRF

An MSMTSI MRF shall include the "isFocus" tag in all of its outgoing SIP headers that support inserting that tag. It shall be capable of handling subscriptions and unsubscriptions for conference event information, as specified in [147].

S.5 Media configuration

S.5.1 General

An MSMTSI client that receives an SDP offer with "m="-lines that it cannot handle or does not understand shall use regular SDP offer/answer procedures [8] to individually reject those unsupported "m="-lines. An MSMTSI client shall not send RTP or RTCP for rejected "m="-lines.

An MSMTSI client shall support controlling its maximum sending rate per "m="-line for media related to that "m="-line, as described by clause 6.2.5.

A "common codec" is a codec (typically a legacy/older generation codec) that is supported by all the participants in the conference and enables transcoder-free MSMTSI conferencing. If all participants share support for more than a single codec for a certain media type, the codec providing the best media quality for a given bitrate should be chosen by the MSMTSI MRF as the "common codec".

A "preferred codec" is a codec (typically a better compression performance, newer generation codec) that is supported by some but not all participants in the conference and enables better media quality.

The common codec and preferred codec(s) may be determined by:

- 1) the codecs that were offered or pre-selected by the conference participant that initiated the conference (ad hoc or pre-scheduled)
- 2) the codecs supported by other conference participants, e.g., by use of SIP OPTIONS (see clause S.5.7.3).

When setting up individual sessions with the conference participants, the MSMTSI MRF:

- 1) shall include the common codec in the SDP offer/answer negotiation, and
- 2) may additionally include the preferred codecs in the SDP offer/answer negotiation to improve conference quality or performance.

To avoid transcoding when multiple codecs of a media type are used in an MSMTSI session,

- 1) simulcast [154] shall be negotiated;
- 2) usage of both preferred and common codecs in the SDP shall be supported;
- 3) the MSMTSI MRF shall include the preferred codecs in the SDP as being simulcast with a corresponding common codec stream for the same media type; and
- 4) a participant sending media using a preferred codec shall also simulcast a representation of the same media using the common codec for that media type.

When constructing "a=rid" [155] line identification of simulcast formats, MSMTSI clients in terminal and MSMTSI MRF shall use a 1:1 mapping per direction between each rid-id and the corresponding RTP payload type number in the "pt=" parameter on the "a=rid" line. It is optional for MSMTSI clients in terminal and MSMTSI MRF to support constraints parameters for the "a=rid" lines. An MSMTSI client in terminal and MSMTSI MRF shall be capable to ignore any "a=rid" constraints parameters it does not understand and shall correctly negotiate them away in the SDP answer, as specified in [155].

NOTE 1: The only, currently defined "a=rid" constraint applicable to 3GPP audio codecs is bitrate ("max-br").

The recommended approach by which the common and preferred codec information is exchanged between the MSMTSI MRF and the MSMTSI terminals is to use the order in which the codecs are listed in the SDP a=simulcast line, which lists simulcast streams in order of decreasing priority. The common codec should be listed first, assuming

that the common codec simulcast stream is to be used as far as possible, to avoid transcoding. The preferred codec may instead be listed first if a limited amount of transcoding is considered an acceptable cost for keeping good media quality.

NOTE 2: An MSMTSI MRF needs to make a trade-off between minimizing transcoding by use of a common codec and maximizing media quality by use of a preferred codec in the conference, which is described in more detail in clauses 6.17 and 6.13.4 in [152]. The common codec being listed with highest priority means that it will be kept, even if some other simulcast streams need to be rejected or dropped (e.g. due to network resource limitations, see clause S.8). The preferred codec being listed with highest priority similarly means that it will be kept, even if common codec simulcast streams need to be rejected or dropped, which can in turn require transcoding to some conference participants. This applies both across different simulcast streams (";" separator) and for alternatives ("," separator) within a single simulcast stream in the "a=simulcast" line. This choice between prioritizing common or preferred codecs does not impact interoperability and is therefore left for individual MSMTSI MRF implementation.

S.5.2 Main video

The main video SDP "m="-line shall be the first video "m="-line in an SDP offer from an MSMTSI client, to increase the probability that it is accepted by a non-MSMTSI client. The main video SDP "m="-line shall be identified by an "a=content:main" SDP attribute [81] under that "m="-line. The MSMTSI client shall be capable of identifying the main video when the video "m="-line contains "a=content:main", even if this is not the first video "m="-line in the SDP. An MSMTSI client shall be capable of receiving an SDP without any "a=content:main" SDP video attribute, and should then choose main video to be the first suitable video "m="-line that cannot be clearly identified for another purpose (such as for example "a=content:slides", see below).

In case of video, where the main video and thumbnail video are being sent from the MSMTSI client to the MSMTSI MRF, use of simulcast shall be indicated in SDP according to [154] and [155] in an SDP offer. SDP simulcast negotiation decides which simulcast formats, if any, that are sent between the MSMTSI clients in terminal and the MSMTSI MRF. An MSMTSI client in terminal shall use send direction simulcast in the SDP when negotiating use of simulcast to send main video thumbnail. An MSMTSI MRF shall use receive direction simulcast in the SDP when negotiating use of simulcast towards an MSMTSI client in terminal to receive main video thumbnail.

An MSMTSI client in terminal that receives an SDP offer using simulcast from a remote party that is not a conference focus (indicated by the SIP isFocus tag not being present in SIP headers), should disable use of simulcast in the corresponding SDP answer.

S.5.3 Thumbnail video

Each thumbnail video that the MSMTSI client supports shall be negotiated as a separate SDP video "m="-line, different from the main video "m="-line ("a=content:main") and any screenshare video "m="-line ("a=content:slides").

An MSMTSI client in the terminal shall use receive-only direction for all thumbnail videos in the SDP. An MSMTSI MRF shall use send-only direction for all thumbnail videos in the SDP that are sent towards an MSMTSI client in the terminal. As a specific case of the general SDP "m="-line handling specified by S.5, an MSMTSI client that receives an SDP offer with more thumbnail video "m="-lines than it can support, shall disable the "m="-lines it cannot support (by setting port to 0) in the SDP answer. Which thumbnail "m="-lines to keep and which to reject, in case all cannot be supported, is left for MSMTSI client implementation preference.

A thumbnail "m="-line is generally not dedicated to a certain conference participant and the number of "m="-lines for thumbnails therefore need not match the number of conference participants. If RTP stream selective forwarding (see clause S.6) is used for thumbnails by the MSMTSI MRF, though the forwarded participant may change at any point in time, a single thumbnail "m="-line can map to an RTP stream from any one of the participants. The number of thumbnail "m="-lines actually used between an individual MSMTSI client in terminal and an MSMTSI MRF is thus decided by the minimum number of thumbnail "m="-lines in any of their SDPs, irrespective of the MSMTSI client in terminal or the MSMTSI MRF being the entity supporting the fewest thumbnail "m="-lines. Therefore, the number of thumbnail "m="-lines supported by an individual MSMTSI client in terminal does not limit the number of thumbnail "m="-lines used between the MSMTSI MRF and other MSMTSI clients in terminals participating in the same conference.

An MSMTSI client in terminal that receives an SDP offer using thumbnails from a remote party that is not a conference focus (indicated by the SIP isFocus tag not being present in SIP headers), should disable thumbnail "m="-lines in the corresponding SDP answer.

S.5.4 Screenshare video

When screenshare video is supported, it shall be indicated as a separate SDP video "m="-line, identified by "a=content:slides" [81] under that "m="-line. There is no restriction in how the screenshare video "m="-line is ordered in relation to other "m="-lines in the SDP, except that it shall be listed after the main video "m="-line.

S.5.5 Audio

The main audio SDP "m="-line shall be the first "m="-line in an SDP offer from an MSMTSI client, to increase the probability that it is accepted by a non-MSMTSI client.

Support for multiple, simultaneous audio streams shall be indicated in SDP as separate audio "m="-lines. The number of supported channels in multi-channel audio shall be indicated per audio stream through the SDP "m="-line <encoding parameters>, with the default being a single channel when <encoding parameters> is omitted. There is no restriction in how additional audio "m="-lines are ordered in relation to other "m="-lines in the SDP, except that they shall be listed after the main audio "m="-line.

S.5.6 BFCP

Use of BFCP (see also clause S.7) is defined in TS 24.147 [147]. BFCP is negotiated with a single "m="-line for BFCP in SDP as specified in [150]. If both screenshare video and main video are negotiated, they shall be negotiated to use separate BFCP floor identifications. An MSMTSI client shall be capable of correctly associating SDP "m="-lines with BFCP floors through the SDP answer, as described in [150].

An MSMTSI MRF shall support at least the BFCP floor control server role in SDP offer/answer.

An MSMTSI client in terminal shall support the BFCP floor control client role in SDP offer/answer, but may in addition support also the BFCP floor control server role. Which role is taken by which part is decided by BFCP SDP offer/answer.

S.5.7 Compact Concurrent Codec Negotiation and Capabilities

S.5.7.1 General

To establish MMCMH sessions that make the most of all the participating terminals' codec capabilities without exceeding the concurrent codec capabilities of each participating terminal, the MMCMH session initiator needs to know the concurrent codec capabilities of all the other terminals. MSMTSI terminals can use the concurrent codec capabilities exchange procedures specified in this clause to obtain this information in a compact format. Furthermore, the specified procedures can be used to negotiate, in a compact manner, which concurrent codecs to use for the MMCMH session.

S.5.7.2 The Compact CCC SDP Attribute

The Compact CCC SDP attribute enables MSMTSI terminals to communicate the CCC information in a more compact format. The ABNF definition is as follows:

```
ccc-list    = "a=ccc_list:" codeclist 1*63( "|" ccc-prof )
codeclist  = codec [SP config] *63( ";" codec [SP config] )
ccc-prof   = "ENC:" num *63( rule num ) ":DEC:" num *63( rule num )
codec      = byte-string
           ; byte-string defined in RFC 4566
level     = 1*3DIGIT
profile    = 1*3DIGIT
config    = ( profile SP level ) / level
rule      = ";" / ","
num       = 1*2DIGIT
```

`codec` is the media subtype name of a codec as defined in the RTP payload format for that codec, e.g. "H264" for H.264 as defined in [25] and "H265" for H.265 as defined in [120], respectively.

`codeclist` is an ordered list of the different codecs that are supported by the terminal. The codecs should be listed in order of decreasing complexity from left to right for the "," rule to indicate the ability to substitute an instance of a more complex codec with a simpler one.

`level`, which is optional, specifies the level of the codec, converted and expressed in hexadecimal format, i.e., for H.264 and H.265 the value is equal to `level_idc` as defined in [25] and `level-id` as defined in [120], respectively.

`profile`, which is optional, specifies the profile of the codec, e.g. for H.264 and H.265 the value is equal to `profile_idc` as defined in [25] and `profile-id` as defined in [120], both expressed in hexadecimal format, respectively. If the profile is included then the level is also included. However, it is allowed to have the level to be present without the profile being present.

`rule` specifies in a particular CCC profile (`ccc-prof`) whether the resources used for a concurrent instance of the codec may be used instead by a concurrent instance of the codec listed afterwards, e.g., the next listed codec is less computationally complex and runs on the same processor as the previously listed codec. When the value is "," then an instance of the subsequent codec can be used in place of an instance of the previously listed codec. When the value is ";" an instance of the subsequent codec cannot be used instead.

`num` specifies the maximum number of supported concurrent encoders (when the combination follows "ENC") or decoders (when the combination follows "DEC") of the specified codec at the specified level and profile (when present). `num` specifies the maximum number of instances of the particular codec that can operate concurrently when all the other codecs in the same `ccc-prof` have their corresponding `num` of instances operating. The number of instances of `num` that immediately follows "ENC" and the number of instances of `num` that immediately follows "DEC" are both the same as the number of instances of `codec`, and an instance of `num` is mapped to an instance of `codec` with the same order in the ordered `codeclist`.

There can be multiple `ccc-prof`'s to indicate support of different configurations of concurrent encoders and decoders, e.g., to indicate that supporting less concurrent encoders enables the terminal to support more concurrent decoders. `ccc-prof`'s should not be in conflict with each other, i.e., indicate a different maximum number of instances (`num`) of a particular codec when the maximum number of instances of all the other codecs in the `ccc-prof`'s are the same. If a conflict is detected between `ccc-prof`'s, the information from the first `ccc-prof` among the conflicting `ccc-prof`'s is used and all the other conflicting `ccc-prof`'s are ignored.

If the terminal has the ability to trim the number of received media streams it actually decodes, it can advertise a `num` value that is larger than it actually has the concurrent decoding capabilities for.

As the "a=ccc_list" attribute is a compact representation of the media capabilities that are expressed in SDP, using it is not expected to introduce any additional security conditions beyond those identified for SDP. Therefore the security considerations for the attribute are covered by the security considerations for SDP as specified in RFC 4566 [8]. There are no identified interoperability concerns for this attribute as it is a new attribute and the values used for the `codec` field are well defined to use values managed by IANA.

S.5.7.3 Using the Compact CCC SDP Attribute for CCC Exchange

The SIP OPTIONS method specified in RFC 3261 is used to query the capabilities of another terminal by asking the conference participant to send a copy of the SDP it would offer.

For example, the SIP OPTIONS request may be made in-advance of a conference call and the SIP OPTIONS response be stored for the queried terminal. Alternatively, immediately before setting up a conference, the conference initiator may query the capabilities of all the terminals it plans to invite for which it does not have the information pre-stored.

In single source multi-unicast (SSMU) topology [152], the MRF sends the SIP OPTIONS request to each of the terminals before setting up the conference. The terminals send the SIP OPTIONS response to the MRF. The MRF can then use this response information to pre-configure and send the SDP Offers to the terminals using simulcast and multiple m- lines for the MSMTSI session. Alternatively, for the case where MSMTSI terminals call in and send SDP Offers to MRF (instead of MRF calling out), the MRF can still use knowledge from SIP OPTIONS response to adjust its SDP Answer.

In multi-unicast topology, the conference initiator sends the SIP OPTIONS request to each of the conference participants well in-advance of the conference call. Upon receiving the SIP OPTIONS response from the participants, the conference initiator may store the encoding/decoding preferences of the conference participants for setting up the MSMTSI session.

A new content type `cccex`, is used to request the `ccc_list` attribute (see S.5.7.4). The SIP OPTIONS response shall contain the text that follows "a=ccc_list:" when using the `ccc_list` attribute.

SIP OPTIONS SDP examples are given in Clause T.3.4.

S.5.7.4 Using the Compact CCC SDP Attribute for Session Initiation

S.5.7.4.1 General

MSMTSI terminals shall be able to interpret and respond to the `ccc_list` attribute included in an SDP answer. MSMTSI terminals should include the `ccc_list` attribute in an SDP offer to reduce the size of the offer when multiple media configurations of concurrent codecs are being offered.

S.5.7.4.2 SDP Offer Rules

When an MSMTSI terminal generates an offer that includes the `ccc_list` attribute, it shall include only a single instance of the `ccc_list` attribute, and one or more media configurations which together cover all the possible concurrent codec configurations that the offerer can support. If the included media configuration(s) offer more concurrent codec configurations than the offerer can support, the offering MSMTSI terminal shall use the `ccc_list` attribute to indicate to the answerer additional restrictions to the media configurations included in the offer. This allows the MSMTSI offerer to include a single media configuration without using MediaCapNeg and requiring much fewer lines of SDP.

When an MSMTSI terminal includes the `ccc_list` attribute in an SDP offer it shall check whether the `ccc_list` attribute is included in the SDP answer it receives. If the attribute is included, the offering MSMTSI terminal may store this CCC information about the other terminal for use in future sessions. If multiple instances of the attribute are included in the SDP answer, the offering MSMTSI terminal shall ignore all but the first instance. If the attribute is not included, the offering MSMTSI terminal shall check that it can support the media configuration selected in the SDP answer in the MMCMH session. If the offering MSMTSI can not support the selected media configuration, it shall send a re-INVITE to the answering terminal without including the `ccc_list` attribute in the new SDP offer.

S.5.7.4.3 SDP Answer Rules

When an answering MSMTSI terminal receives an offer that includes the `ccc_list` attribute, it shall include exactly one instance of the `ccc_list` attribute in the SDP answer. If multiple instances of the attribute are received in the offer, the answering MSMTSI terminal shall ignore all but the first instance. The answering MSMTSI terminal may store this CCC information about the offering terminal for use in future sessions.

When an answering MSMTSI terminal includes the `ccc_list` attribute in an SDP answer, it shall set the attribute to describe the complete concurrent codec capabilities of the answering MSMTSI terminal, independent of what was received in the SDP offer.

S.6 Media transport

S.6.1 RTP

An MSMTSI client shall be capable of receiving multiple, separate RTP streams [9] related to a single SIP dialog and route them to the correct decoder based on RTP SSRC, RTP Payload Type, and any "a=content" information in the SDP. The number of RTP streams used in each direction and for each media type is limited to what is negotiated by SDP offer/answer for that SIP dialog, allowing the involved MSMTSI clients to express a limit to the number of streams they can handle simultaneously.

An MSMTSI client in terminal shall be capable of relating received RTP SSRC and CSRC with information from the conference event information (clause S.4), to enable indicating the identity of the RTP stream source. If a received RTP SSRC has no match in the conference event information, and if RTP CSRC is present, CSRC shall also be matched against conference event information. It is left up to MSMTSI client implementation what RTP stream source identification to use if neither SSRC nor CSRC can be matched against conference event information.

S.6.2 RTCP

An MSMTSI client in the terminal shall, as an MTSI client in the terminal, be capable to receive RTCP feedback (FIR, PLI, TMMBR, etc) [40][43] and respond accordingly, but shall also be capable to identify which sent RTP stream (SSRC) the RTCP feedback targets and direct it to the appropriate encoder.

An MSMTSI client shall support RTP-level pause and resume functionality according to [156] for all of its RTP streams, with the exception of the main audio stream from an MSMTSI client in terminal towards an MSMTSI MRF, which is not required to (but may) support RTP-level pause and resume functionality. An MSMTSI client shall be capable of restricting its use of RTP-level pause and resume functionality according to received pause/resume config parameter information in SDP, including not using it at all if that is the negotiation outcome.

An MSMTSI client in terminal shall support RTP-level pause and resume functionality at least corresponding to config=3, and should support full RTP-level pause and resume functionality corresponding to config=1.

An MSMTSI MRF shall support RTP-level pause and resume functionality at least corresponding to config=2, and should support full RTP-level pause and resume functionality corresponding to config=1.

S.6.3 RTP Stream Selective Forwarding

When a conference includes a large number of terminals or participants, the size of the SDP Offer listing the concurrent codec capabilities (CCC) using multiple m= lines can increase considerably. To reduce the size of the SDP offer when a conference includes a large number of participants, RTP-level selective forwarding by the MSMTSI MRF shall support RTP pause/suspend, reuse, replace, and resume actions. This RTP-level "suspend, resume, replace, and resume" action is described using an example in Clause 6.13.4.2, TR 26.980 [152].

In the SDP Offer the MSMTSI MRF may use fewer downlink m= lines, corresponding to dynamically selected streams even though there are large number of participants or terminals in the MSMTSI session.

To support RTP stream selective forwarding when some participants in a conference supports both the "common" and one or more "preferred" codecs while other participants support only the "common" codec, certain conditions need to be fulfilled for MSMTSI clients both in the terminal and in MSMTSI MRF. This is described below.

The MSMTSI MRF should construct the SDP Offer such that the downlink RTP stream from one participant fits sufficiently well with the capabilities of another participant such that the same downlink m= line can be used for RTP stream selective forwarding. For this, the MSMTSI MRF should pre-analyze the CCCEX (e.g., based on SIP OPTIONS) and construct sub-groups of participants that can share a given m= line with a given set of "preferred" and "common" codecs in the simulcast.

An MSMTSI client in terminal that supports both "common" and "preferred" codecs for a media type shall support concurrent encoding, sending simulcast [155], and concurrent decoding, with more than a single one codec for every sending media source ("m=" line) of that media type. An MSMTSI MRF that supports both "common" and "preferred" codecs for a media type shall support receiving simulcast with more than a single codec for every received media source ("m=" line) of that media type.

An MSMTSI client in terminal that supports both "common" and "preferred" codecs for a media type shall be capable to receive media using any one of those codecs, even if the used codec (RTP payload type) is changing from one RTP packet to the next, for all received media sources ("m=" lines) of that media type. This capability to receive RTP payload type changing from one RTP packet to the next is indicated by accepting payload types with both "common" and "preferred" codecs from the SDP offer in the SDP answer. This way, no SDP re-negotiation is needed to change codec between the ones included in the answer. This ability to support RTP payload type changes every RTP packet is an essential part of the receiving end of a codec simulcast scenario, and shall be explicitly indicated in SDP by including "a=simulcast" with the supported codecs as simulcast format alternatives, even if it means that only a single simulcast stream with alternatives is listed in simulcast receive direction.

An MSMTSI client in the terminal that negotiated receiving simulcast with different codecs and that receives a set of such simulcast streams, can dynamically choose which simulcast stream to decode and should use the best quality simulcast stream that is available.

An MSMTSI MRF that negotiated sending simulcast with a set of different codecs and that needs to forward streams using those codecs, may change RTP payload type within that set in its sent stream at any point in time, as needed to support the communication scenario and forwarding strategy used by the MSMTSI MRF.

The MSMTSI MRF should take action to allow decoder memories to clear when replacing one RTP stream with another RTP stream during switching, to avoid media distortion in the receiving MSMTSI client in the terminal. To enable this, it is currently necessary for the MSMTSI MRF to know the RTP payload format of the used codecs, even if no transcoding is needed. The MSMTSI MRF should therefore, before suspending forwarding of RTP stream A and replacing it by instead forwarding RTP stream B:

- For speech
 - Replace the content of a few RTP packet payloads from RTP stream A with a few silence indication (SID) or discontinuous transmission (DTX) frames for the used codec
- For video
 - Take assistance from the sender of RTP stream B, asking for a decoder refresh point by sending RTCP FIR to that RTP stream B sender
 - Continue forwarding RTP stream A while monitoring RTP stream B for decoder refresh points
 - From the point where a decoder refresh point is received in RTP stream A, suspend forwarding of RTP stream A and replace it by forwarding RTP stream B

S.7 BFCP

S.7.1 General

BFCP with TCP transport according to [149][150] shall be supported. An MSMTSI client in terminal shall be capable of indicating to the user when it is granted a floor, when a floor request is rejected, and when a floor grant is revoked. The details of such indication are left for MSMTSI client implementation. If the MSMTSI client in terminal supports floor control of more than a single video, it shall be able to make such indications separately for each supported floor.

An MSMTSI client may support functionality for moderated BFCP floor handling, involving a floor chair.

An MSMTSI MRF should silently ignore and discard any received video that is currently under active floor control, but where another MSMTSI or MTSI client than the one sending such received stream currently owns the floor.

S.7.2 Floor controlled main video

An MSMTSI client shall support BFCP [147][149] to control the main video.

When BFCP is used to control the MSMTSI client main video, all MSMTSI clients are allowed to send video as long as no one owns that BFCP floor. Whenever any MSMTSI client owns the main video floor, only that MSMTSI client is allowed to send main video in the group video call, and the MSMTSI MRF forwards that video to all other participants. The MSMTSI MRF should continue to use any previously used video forwarding strategy towards the MSMTSI client in the terminal that owns the main video floor.

S.7.3 Floor controlled screenshare video

If screenshare video is supported by an MSMTSI client, BFCP [147][149] to control the screenshare video shall also be supported.

If BFCP is not available to control the screenshare video, and considering that the MSMTSI client supports screenshare video as a separate video "m="-line, implicit screenshare floor control of that "m="-line shall be assumed. Such implicit floor control shall here be taken to mean that the MSMTSI client is allowed to start sending screenshare video at any point in time, whenever initiated by the user of the MSMTSI client. No explicit screenshare floor grant indication to the sending MSMTSI client is possible in this case. An MSMTSI client in the terminal using implicitly floor controlled screenshare video that begins receiving screenshare video after itself started sending, shall immediately stop sending screenshare video, since this shall be interpreted as the implicit screenshare floor grant being revoked.

S.7.4 Implicit floor control for audio

An MSMTSI client in terminal is allowed to receive more audio streams than it is capable of decoding concurrently at a given time. In such case, the MSMTSI client in terminal shall have means for choosing which streams to prioritize and which ones to ignore. This selection can be made based on which streams are not in DTX mode. Media streams may also be prioritized based on the active level or volume of the audio stream. However, this requires decoding of the media from each stream to determine the loudest stream. The prioritized streams may further be spatially mixed for rendering.

S.7.5 Floor control interworking with DTMF-capable MTSI clients

If MTSI clients participate in the group video call, not supporting BFCP for the main video but having successfully negotiated use of DTMF with the MSMTSI MRF according to Annex G, the MSMTSI MRF may act as a signalling gateway between DTMF and BFCP main video floor control. It is then assumed that the MSMTSI MRF is configured with a DTMF sequence that can be used to request and release that BFCP floor, and that this DTMF sequence is somehow communicated to the MTSI client, but the details of that are out of scope for this specification.

If such MTSI client uses the designated DTMF sequence to request the main video floor, it shall be treated in the MSMTSI MRF as equivalent to receiving a BFCP request for the main video floor.

Lacking other possibilities to indicate main video floor grant status back to such DTMF-requesting MTSI client, the MSMTSI MRF should send its main video back to it as long as it owns the main video floor.

When the MTSI client owning the main video floor uses the DTMF sequence to release the main video floor, it shall be treated in the MSMTSI MRF as equivalent to receiving a BFCP release for the main video floor.

S.8 Rate Adaptation

An MSMTSI client in the terminal should know which RTP streams that share a common channel resource (such as a radio bearer), and shall take this into account when performing per-stream rate adaptation. Per-stream adaptation is specified in clause 10. An MSMTSI MRF should, unless explicitly configured otherwise, assume during rate adaptation that all media streams of the same SDP media type (audio, video, etc) from a single MSMTSI client share a common channel resource. The sum of bitrates used for streams sharing a common channel resource shall be controlled such that they jointly do not exceed the estimated available channel resource. This applies to all MSMTSI clients both in the send direction and in the receive direction, providing rate adaptation information feedback (e.g. TMMBR or any rate control mechanism natively available to the used codec) to the sending party.

In situations where a MSMTSI client or MRF media sender that has the ability to send multiple media streams is faced with scarce network resources, insufficient to support maximum bitrate for all streams the sender is capable to send, it can be necessary for the sender to choose which streams to either entirely omit, temporarily stop, or degrade more than others. Degrading all streams evenly may not be the best approach, but consistently degrading one stream at a time may also not be preferable. The individual media receiver of those multiple streams is likely the one best equipped to decide which streams are most interesting to receive in a specific communication scenario, but some general guidance can be given. It is recommended that the main audio stream is considered most important, except for special use cases that do not require use of audio. Screenshare video is likely next most important, after which comes main video and BFCP. When using simulcast, main video simulcast using different codecs is likely more important than video thumbnail simulcast. Video thumbnails are likely least important in most use cases.

Annex T (informative): SDP examples for Multi-party Multimedia Conference Media Handling

T.1 General

T.1.1 Introduction

This annex gives a few examples of media portions from possible SDP offers and answers involving an MSMTSI client and/or MSMTSI MRF. It is not feasible to cover all possible variants of different communication scenarios and MSMTSI capabilities and hence these examples should be regarded as just a few examples of many possible alternatives. For brevity, these examples do not make use of all functionality (like robustness etc) or list all codecs that could be used in a real SDP offer/answer, and for the same reason also limits the number of supported streams compared to what could be used in a real SDP offer/answer.

T.1.2 Quality of Service examples

This clause describes how the QoS bandwidth is reserved for MMCMH sessions based on the codec and bandwidth information in the SDP answer, the number of conference participants, and the topology of the multi-party session.

When determining the QoS bandwidth to reserve for the MMCMH session, the SDP answer is examined for the following:

- When simulcast is included, the bandwidth reserved for the transmission of the media (on either the uplink or downlink) is enough to cover the simulcast. If multiple media formats are sent concurrently then the reserved bandwidth is enough to carry the sum of their bandwidth requirements. If media formats are sent alternatively, then the reserved bandwidth is enough to carry the bandwidth requirements of the highest codec rate that can be used.
- If multiple active media lines are included, the reserved bandwidth is enough to carry the sum of the bandwidth requirements of each active media line.

Table T.0 provides examples of the QoS bandwidth that could be reserved for the example SDP answers listed in the rest of this Annex.

Table T.0: Example QoS bandwidth reservations for example SDP answers

| Table in Annex which has the example SDP answer | Uplink GBR (kbps) | Uplink MBR (kbps) | Dowlink GBR (kbps) | Downlink MBR (kbps) | Comments |
|---|-------------------|-------------------|--------------------|---------------------|--|
| T.2 | | 452.5 | | 452.5 | Assumed PCC chose symmetric QoS for DL |
| T.3 | | 2105 | | 2590 | Assumed PCC chose symmetric QoS for DL |
| T.4 | | 1865 | | 1865 | Estimating QoS for MSMTSI terminal's links, assumed PCC chose symmetric QoS for DL |
| T.6 | 118 | 118 | 133.5 | 133.5 | Can not assume symmetric because simulcast is on uplink. Assuming GBR= MBR for speech, but can set GBR lower if PCRF looks at codec-specific parameters and determines the lowest operating rate of the codecs |
| T.9 | 118 | 118 | 194 | 194 | Can not assume symmetric because simulcast is on uplink. Assuming GBR= MBR for speech, but can set GBR lower if PCRF looks at codec-specific parameters and determines the lowest operating rate of the codecs |
| T.10 | 118 | 118 | 133.5 | 133.5 | Can not assume symmetric because simulcast is on uplink. Assuming GBR= MBR for speech, but can set GBR lower if PCRF looks at codec-specific parameters and determines the lowest operating rate of the codecs |

T.2 MSMTSI video offer/answer examples

T.2.1 MSMTSI offer/answer towards an MTSI client

This offer includes sending two different simulcast streams for the main video, receiving two thumbnail videos, both sending and receiving screenshare video, and has support for BFCP to control screenshare and (possibly) main video, which are all features that can be supported by MSMTSI but that are not supported by a regular MTSI client. All audio is omitted in this example, for brevity, but could be added according to the other examples (e.g., in Clause T.3) in this annex.

Video levels are in this example aligned with the maximum size of the video stream, and the maximum receive bandwidth limit is set by the "b="-line rather than just implicitly by the video level bandwidth limit.

The main video is listed as the first video "m="-line and is also explicitly identified through "a=content:main".

One subsequent "m="-line is explicitly identified as screenshare video, using "a=content:slides".

The rest of the video "m="-lines indicate support for two additional, receive-only video thumbnails.

All video "m="-lines offer support for RTP level pause/resume, indicated through "a=rtcp-fb:* ccm pause". The "nowait" parameter is set, indicating that the MSMTSI client expects the RTP media streams to be sent point-to-point on RTP level. That can for example be either between MSMTSI clients in terminal, or between an MSMTSI client in terminal and an MSMTSI MRF. In either case, the "nowait" parameter indicates it is not expected that multiple receivers of the RTP streams are able to send RTCP back to request RTP level pause/resume.

Support for reduced-size RTCP (see clause 7.3.6 and [87]) is offered for all video "m="-lines, to save bandwidth for RTCP feedback messages listed on "a=rtcp-fb"-lines.

BFCP support is offered with client role.

Blank lines are here added in the SDP for improved readability, but are not included in an actual SDP. SDP lines specifically interesting to this example are highlighted in **bold**, which would also not be the case in an actual SDP.

Table T.1: Example SDP offer from MSMTSI towards MTSI

| SDP offer |
|---|
| m=video 49154 RTP/AVPF 101 102 b=AS:1060 b=RS:0 |

```
b=RR:2500
a=rtpmap:101 H264/90000
a=rtpmap:102 H264/90000
a=fmtp:101 packetization-mode=0; profile-level-id=42e01f; \
  sprop-parameter-sets=Z0KADZWgUH6Af1A=,aM46gA==
a=fmtp:102 packetization-mode=0; profile-level-id=42e00c; \
  sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA==
a=imageattr:101 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] [x=320,y=240] \
  recv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360] [x=320,y=240]
a=imageattr:102 send [x=176,y=144] [x=224,y=176] [x=320,y=180] \
  recv [x=176,y=144] [x=224,y=176] [x=320,y=180,q=0.6]
a=rid:0 send pt=101
a=rid:1 send pt=102
a=rid:2 recv pt=101
a=simulcast:send 0;1 recv 2
a=content:main
a=rtcp-rsize
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=rtcp-fb:* ccm pause nowait
a=extmap:4 urn:3gpp:video-orientation

m=video 49156 RTP/AVPF 103
b=AS:800
b=RS:0
b=RR:2500
a=rtpmap:103 H264/90000
a=fmtp:103 packetization-mode=0; profile-level-id=42e01f; \
  sprop-parameter-sets=Z0KADZWgUH6Af1A=,aM46gA==
a=imageattr:103 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] \
  recv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360]
a=content:slides
a=rtcp-rsize
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=rtcp-fb:* ccm pause nowait
a=extmap:4 urn:3gpp:video-orientation

m=video 49158 RTP/AVPF 104
b=AS:240
b=RS:0
b=RR:2500
a=rtpmap:104 H264/90000
a=fmtp:104 packetization-mode=0; profile-level-id=42e00c
a=imageattr:104 recv [x=176,y=144] [x=224,y=176] [x=320,y=180,q=0.6]
a=recvonly
a=rtcp-rsize
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=rtcp-fb:* ccm pause nowait
a=extmap:4 urn:3gpp:video-orientation

m=video 49160 RTP/AVPF 105
b=AS:240
b=RS:0
b=RR:2500
a=rtpmap:105 H264/90000
a=fmtp:105 packetization-mode=0; profile-level-id=42e00c
a=imageattr:105 recv [x=176,y=144] [x=224,y=176] [x=320,y=180,q=0.6]
a=recvonly
a=rtcp-rsize
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
```

```

a=rtcp-fb:* ccm tmmbr
a=rtcp-fb:* ccm pause nowait
a=extmap:4 urn:3gpp:video-orientation

m=application 50324 TCP/BFCP *
a=floorctrl:c-only
a=setup:actpass
a=connection:new

```

Table T.2: Example SDP answer from MTSI towards MSMTSI

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 101 b=AS:450 b=RS:0 b=RR:2500 a=rtpmap:101 H264/90000 a=fmtp:101 packetization-mode=0; profile-level-id=42e00d; \ sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA==; a=imageattr:101 send [x=320,y=240] rcv [x=320,y=240] a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=extmap:4 urn:3gpp:video-orientation m=video 0 RTP/AVPF 103 m=video 0 RTP/AVPF 104 m=video 0 RTP/AVPF 105 m=application 0 TCP/BFCP * </pre> |

The answerer, being an MTSI client, knows of no MSMTSI features, but has correctly disabled all of them in the SDP answer, according to generic SDP offer/answer rules, keeping unsupported "m="-lines with port set to zero, leaving the session effectively identical to a regular MTSI session with a single video and mono audio.

T.2.2 MSMTSI answer from an MSMTSI MRF

This example assumes the same offer as in Table T.1, which is thus not repeated here, but the answerer is here an MSMTSI MRF, supporting all of the offered MSMTSI-specific features. All audio is omitted in this example, for brevity, but could be added according to any of the other examples in this annex.

Note that the "isFocus" tag, identifying this answer as an MRF, is included as part of SIP headers and is thus not visible in the SDP in this example.

The MSMTSI MRF accepts to receive the two offered simulcast streams for the main video. Then, the MSMTSI MRF can send video for the two supported thumbnails, and can also control both main video and screenshare video through BFCP.

It can be noted that the MSMTSI MRF needs to keep all payload type formats that it accepts to use for simulcast on the "m="-line in the answer.

The MSMTSI MRF is, by including the RTP-level "nowait" parameter on the "a=rtcp-fb:* ccm pause" line in the SDP answer, confirming that exchange of RTP level pause/resume messages will be point-to-point and no hold-off period will therefore be necessary when resuming paused streams (see [156]).

Support for reduced-size RTCP (see clause 7.3.6 and [87]) is accepted for all video "m="-lines, to save bandwidth for RTCP feedback messages listed on "a=rtcp-fb"-lines.

The "a=label" lines are added by the MSMTSI MRF to support identification of BFCP-controlled "m="-lines, relating BFCP floor identifications to "m="-lines through "a=floorid" lines under the BFCP "m="-line. In this example, both the main video and the screenshare video "m="-lines are floor controlled. The thumbnail videos are however not floor controlled, so there are no "a=label" lines for those "m="-lines.

Blank lines are here added in the SDP for improved readability, but are not included in an actual SDP. SDP lines specifically interesting to this example are highlighted in **bold**, which would also not be the case in an actual SDP.

Table T.3: Example SDP answer from MSMTSI MRF towards MSMTSI

| SDP answer |
|--|
| <pre> m=video 49154 RTP/AVPF 101 102 b=AS:1300 b=RS:0 b=RR:2500 a=rtpmap:101 H264/90000 a=rtpmap:102 H264/90000 a=fmtp:101 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KADZWgUH6Af1A=,aM46gA== a=fmtp:102 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=imageattr:101 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] [x=320,y=240] \ recv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360] [x=320,y=240] a=imageattr:102 send [x=176,y=144] [x=224,y=176] [x=320,y=180] \ recv [x=176,y=144] [x=224,y=176] [x=320,y=180,q=0.6] a=rid:0 recv pt=101 a=rid:1 recv pt=102 a=rid:2 send pt=101 a=simulcast:recv 0;1 send 2 a=content:main a=label:m a=rtcp-rsize a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=rtcp-fb:* ccm pause nowait a=extmap:4 urn:3gpp:video-orientation m=video 49156 RTP/AVPF 103 b=AS:800 b=RS:0 b=RR:2500 a=rtpmap:103 H264/90000 a=fmtp:103 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KADZWgUH6Af1A=,aM46gA== a=imageattr:103 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] \ recv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360] a=content:slides a=label:s a=rtcp-rsize a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=rtcp-fb:* ccm pause nowait a=extmap:4 urn:3gpp:video-orientation m=video 49158 RTP/AVPF 104 b=AS:240 b=RS:0 b=RR:2500 a=rtpmap:104 H264/90000 a=fmtp:104 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=imageattr:104 send [x=176,y=144] [x=224,y=176] [x=320,y=180,q=0.6] a=sendonly a=rtcp-rsize a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=rtcp-fb:* ccm pause nowait a=extmap:4 urn:3gpp:video-orientation m=video 49160 RTP/AVPF 105 b=AS:240 b=RS:0 b=RR:2500 </pre> |


```

a=rtpmap:105 H264/90000
a=fmtp:105 packetization-mode=0; profile-level-id=42e00c; \
    sprop-parameter-sets=J0LgDJWgUH6AflA=,KM46gA==
a=imageattr:105 send [x=176,y=144] [x=224,y=176] [x=320,y=180,q=0.6]
a=sendonly
a=rtcp-rsize
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=rtcp-fb:* ccm pause nowait
a=extmap:4 urn:3gpp:video-orientation

m=application 50324 TCP/BFCP *
a=floorctrl:s-only
a=floorid:1 mstrm:m
a=floorid:2 mstrm:s
a=confid:23984
a=userid:48249
a=setup:active
a=connection:new

```

T.2.3 MSMTSI answer from an MSMTSI client in terminal

This example assumes the same offer as in clause T.2, which is thus not repeated here, but the answerer is here an MSMTSI client in terminal, supporting all of the offered MSMTSI-specific features, although not all of them are feasible to use between MSMTSI clients in terminal. All audio is omitted in this example, for brevity, but could be added according to any of the other examples in this annex.

Note that the "isFocus" tag is here not included as part of the SIP headers (compare to annex T.3). This information is used by the answering MSMTSI client to construct an appropriate SDP answer.

The answering MSMTSI client has no reason to send any thumbnail videos to another MSMTSI client in terminal, and has thus disabled them. There is no need for simulcast, meaning that the simulcast attribute and the corresponding "a=rid" attributes are removed from the SDP and simulcast will not be used. It can however act as BFCP server and can also support simultaneous main video and screen sharing, which are both kept.

Blank lines are here added in the SDP for improved readability, but are not included in an actual SDP. SDP lines specifically interesting to this example are highlighted in **bold**, which would also not be the case in an actual SDP.

Table T.4: Example SDP answer from MSMTSI towards another MSMTSI

| SDP answer |
|---|
| <pre> m=video 49154 RTP/AVPF 101 b=AS:1060 b=RS:0 b=RR:2500 a=rtpmap:101 H264/90000 a=fmtp:101 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KADZWgUH6AflA=,aM46gA== a=imageattr:101 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] [x=320,y=240] \ rcv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360] [x=320,y=240] a=content:main a=rtcp-rsize a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=rtcp-fb:* ccm pause nowait a=extmap:4 urn:3gpp:video-orientation m=video 49156 RTP/AVPF 103 b=AS:800 b=RS:0 b=RR:2500 a=rtpmap:103 H264/90000 a=fmtp:103 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KADZWgUH6AflA=,aM46gA== a=imageattr:103 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] \ rcv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360] </pre> |

```

a=content:slides
a=label:10
a=rtcp-rsize
a=rtcp-fb:* trr-int 5000
a=rtcp-fb:* nack
a=rtcp-fb:* nack pli
a=rtcp-fb:* ccm fir
a=rtcp-fb:* ccm tmmbr
a=rtcp-fb:* ccm pause nowait
a=extmap:4 urn:3gpp:video-orientation

m=video 0 RTP/AVPF 104

m=video 0 RTP/AVPF 105

m=application 50324 TCP/BFCP *
a=floorctrl:s-only
a=floorid:1 mstrm:10
a=confid:237
a=userid:278
a=setup:passive
a=connection:new

```

T.2.4 MSMTSI simulcast offer using a single payload type

This example offer is an excerpt of just the first "m=" block from the example in clause T.2.1 and effectively offers the same functionality, but does not make use of different payload types to distinguish the simulcast formats. Instead different "a=rid" constraints parameters are used. Because there is only a single format listed on the "m=" line containing the "a=simulcast" line, the "a=rid" lines contains constraints parameters (here max-width and max-height) to differentiate the simulcast formats.

SDP lines specifically interesting to this example are highlighted in **bold**, which would not be the case in an actual SDP.

Table T.4a: Example SDP simulcast offer from MSMTSI using a single payload type

| SDP offer |
|--|
| <pre> m=video 49154 RTP/AVPF 101 b=AS:1060 b=RS:0 b=RR:2500 a=rtptime:101 H264/90000 a=fmtp:101 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KADZWgUH6AflA=,aM46gA= a=imageattr:101 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] [x=320,y=240] \ [x=176,y=144] [x=224,y=176] [x=320,y=180] \ recv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360] [x=320,y=240] \ [x=176,y=144] [x=224,y=176] [x=320,y=180] a=rid:0 send pt=101 max-width=1280;max-height=720 a=rid:1 send pt=101 max-width=320;max-height=180 a=rid:2 recv pt=101 max-width=1280;max-height=720 a=simulcast:send 0;1 recv 2 a=content:main a=rtcp-rsize a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbr a=rtcp-fb:* ccm pause nowait a=extmap:4 urn:3gpp:video-orientation </pre> |

T.2.5 MSMTSI simulcast offer using two codecs

This example offer is an excerpt of just the first "m=" block from the example in clause T.2.1 and effectively offers the same functionality, but makes use of simulcast with different video codecs, H.264 and H.265, in addition to the simulcast used for main video thumbnail in clause T.2.1.

SDP lines specifically interesting to this example are highlighted in **bold**, which would not be the case in an actual SDP.

Table T.4b: Example SDP simulcast offer from MSMTSI using two codecs

| SDP offer |
|---|
| <pre> m=video 49154 RTP/AVPF 101 102 103 104 b=AS:1960 b=RS:0 b=RR:2500 a=rtpmap:101 H265/90000 a=rtpmap:102 H264/90000 a=rtpmap:103 H265/90000 a=rtpmap:104 H264/90000 a=fmtp:101 profile-id=1; level-id=93; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwBdLAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaAAAwBdoAKAgCOWUuS0i9AHcIBB; \ sprop-pps=RAHAcYDZIA== a=fmtp:103 profile-id=1; level-id=30; \ sprop-vps=QAEMAf//AWAAAAMAgAAAaAAAwA8LAUg; \ sprop-sps=QgEBAWAAAAMAgAAAaAAAwA8oAoIDxZS5LSL0AdwgEE=; \ sprop-pps=RAHAcYDZIA==; a=fmtp:102 packetization-mode=0; profile-level-id=42e01f; \ sprop-parameter-sets=Z0KADZWgUH6Af1A=,aM46gA== a=fmtp:104 packetization-mode=0; profile-level-id=42e00c; \ sprop-parameter-sets=J0LgDJWgUH6Af1A=,KM46gA== a=imageattr:101 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] [x=320,y=240] \ recv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360] [x=320,y=240] a=imageattr:103 send [x=176,y=144] [x=224,y=176] [x=320,y=180] \ recv [x=176,y=144] [x=224,y=176] [x=320,y=180,q=0.6] a=imageattr:102 send [x=1280,y=720] [x=848,y=480] [x=640,y=360] [x=320,y=240] \ recv [x=1280,y=720,q=0.6] [x=848,y=480] [x=640,y=360] [x=320,y=240] a=imageattr:104 send [x=176,y=144] [x=224,y=176] [x=320,y=180] \ recv [x=176,y=144] [x=224,y=176] [x=320,y=180,q=0.6] a=rid:0 send pt=101 a=rid:1 send pt=103 a=rid:2 recv pt=101 a=rid:4 send pt=102 a=rid:5 send pt=104 a=rid:6 recv pt=102 a=simulcast:send 0;4;1;5 recv 2,6 a=content:main a=rtcp-rsize a=rtcp-fb:* trr-int 5000 a=rtcp-fb:* nack a=rtcp-fb:* nack pli a=rtcp-fb:* ccm fir a=rtcp-fb:* ccm tmmbrr a=rtcp-fb:* ccm pause nowait a=extmap:4 urn:3gpp:video-orientation </pre> |

T.3 MSMTSI audio offer/answer examples

T.3.1 MSMTSI offer with multi-stream audio support

This offer includes multi-stream audio, sending simulcast with multiple codecs for the main audio, receiving two additional audio participants for local rendering, and receiving codec simulcast for all received audio streams. All video is omitted in this example, for brevity, but could be added according to any of the other examples in this annex.

Three different codecs are offered as audio simulcast in the send direction, separated by semicolons. In the simulcast receive direction, a single stream is offered, explicitly accepting three different alternative simulcast formats, separated by comma, which means that the used audio codec (RTP payload format) is allowed to change from one RTP packet to the next.

Blank lines are here added in the SDP for improved readability, but are not included in an actual SDP. SDP lines specifically interesting to this example are highlighted in **bold**, which would also not be the case in an actual SDP.

Table T.5: Example SDP offer from MSMTSI multi-stream audio

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 98 97 99 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:98 EVS/16000/1 a=fmtp:98 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 a=rid:0 send pt=97 a=rid:1 send pt=98 a=rid:2 send pt=99 a=rid:3 recv pt=97 a=rid:4 recv pt=98 a=rid:5 recv pt=99 a=simulcast:send 0;1;2 recv 3,4,5 m=audio 49154 RTP/AVP 101 102 103 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=recvonly a=rtpmap:101 EVS/16000/1 a=fmtp:101 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:102 AMR-WB/16000/1 a=fmtp:102 mode-change-capability=2; max-red=220 a=rtpmap:103 AMR/8000/1 a=fmtp:103 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 a=rid:6 recv pt=101 a=rid:7 recv pt=102 a=rid:8 recv pt=103 a=simulcast:recv 6,7,8 m=audio 49156 RTP/AVPF 104 105 106 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=recvonly a=rtpmap:104 EVS/16000/1 a=fmtp:104 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:105 AMR-WB/16000/1 a=fmtp:105 mode-change-capability=2; max-red=220 a=rtpmap:106 AMR/8000/1 a=fmtp:106 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 a=rid:9 recv pt=104 a=rid:10 recv pt=105 a=rid:11 recv pt=106 a=simulcast:recv 9,10,11 </pre> |

T.3.2 MSMTSI answer with multi-stream audio support

This answer builds on the offer in annex T.3.1, includes multi-stream audio, accepting to receive simulcast with multiple codecs for the main audio, sending two additional audio participants for local rendering, and sending codec simulcast for all received audio streams. All video is omitted in this example, for brevity, but could be added according to any of the other examples in this annex.

The three different codecs from the offer are accepted as audio simulcast in the receive direction, separated by semicolons. In the simulcast send direction, a single stream is accepted, explicitly listing three different alternative simulcast formats, separated by comma, which means that the used audio codec (RTP payload format) may change from one RTP packet to the next.

As in annex T.2.2, it can be noted that the MSMTSI MRF needs to keep all payload type formats that it accepts to use for simulcast on the "m"-line in the answer.

Blank lines are here added in the SDP for improved readability, but not included in an actual SDP. SDP lines specifically interesting to this example are highlighted in **bold**, which would also not be the case in an actual SDP.

Table T.6: Example SDP answer from MSMTSI MRF accepting multi-stream audio

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVPF 98 97 99 b=AS:113 a=acfg:1 t=1 a=rtpmap:98 EVS/16000/1 a=fmtp:98 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:99 AMR/8000/1 a=fmtp:99 mode-change-capability=2; max-red=220 aptime:20 amaxptime:240 a=rid:0 recv pt=97 a=rid:1 recv pt=98 a=rid:2 recv pt=99 a=rid:3 send pt=98 a=rid:4 send pt=97 a=rid:5 send pt=99 a=simulcast:recv 0;1;2 send 3,4,5 m=audio 49154 RTP/AVPF 101 102 103 b=AS:42 a=acfg:1 t=1 a=sendonly a=rtpmap:101 EVS/16000/1 a=fmtp:101 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:102 AMR-WB/16000/1 a=fmtp:102 mode-change-capability=2; max-red=220 a=rtpmap:103 AMR/8000/1 a=fmtp:103 mode-change-capability=2; max-red=220 aptime:20 amaxptime:240 a=rid:6 send pt=101 a=rid:7 send pt=102 a=rid:8 send pt=103 a=simulcast:send 6,7,8 m=audio 49156 RTP/AVPF 104 105 106 b=AS:42 a=acfg:1 t=1 a=sendonly a=rtpmap:104 EVS/16000/1 a=fmtp:104 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:105 AMR-WB/16000/1 a=fmtp:105 mode-change-capability=2; max-red=220 a=rtpmap:106 AMR/8000/1 a=fmtp:106 mode-change-capability=2; max-red=220 aptime:20 amaxptime:240 a=rid:9 send pt=104 a=rid:10 send pt=105 a=rid:11 send pt=106 a=simulcast:send 9,10,11 </pre> |

T.3.3 MSMTSI CCCEX SDP offer/answer example

Table T.7 shows example concurrent codec combinations supported at the terminal. All video is omitted in this example, for brevity, but could be added according to any of the other examples in this annex. As shown in Table T.7, the terminal may have identified three possible CCCEX combinations through profiles (shown for 6 participants). SDP offer examples from the terminal to the MRF for profile A is shown in Table T.8. The MSMTSI MRF may then send an SDP answer as shown using the simulcast attribute and multiple m-lines in Table T.9 enabling a multi-stream multiparty conference (among 6 participants).

Table T.7: Example concurrent codec capability configurations in MSMTSI terminals

| Number of participants | CCCEX combinations supported at the MSMTSI terminals A, B and C |
|------------------------|--|
| N = 6 | <p>A. [Encoder/send: AMR, AMR-WB, EVS] [Decoder/recv: 1 AMR, 1 AMR-WB, 3 EVS]</p> <p>B. [Encoder/send: AMR-WB, EVS] [Decoder/recv: 1 AMR-WB, 4 EVS]</p> <p>C. [Encoder/send: AMR, EVS] [Decoder/recv: 5 EVS]</p> |

Table T.8: Example SDP offer for CCCEX example configuration from MSMTSI terminal A

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 96 97 98 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:96 EVS/16000/1 a=fmtp:96 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 a=rid:0 send pt=96 a=rid:1 send pt=97 a=rid:2 send pt=98 a=rid:3 recv pt=96 a=rid:4 recv pt=97 a=rid:5 recv pt=98 a=simulcast:send 0;1;2 recv 3,4,5 m=audio 49154 RTP/AVP 101 102 103 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=recvonly a=rtpmap:101 EVS/16000/1 a=fmtp:101 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:102 AMR-WB/16000/1 a=fmtp:102 mode-change-capability=2; max-red=220 a=rtpmap:103 AMR/8000/1 a=fmtp:103 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 a=rid:6 recv pt=101 a=rid:7 recv pt=102 a=rid:8 recv pt=103 a=simulcast:recv 6,7,8 m=audio 49156 RTP/AVP 104 105 106 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=recvonly a=rtpmap:104 EVS/16000/1 a=fmtp:104 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:105 AMR-WB/16000/1 a=fmtp:105 mode-change-capability=2; max-red=220 a=rtpmap:106 AMR/8000/1 a=fmtp:106 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 a=rid:9 recv pt=104 a=rid:10 recv pt=105 a=rid:11 recv pt=106 a=simulcast:recv 9,10,11 m=audio 49158 RTP/AVP 107 108 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 </pre> |

```

a=recvonly
a=rtpmap:107 AMR-WB/16000/1
a=fmtp:107 mode-change-capability=2; max-red=220
a=rtpmap:108 AMR/8000/1
a=fmtp:108 mode-change-capability=2; max-red=220
a=ptime:20
a=maxptime:240
a=rid:12 recv pt=107
a=rid:13 recv pt=108
a=simulcast:recv 12,13

m=audio 49160 RTP/AVP 109
b=AS:29
a=tcap:1 RTP/AVPF
a=pcfg:1 t=1
a=recvonly
a=rtpmap:109 AMR/8000/1
a=fmtp:109 mode-change-capability=2; max-red=220
a=ptime:20
a=maxptime:240
a=rid:14 recv pt=109
a=simulcast:recv 14

```

Table T.9: Example SDP answer from MSMTSI MRF accepting multi-stream audio and enabling a conference with 6 participants (for SDP offer in Table T.8)

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVPF 96 97 98 b=AS:113 a=acfg:1 t=1 a=rtpmap:96 EVS/16000/1 a=fmtp:96 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; mode-set=0,2,4,7; max-red=220 a=ptime:20 a=maxptime:240 a=rid:0 recv pt=96 a=rid:1 recv pt=97 a=rid:2 recv pt=98 a=rid:3 send pt=96 a=rid:4 send pt=97 a=rid:5 send pt=98 a=simulcast: recv 0;1;2 send 3,4,5 m=audio 49154 RTP/AVPF 101 102 103 b=AS:42 a=acfg:1 t=1 a=sendonly a=rtpmap:101 EVS/16000/1 a=fmtp:101 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:102 AMR-WB/16000/1 a=fmtp:102 mode-change-capability=2; max-red=220 a=rtpmap:103 AMR/8000/1 a=fmtp:103 mode-change-capability=2; mode-set=0,2,4,7; max-red=220 a=ptime:20 a=maxptime:240 a=rid:6 send pt=101 a=rid:7 send pt=102 a=rid:8 send pt=103 a=simulcast:send 6,7,8 m=audio 49156 RTP/AVPF 105 106 b=AS:42 a=acfg:1 t=1 a=sendonly a=rtpmap:105 AMR-WB/16000/1 a=fmtp:105 mode-change-capability=2; max-red=220 a=rtpmap:106 AMR/8000/1 a=fmtp:106 mode-change-capability=2; mode-set=0,2,4,7; max-red=220 a=ptime:20 a=maxptime:240 a=rid:10 send pt=105 a=rid:11 send pt=106 </pre> |

```

a=simulcast:send 10,11

m=audio 49158 RTP/AVPF 108
b=AS:29
a=acfg:1 t=1
a=sendonly
a=rtpmap:108 AMR/8000/1
a=fmtp:108 mode-change-capability=2; mode-set=0,2,4,7; max-red=220
aptime:20
a=maxptime:240
a=rid:13 send pt=108
a=simulcast:send 13

m=audio 49160 RTP/AVPF 109
b=AS:29
a=acfg:1 t=1
a=sendonly
a=rtpmap:109 AMR/8000/1
a=fmtp:109 mode-change-capability=2; mode-set=0,2,4,7; max-red=220
aptime:20
a=maxptime:240
a=rid:14 send pt=109
a=simulcast:send 14

```

Table T.10 shows an example, where the MSMTSI MRF identifies that there are only four participants in the conference and it therefore disables the two "m="lines.

Table T.10: Example SDP answer from MSMTSI MRF accepting multi-stream audio and enabling a conference with 4 participants (for SDP offer in Table T.8)

| SDP answer |
|---|
| <pre> m=audio 49152 RTP/AVPF 96 97 98 b=AS:113 a=acfg:1 t=1 a=rtpmap:96 EVS/16000/1 a=fmtp:96 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; mode-set=0,2,4,7; max-red=220 aptime:20 a=maxptime:240 a=rid:0 recv pt=96 a=rid:1 recv pt=97 a=rid:2 recv pt=98 a=rid:3 send pt=96 a=rid:4 send pt=97 a=rid:5 send pt=98 a=simulcast:recv 0;1;2 send 3,4,5 m=audio 49154 RTP/AVPF 101 102 103 b=AS:42 a=acfg:1 t=1 a=sendonly a=rtpmap:101 EVS/16000/1 a=fmtp:101 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:102 AMR-WB/16000/1 a=fmtp:102 mode-change-capability=2; max-red=220 a=rtpmap:103 AMR/8000/1 a=fmtp:103 mode-change-capability=2; mode-set=0,2,4,7; max-red=220 aptime:20 a=maxptime:240 a=rid:6 send pt=101 a=rid:7 send pt=102 a=rid:8 send pt=103 a=simulcast:send 6,7,8 m=audio 49156 RTP/AVPF 105 106 b=AS:42 a=acfg:1 t=1 a=sendonly a=rtpmap:105 AMR-WB/16000/1 a=fmtp:105 mode-change-capability=2; max-red=220 a=rtpmap:106 AMR/8000/1 a=fmtp:106 mode-change-capability=2; mode-set=0,2,4,7; max-red=220 </pre> |


```

a=ptime:20
a=maxptime:240
a=rid:10 send pt=105
a=rid:11 send pt=106
a=simulcast:send 10,11

m=audio 0 RTP/AVP 108
m=audio 0 RTP/AVP 109

```

T.3.3a MSMTSI offer/answer examples using the compact CCC SDP attribute

Table T.3a1 shows an example SDP offer using the compact `ccc_list` SDP attribute to simultaneously describe the three media configurations listed for N=6 in Table T.7. Note that the media configuration for the codec is broader than all the concurrent codec configurations that can be supported by the offering MSMTSI terminal. Similar to table T.8, all video is omitted in this example for brevity, but could be added according to any of the other examples in this annex.

Table T.3a1: Example SDP offer from an MSMTSI terminal

| SDP offer |
|---|
| <pre> a = ccc_list:EVS;AMR-WB;AMR ENC:1;1;1:DEC:3,1,1 ENC:1;1;0:DEC:4,1,0 ENC:1;0;1:DEC:5,0,0 m=audio 49152 RTP/AVP 96 97 98 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:96 EVS/16000/1 a=fmtp:96 br=13.2-24.4; bw=nb-swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 a=rid:0 send pt=96 a=rid:1 send pt=97 a=rid:2 send pt=98 a=rid:3 recv pt=96 a=rid:4 recv pt=97 a=rid:5 recv pt=98 a=simulcast:send 0;1;2 recv 3,4,5 m=audio 49154 RTP/AVP 101 102 103 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=recvonly a=rtpmap:101 EVS/16000/1 a=fmtp:101 br=13.2-24.4; bw=nb-swb; max-red=220 a=rtpmap:102 AMR-WB/16000/1 a=fmtp:102 mode-change-capability=2; max-red=220 a=rtpmap:103 AMR/8000/1 a=fmtp:103 mode-change-capability=2; max-red=220 a=ptime:20 a=maxptime:240 a=rid:6 recv pt=101 a=rid:7 recv pt=102 a=rid:8 recv pt=103 a=simulcast:recv 6,7,8 m=audio 49156 RTP/AVP 104 105 106 b=AS:42 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=recvonly a=rtpmap:104 EVS/16000/1 a=fmtp:104 br=13.2-24.4; bw=nb-swb; max-red=220 a=rtpmap:105 AMR-WB/16000/1 a=fmtp:105 mode-change-capability=2; max-red=220 a=rtpmap:106 AMR/8000/1 a=fmtp:106 mode-change-capability=2; max-red=220 </pre> |

```

a=ptime:20
a=maxptime:240
a=rid:9 recv pt=104
a=rid:10 recv pt=105
a=rid:11 recv pt=106
a=simulcast:recv 9,10,11

m=audio 49158 RTP/AVP 107 108 109
b=AS:42
a=tcap:1 RTP/AVPF
a=pcfg:1 t=1
a=recvonly
a=rtpmap:107 EVS/16000/1
a=fmtp:107 br=13.2-24.4; bw=nb-swb; max-red=220
a=rtpmap:108 AMR-WB/16000/1
a=fmtp:108 mode-change-capability=2; max-red=220
a=rtpmap:109 AMR/8000/1
a=fmtp:109 mode-change-capability=2; max-red=220
a=ptime:20
a=maxptime:240
a=rid:12 recv pt=107
a=rid:13 recv pt=108
a=rid:14 recv pt=109
a=simulcast:recv 12,13,14

m=audio 49160 RTP/AVP 110,111,112
b=AS:42
a=tcap:1 RTP/AVPF
a=pcfg:1 t=1
a=recvonly
a=rtpmap:110 EVS/16000/1
a=fmtp:110 br=13.2-24.4; bw=nb-swb; max-red=220
a=rtpmap:111 AMR-WB/16000/1
a=fmtp:111 mode-change-capability=2; max-red=220
a=rtpmap:112 AMR/8000/1
a=fmtp:112 mode-change-capability=2; max-red=220
a=ptime:20
a=maxptime:240
a=rid:15 recv pt=110
a=rid:16 recv pt=111
a=rid:17 recv pt=112
a=simulcast:recv 15,16,17

```

An answering MSMTSI MRF that supports the `ccc_list` attribute may then send an SDP answer as shown Table T.x2, including the `ccc_list` attribute and using the `simulcast` attribute and multiple `m`-lines to enable a multi-stream multiparty conference among 6 participants. The included `ccc_list` attribute indicates the full capabilities of the MRF to concurrently switch multiple audio streams (10 EVS, 5 AMR-WB, 5 AMR-NB, etc...) towards the offering MSMTSI terminal eventhough the selected media configurations in the answer are only using a subset of the MRF's capabilities.

Table T.3a2: Example SDP answer from the MSMTSI MRF accepting multi-stream audio and enabling a conference with 6 participants (for SDP offer in Table T.3a1)

| SDP answer |
|---|
| <pre> a = ccc_list:EVS;AMR-WB;AMR ENC:10,5,5:DEC:1;1;1 m=audio 49152 RTP/AVPF 96 97 98 b=AS:113 a=acfg:1 t=1 a=rtpmap:96 EVS/16000/1 a=fmtp:96 br=13.2-24.4; bw=nb-swb; max-red=220 a=rtpmap:97 AMR-WB/16000/1 a=fmtp:97 mode-change-capability=2; max-red=220 a=rtpmap:98 AMR/8000/1 a=fmtp:98 mode-change-capability=2; mode-set=0,2,4,7; max-red=220 a=ptime:20 a=maxptime:240 a=rid:0 recv pt=96 a=rid:1 recv pt=97 a=rid:2 recv pt=98 a=rid:3 send pt=96 a=rid:4 send pt=97 a=rid:5 send pt=98 </pre> |

```

a=simulcast: recv 0;1;2 send 3,4,5

m=audio 49154 RTP/AVPF 101 102 103
b=AS:42
a=acfg:1 t=1
a=sendonly
a=rtpmap:101 EVS/16000/1
a=fmtp:101 br=13.2-24.4; bw=nb-swb; max-red=220
a=rtpmap:102 AMR-WB/16000/1
a=fmtp:102 mode-change-capability=2; max-red=220
a=rtpmap:103 AMR/8000/1
a=fmtp:103 mode-change-capability=2; mode-set=0,2,4,7; max-red=220
aptime:20
a=maxptime:240
a=rid:6 send pt=101
a=rid:7 send pt=102
a=rid:8 send pt=103
a=simulcast:send 6,7,8

m=audio 49156 RTP/AVPF 105 106
b=AS:42
a=acfg:1 t=1
a=sendonly
a=rtpmap:105 AMR-WB/16000/1
a=fmtp:105 mode-change-capability=2; max-red=220
a=rtpmap:106 AMR/8000/1
a=fmtp:106 mode-change-capability=2; mode-set=0,2,4,7; max-red=220
aptime:20
a=maxptime:240
a=rid:10 send pt=105
a=rid:11 send pt=106
a=simulcast:send 10,11

m=audio 49158 RTP/AVPF 108
b=AS:29
a=acfg:1 t=1
a=sendonly
a=rtpmap:108 AMR/8000/1
a=fmtp:108 mode-change-capability=2; mode-set=0,2,4,7; max-red=220
aptime:20
a=maxptime:240
a=rid:13 send pt=108
a=simulcast:send 13

m=audio 49160 RTP/AVPF 109
b=AS:29
a=acfg:1 t=1
a=sendonly
a=rtpmap:109 AMR/8000/1
a=fmtp:109 mode-change-capability=2; mode-set=0,2,4,7; max-red=220
aptime:20
a=maxptime:240
a=rid:14 send pt=109
a=simulcast:send 14

```

T.3.4 SIP OPTIONS request for multi-stream audio support

This offer includes multi-stream audio, sending simulcast with multiple codecs for the main audio, receiving two additional audio participants for local rendering, and receiving codec simulcast for all received audio streams. All video is omitted in this example, for brevity, but could be added according to any of the other examples in this annex.

Blank lines are here added in the SDP for improved readability, but are not included in an actual SDP. SDP lines specifically interesting to this example are highlighted in **bold**, which would also not be the case in an actual SDP.

Table T.11 shows an example SIP OPTIONS request from an MRF or a conference initiator. Table T.12 shows an example SIP OPTIONS response from a conference participant to the MRF or the initiator. The SIP OPTIONS response includes the SDP Offer of the conference participant. From Table T.12, the conference participant can allow for three concurrent encoding and three concurrent decoding of audio streams.

Table T.11: Example SIP OPTIONS request from an MRF or a conference initiator

| SIP OPTIONS request |
|---|
| <pre> OPTIONS sip:cccEx@mmcmh.com SIP/2.0 To: <sip:cccEx@mmcmh.com> From: P1 <sip:p1@msmtsi.com>;tag=TR26980 Call-ID: abcdefgh CSeq: 16384 OPTIONS Max-Forwards: 100 Via: SIP/2.0/UDP msmtsi.com; branch=z9hG4bKxxxxxx Contact: <sip:p1@msmtsi.com> Accept: application/cccx </pre> |

Table T.12: Example SIP OPTIONS response from a conference participant to the MRF or the initiator

| SIP OPTIONS response |
|---|
| <pre> SIP/2.0 200 OK Via: SIP/2.0/UDP msmtsi.com; branch= z9hG4bKxxxxxx; received=10.10.10.10 To: <sip:cccEx@mmcmh.com>;tag= TR26980E From: P1 <sip:p1@msmtsi.com>;tag=TR26980 Call-ID: abcdefgh CSeq: 16384 OPTIONS Allow: INVITE, ACK, CANCEL, OPTIONS, BYE Accept: application/cccx Content-Type: application/cccx a=ccc_list:EVS:AMR-WB:AMR ENC:1;1;1:DEC:3,1,1 </pre> |

Annex U (informative): IANA registration information for media types

U.1 Introduction

This Annex provides the media type registration information that is referenced from the IANA registry at <http://www.iana.org/>.

U.2 application/ccce

To: media-types@iana.org

Subject: Registration of media type "application/cccx"

Type name: application

Subtype name: ccce

Required parameters: None

Optional parameters: None

Encoding considerations:

ccce content is UTF-8 format text.

Security considerations:

See clause S.5.7.2 of TS 26.114

Interoperability considerations:

See clause S.5.7.2 of TS 26.114

Published specification:

See TS 26.114

Applications usage:

Voice over IP, video teleconferencing, streaming media, among others.

Additional information:

Magic number(s): None

File extension(s): The extension ".c3ex"

Macintosh File Type Code(s): "c3ex"

Intended usage: COMMON

Contact Person:

Contact Name: 3GPP Specifications Manager

Contact Email Address: 3gppContact@etsi.org

Author/Change Controller (for standards tree registrations, this is typically the standards body): 3GPP SA Working Group 4

Annex V (informative): Delay Adaptation and Example Uses of DBI Signaling

V.1 General

End-to-end delay and jitter performance plays an important role in determining MTSI quality of experience. MTSI sender and MTSI receiver may perform adaptation based on delay budget adjustments, including jitter buffer size adaptation at the receiver, air interface delay adjustments at both sender and receiver using RAN delay budget reporting as specified in TS 36.331 [160] for E-UTRA and TS 38.331 [163] for NR. MTSI sender and MTSI receiver may also exchange delay budget information (DBI) with each other as described in clause 7.3.8, to signal available delay budget as well as to request delay budget.

This clause provides various example informative signalling flows on delay adaptation using these mechanisms.

Figure V.2.1 presents a signaling flow example for RAN delay budget reporting usage for voice in MTSI without DBI signalling.

Figure V.2.2 and V.2.3 present signaling flow examples for RAN delay budget reporting usage in MTSI involving uni-directional DBI signalling with only indication of available delay budget from MTSI receiver to MTSI sender.

Figure V.2.4 presents a signaling flow example for RAN delay budget reporting usage in MTSI involving bi-directional DBI signalling with indication of available delay budget from MTSI receiver to MTSI sender and requested delay budget from MTSI sender to MTSI receiver.

Figure V.2.5 presents a signalling flow example on usage of RAN delay budget request in MTSI with bi-directional DBI signalling and with jitter buffer adjustment.

V.2 Example Signaling Flows on Delay Adaptation

In Figure V.2.1, a signaling flow for RAN delay budget reporting usage for voice in MTSI without DBI signalling is presented.

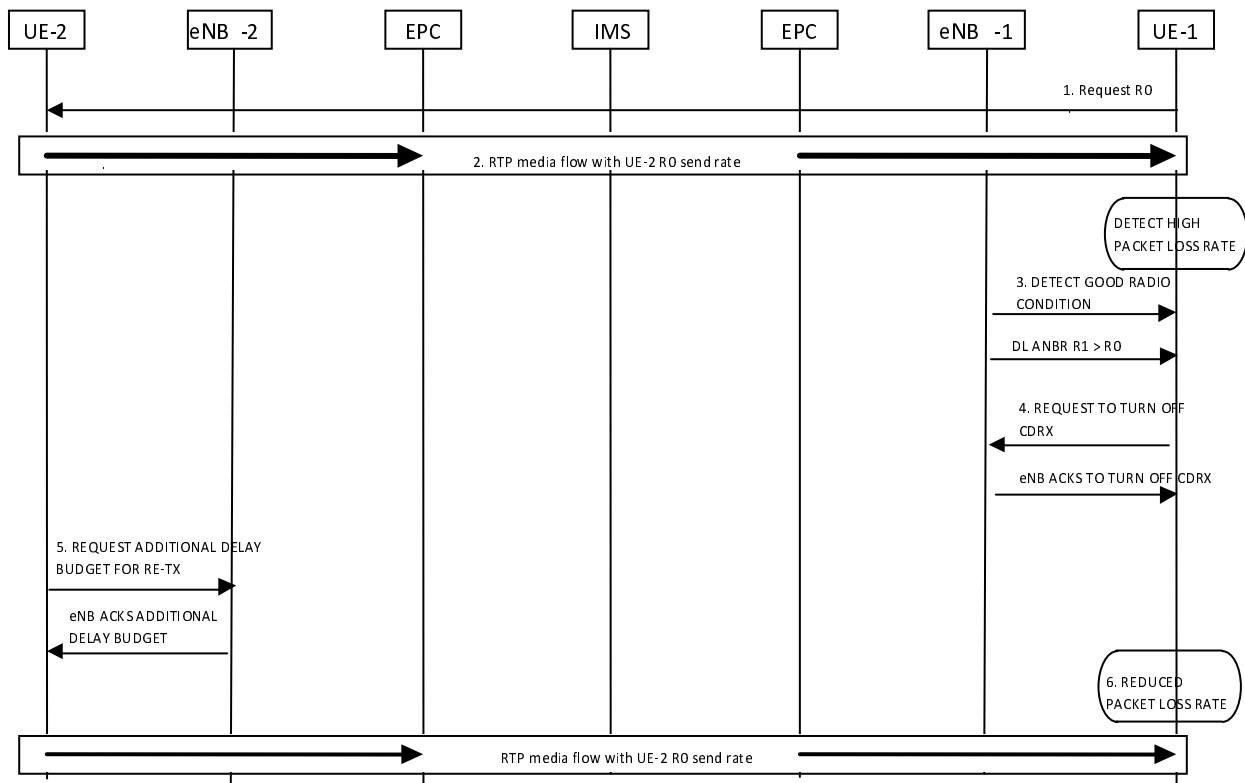


Figure V.2.1: Signaling flow on usage of RAN delay budget reporting in MTSI without DBI signalling.

Step 1: UE-1 sends UE-2 rate request via CMR or RTCP-APP for voice at bitrate R_0 . The "Request" message here is a generalized application level rate request message that corresponds to CMR or RTCP-APP for voice.

Step 2: UE-2 sends RTP media flow for voice with bitrate R_0 .

Step 3: UE-1 detects good radio conditions locally, e.g., eNB-1 sends a DL access network bitrate recommendation (ANBR) of bitrate $R_1 > R_0$ to UE-1, and UE-1 measures low block error rate (BLER) over the local radio link based on the monitoring of successful downlink packet transmissions, and it may also measure downlink throughput over the radio air interface that is much higher than the received bitrate (after accounting for the relevant headers). In the meantime, UE-1 detects high packet losses after monitoring reception of RTP packets (also by monitoring RTCP sender and receiver reports) and applying the highest possible jitter buffer according to the reference Jitter Buffer Management (JBM) in clause 8 (subject to the JBM compliance requirement of MTSI). Hence, UE1 concludes that UE2's local radio conditions are poor.

Step 4: UE-1 sends a *UEAssistanceInformation* message as specified in TS 36.331 [160] to eNB-1 with type-1 to turn off cDRX. It is assumed that eNB-1 grants this request and turns off cDRX for UE-1. Turning off cDRX is relevant only when PLR is high, which is the conclusion of UE-1 in this example, as per Step 3. It should however be noted that UE-1 can increase the JBM depth to compensate the delay for high jitter. In this scenario, delay budget request from UE-1 to eNB-1 is not necessary and *UEAssistanceInformation* message may not be sent. Moreover, due to other considerations, UE-1 may choose not to turn cDRX off, e.g., when saving battery power is critical.

Step 5: UE-2 detects high packet losses on its uplink due to poor coverage conditions, e.g., it may measure high BLER over its local radio link based on the monitoring of successful uplink packet transmissions, e.g., by monitoring the HARQ acknowledgements received. UE-2 requests additional delay budget from eNB-2 in order to perform additional re-transmissions to increase the reliability of its UL transmissions. When requesting this additional delay budget, UE-2 may also consider end-to-end RTT measured based on RTCP reports. It is assumed that eNB-2 grants this request. Because UE-1 has already turned its cDRX off, it is unlikely that the JBM constraint at UE-1 will lead to packet losses in response to the increase air interface delay over the RAN corresponding to UE-2.

Step-6: UE-1 measures reduced packet losses and improved voice quality.

It should be noted that the actions of UE-1 in Steps 3-4 above and actions of UE-2 in Step 5 above are completely independent and these are not necessarily sequential, as there is no coordination between the two UEs in this example.

In Figure V.2.2, another signaling flow example for RAN delay budget reporting usage in MTSI involving delay budget information (DBI) signalling as described in clause 7.3.8 is presented. In this example, uni-directional DBI signalling is depicted with only indication of available delay budget from MTSI receiver to MTSI sender.

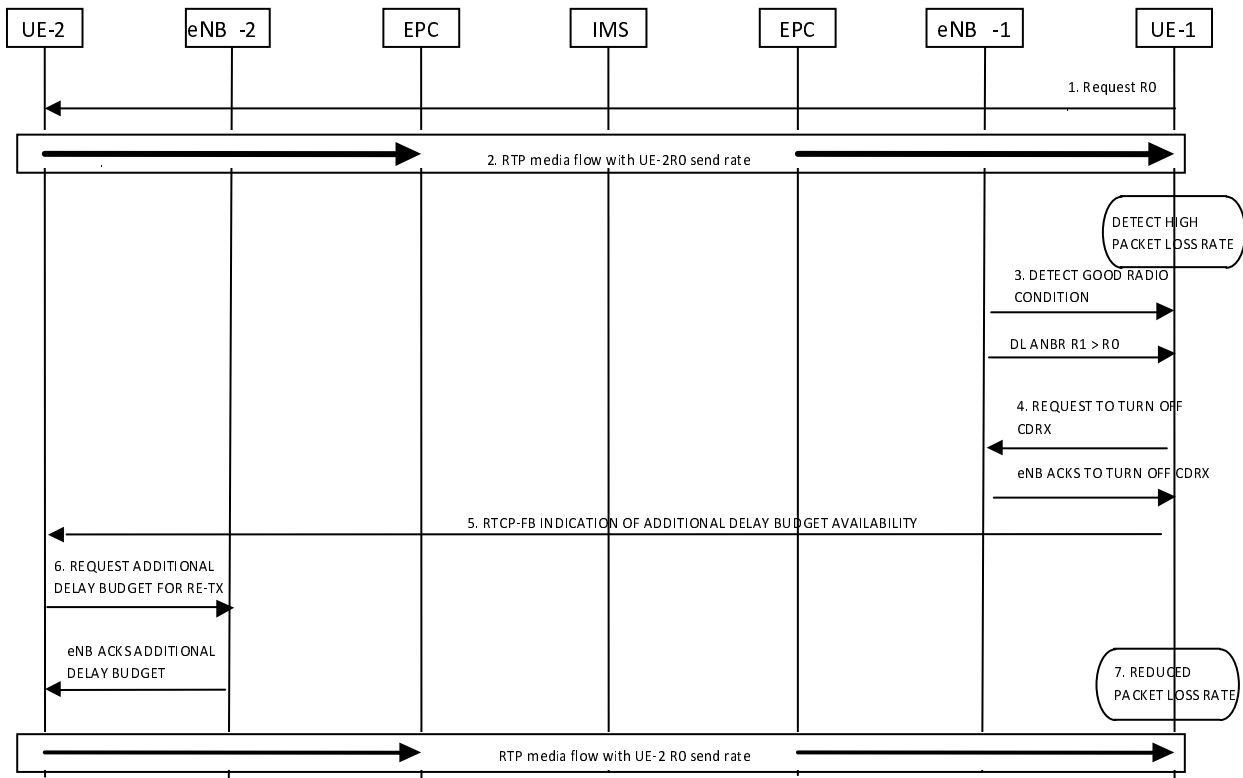


Figure V.2.2: Signaling flow on usage of RAN delay budget reporting with uni-directional DBI signaling in MTSI.

Steps 1-4: These are identical to the earlier signalling flow in Figure V.2.1.

Step 5: If delay budget information signalling is supported between UE-1 and UE-2, UE-1 sends an RTCP feedback (RTCP-FB) message as specified in clause 7.3.8 to UE-2 indicating the availability of additional delay budget due to cDRX being turned off. A concrete delay number may also be reported as part of the RTCP-FB message that corresponds to the air interface delay reduction on UE-1's RAN after turning off cDRX, which would essentially be available for UE-2 to improve the reliability of its uplink transmissions. The reported delay number may also be determined considering UE1's JBM constraints and can be based on its assessment of how much additional delay it can tolerate.

Step 6: UE-2 detects high packet losses on its uplink due to poor coverage conditions. UE-2 requests additional delay budget from eNB-2 in order to perform further re-transmissions to increase the reliability of its UL transmissions. When requesting the additional delay budget from eNB-2, UE-2 may also consider the RTCP-FB message it received from UE-1 on the availability of delay budget from UE-1's perspective. It is assumed that eNB-2 grants this request.

Step-7: UE-1 measures reduced packet losses and improved voice quality.

In this example, the available delay budget may be computed by the UE-1 (MTSI receiver) based on network delay, jitter, packet loss rate (PLR) and potentially other parameters. It may also take into account constraints on JBM (i.e., based on reference JBM in clause 8). In this respect, the following observations can be made on the expected UE behaviour:

- a) Allowing UE-2 to use more retransmission will increase the jitter. This may potentially cause more packets dropped at the JBM for UE-1.
- b) On the other hand, more retransmission also allows UE-2 to reduce packet losses in its RAN uplink and this means more end-to-end reliability. So there is a fine balance here, while it would also be expected that the end-to-end performance is limited by the high packet losses on the UL, and hence more retransmissions will help improve the end-to-end quality and delay performance.

- c) UE-1 turning off cDRX will help to reduce end-to-end delay. In the meantime, if UE-1 needs to save on battery power and it is critical that cDRX is kept on for this purpose, then UE-1 may choose not to turn cDRX off. Even with cDRX off, if UE-1 decides that it can tolerate any further delay or jitter at its JBM, it may indicate this available delay budget via the RTCP-FB message for DBI signalling as defined in clause 7.3.8.
- d) It is up to UE-1 to signal any additional delay budget to UE-2. If UE-1 figures that it is already close to its JBM constraint and cannot tolerate any additional delay or jitter (as this would lead to more packets being dropped), it may not signal additional delay availability to UE-2.

The signalling flow example in Figure V.2.2 relies on DBI signalling between the MTSI sender and MTSI receiver whereas the signalling flow in Figure V.2.1 does not include such signalling. The following can be observed when DBI signalling is not present:

- 1) While the MTSI sender and MTSI receiver UEs may both be independently able to adjust their air interface delays based on the information in their MTSI clients, they are never aware of the capabilities or actions of the other UE. For example, while an MTSI receiver in good coverage may turn off cDRX to create delay budget for an MTSI sender, it may be the case that the MTSI sender does not even support delay budget reporting, or that the MTSI sender's eNB may not grant the additional delay budget to the MTSI sender, so the effort of the MTSI receiver may not deliver any end-to-end performance gain, and end up wasting the battery power of the MTSI receiver UE. Likewise, an MTSI sender in poor coverage may increase its air interface delay in an attempt to perform further retransmissions to mitigate against packet losses, without any knowledge of the possible detrimental impacts on the MTSI receiver, e.g., packets being dropped at the jitter buffer management (JBM) level.
- 2) When UE-1 and UE-2 independently adjust their air interface delays, they rely on the end-to-end measurements available at their MTSI clients, e.g., by monitoring reception of RTP packets and RTCP sender and receiver reports, and knowledge of their local radio conditions. Purely relying on this information, an MTSI receiver may not be able to correctly detect the need for additional delay budget at the MTSI sender, e.g., as it may be the case that the losses are caused in the network. An explicit indication from the MTSI sender as would be enabled by DBI signaling can help the MTSI receiver make the right conclusion. Moreover, the MTSI receiver may not be able to determine exactly how much additional delay budget is needed on the air interface for the MTSI sender UE. Likewise, an MTSI sender may not be able to determine exactly how much additional delay budget it could ask from its eNB in the absence of any DBI signaling.
- 3) With the use of DBI signaling, delay budget adaptation and consequent larger number of retransmissions can be done faster via the real time exchange of delay budget information using RTP/RTCP signalling compared to the case when DBI signalling is absent, which would have to rely on measurements and inference at the UEs based on packet statistics, collection of which requires a certain observation period and averaging window.

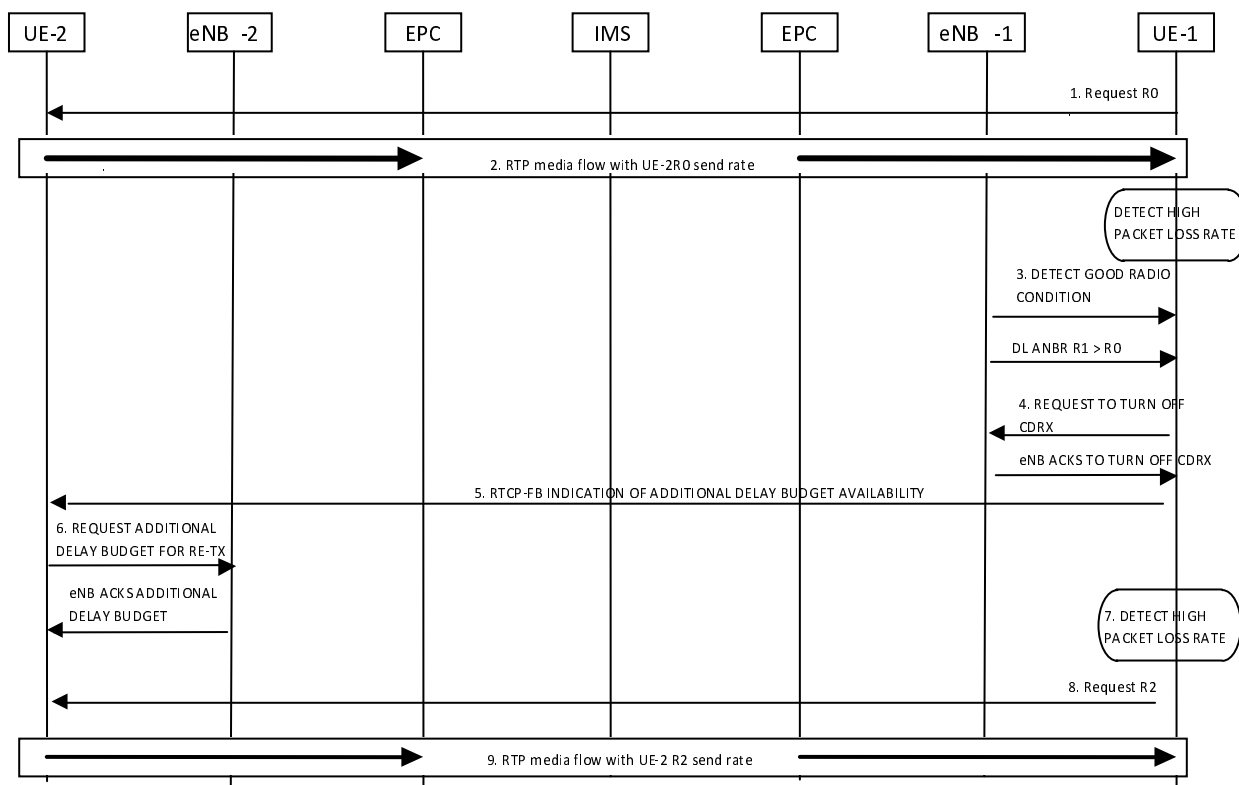


Figure V.2.3: Another signaling flow on usage of RAN delay budget reporting with uni-directional DBI signalling in MTSI.

The signaling flow in Figure V.2.3 is identical to the one in Figure V.2.2, except that in step 7, UE-1 still observes high packet loss rate and the following additional steps are taken:

Step 8: UE-1 sends UE-2 rate request via CMR or RTCP-APP for voice at bitrate $R2 < R0$. Other kinds of adaptation may also be invoked, for instance the use of application layer redundancy or transitioning to a more robust codec mode based on the negotiated codecs (e.g., channel-aware mode for EVS), in case the receiver side detects major packet loss but delay and jitter are within desired bounds.

Step 9: UE-2 sends RTP media flow for voice with bitrate $R2$.

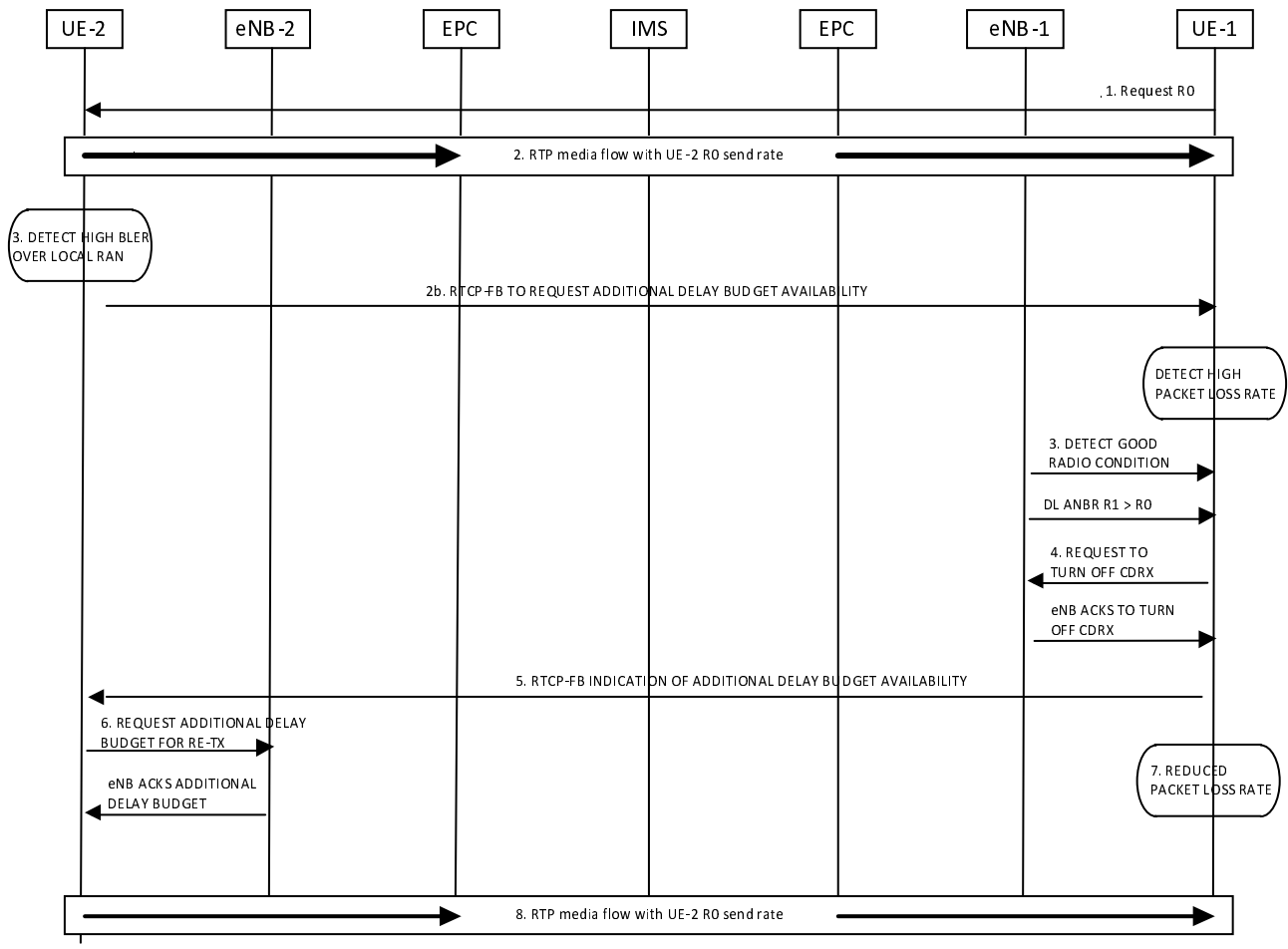


Figure V.2.4: Signaling flow on usage of RAN delay budget reporting in MTSI with bi-directional DBI signalling.

The signalling flow in Figure V.2.4 is a variant of the one in Figure V.2.2, where UE-2 requests additional delay budget from UE-1 during the media flow (as depicted by Step 2b in Figure V.2.4), e.g., via the use of an RTCP-FB message for DBI signalling as defined in clause 7.3.8, after having detected poor radio conditions (e.g., high BLER) over the local RAN. The presence of this request message may further inform UE-1 that the radio conditions on UE-2's side are poor (in addition to its own detection, e.g., based on monitoring of RTP receive statistics). Another benefit of the bi-directional exchange of delay budget information between the two UEs is that this could help in identifying the scenario where the packet losses are introduced by neither of the RANs of UE1 and UE2 (i.e., both UEs enjoying good radio conditions) but rather by the network. In the meantime, it should be noted that the dominant cause of packet losses is expected to be from RAN impediments and the likelihood that the packet losses are caused by the network is quite small.

When DBI signalling is present as in examples provided in Figures V.2.2, V.2.3 and V.2.4, the frequency of the DBI signalling needs to be limited such that terminals do not continuously have to expect new delay budget signaling, i.e., RTCP-FB message frequencies for both available delay budget and requested delay budget signalling need to be limited.

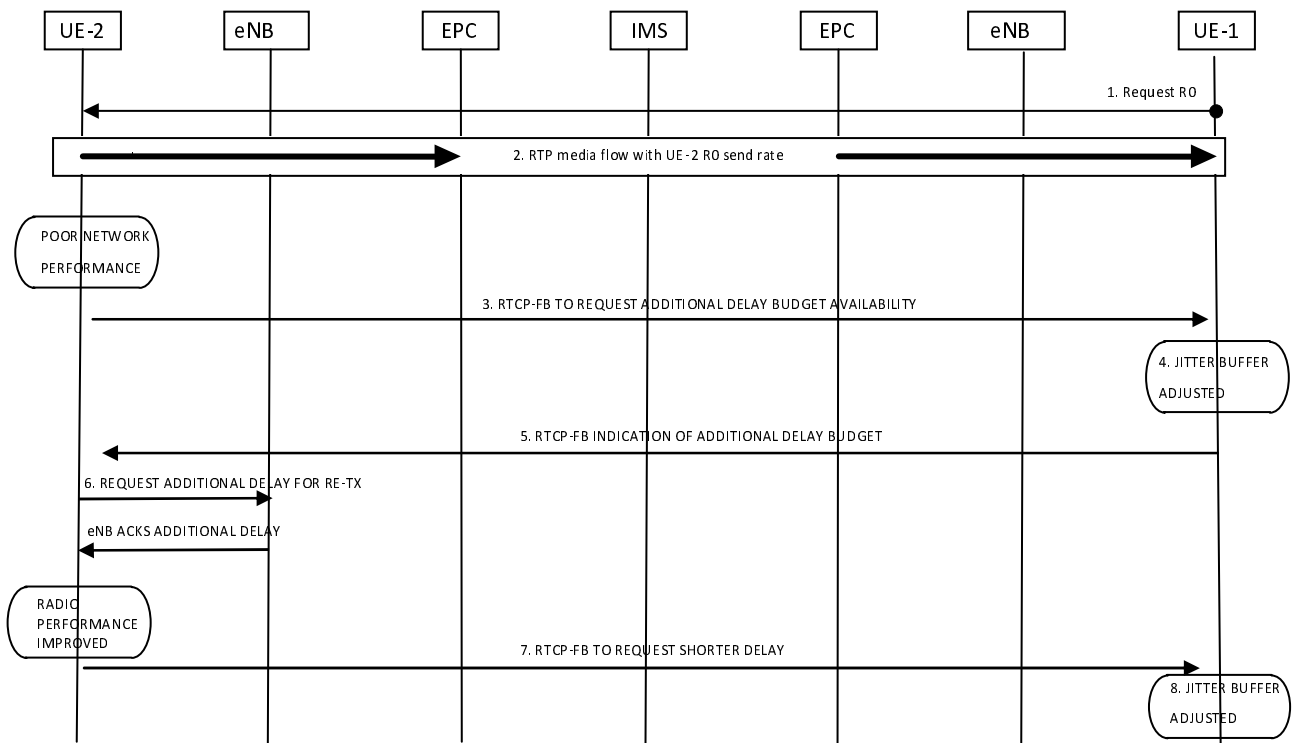


Figure V.2.5: Signaling flow on usage of RAN delay budget reporting in MTSI with bi-directional DBI signalling and jitter buffer adjustment.

The signalling flow in Figure V.2.5 is a case of jitter buffer adjustment where UE-2 is suffering from poor network performance, e.g. under continuous weak coverage. UE-2 requests additional delay budget from UE-1 during the media flow (Step 3 in Figure V.2.5), e.g., via the use of an RTCP-FB message for DBI signalling as defined in clause 7.3.8, where a certain amount of expected extra delay is indicated. After receiving the request, jitter buffer in UE-1 may be extended (Step 4 in Figure V.2.5) to allow the sender to perform more uplink retransmissions and feedback to UE-2 how much additional delay is available for the uplink retransmissions after jitter buffer adjustment (Step 5 in Figure V.2.5). When UE-2 detects radio performance improved, it could notify UE-1 that it does not need an additional delay via another RTCP-FB message for DBI signalling to request a negative delay budget value (Step 7 in Figure V.2.5) and then UE-1 can shorten the jitter buffer accordingly (Step 8 in Figure V.2.5).

V.3 SDP Examples on DBI Signaling Capability

Table V.3.1 demonstrates an example SDP offer with Delay Budget Information (DBI) signalling capability for speech, based on the procedures specified in clause 6.2.8. The offer for DBI is indicated in the last line. The use of RTCP feedback messages carrying 'DBI' is negotiated with the '3gpp-delay-budget' parameter.

Table V.3.1: Example SDP offer with DBI

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 b=AS:42 b=RS:0 b=RR:2000 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=220 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=220 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=220; octet-align=1 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=220 a=rtpmap:101 AMR/8000/1 </pre> |

```
a=fmtp:101 mode-change-capability=2; max-red=220; octet-align=1
a=ptime:20
a=maxptime:240
a=rtcp-fb:* 3gpp-delay-budget
```

An example SDP answer is shown in Table V.3.2, where the DBI signalling capability is also supported by the answerer, as indicated by the last line.

Table V.3.2: Example SDP answer with DBI

| SDP answer |
|--|
| <pre>m=audio 49152 RTP/AVPF 97 b=AS:30 b=RS:0 b=RR:2000 a=acfg:1 t=1 a=rtptime:97 EVS/16000/1 a=fmtp:97 br=5.9-13.2; bw=nb-wb; mode-set=0,1,2; max-red=220 a=ptime:20 a=maxptime:240 a=rtcp-fb:* 3gpp-delay-budget</pre> |

Annex W (Normative): Coverage and Handoff Enhancements using Multimedia Error Robustness (CHEM)

W.1 General

The robustness of media to packet loss can be affected by the following factors in the MTSI terminal:

- codec
- codec mode
- application layer redundancy
- packet loss concealment implementation
- dejitter buffer implementation

The above factors are considered to be part of the codec configuration.

The level of robustness can affect the coverage area of the service in a network. Furthermore, communicating the level of robustness of the media to the network enables the eNB/gNB to use this information to determine a threshold for when the terminal should be handed off to another cell, domain (circuit-switched vs. packet-switched), or radio access technology.

The MTSI terminal may support the coverage and handover enhancements using the multimedia error robustness (CHEM) feature specified in this clause. When CHEM is supported by terminals, the terminals exchange packet loss rate (PLR) information via specified SDP parameters. A PCF/PCRF may use these SDP PLR values to set handover thresholds in their local eNB/gNBs.

In the procedures specified in the rest of this clause, whenever the UE estimates the packet loss rate at the media receiver, the estimate can be measured either pre-dejitter buffer or post-dejitter buffer. When packet loss rate is measured post-dejitter buffer, presentation of partial redundancy information (e.g., in EVS channel aware mode) does not count towards a correctly received packet. Unless specifically configured using the OMA-DM Management Objects, the post-de-jitter buffer PLR estimate shall be used.

The PLR_adapt attribute to be used for this feature is specified using the following ABNF:

Name: PLR_adapt

Value: [plr-adapt-value]

Usage Level: media

Charset Dependent: no

Syntax:

plr-adapt-value = %s"ALR" / token

Examples:

a=PLR_adapt

a=PLR_adapt:ALR

The "ALR" parameter is optional and may be omitted when use of Application-Level Redundancy is not supported. The semantics of the above attribute and parameter are specified below. Unsupported parameters of the PLR_adapt attribute may be ignored.

The IANA registration information for the PLR_adapt SDP attribute is provided in Annex M.9.

W.2 Adaptation to Packet Losses without application layer redundancy

An MTSI terminal supporting the CHEM feature without using application layer redundancy for a media type (e.g., speech or video) shall support the following procedures:

- when sending an SDP offer, the MTSI client shall include the `PLR_adapt` attribute in the media description for that media type in the SDP offer
- when sending an SDP answer, the MTSI client shall include the `PLR_adapt` attribute in the media description for that media type in the SDP answer regardless of whether the `PLR_adapt` attribute was received in an SDP offer
- if, and only if, the MTSI client receives the `PLR_adapt` attribute in either an SDP offer or SDP answer then,
 - when the MTSI client receiving media detects packet losses higher than tolerable by the current codec mode/configuration in use and a more robust codec mode/configuration is available for the same codec, the MTSI client shall send a CMR request to the media sender to use a more robust codec mode/configuration of the same codec
 - when the MTSI client receiving media detects a packet loss rate low enough to support a codec mode/configuration of the same codec that provides better media quality than the current codec mode/configuration, and switching to the new codec mode/configuration will not cause oscillating between more robust and less robust codec modes/configurations, then the MTSI client should send a CMR request to the media sender to use the codec mode/configuration of the same codec that provides better media quality

Clause X.2.2 provides examples of how the `PLR_adapt` attribute without application layer redundancy is used in SDP.

Robustness adaptation without application layer redundancy can involve switching to a codec mode/configuration with partial-redundancy (e.g., EVS Channel Aware mode for speech), better error correction, or lower bit rate. In the last scenario this can benefit the resulting media quality if the terminal is near the edge of coverage and the link can better support transport of lower rate codec modes/configurations.

W.3 Adaptation to Packet Losses using Application Layer Redundancy

An MTSI terminal supporting the CHEM feature supporting use of application layer redundancy to improve robustness for a media type (e.g., speech or video) shall support the following procedures:

- when sending an SDP offer, the MTSI client shall include the `ALR` parameter in the `PLR_adapt` attribute in the media description for that media type in the SDP offer
- when sending an SDP answer, the MTSI client shall include the `ALR` parameter in the `PLR` attribute in the media description for that media type in the SDP answer only if the `ALR` parameter in the `PLR_adapt` attribute was received in an SDP offer

In addition to requesting the media sender to adapt codec mode/configuration as described in clause W.2, the MTSI terminal may request the media sender to use non-zero application layer redundancy to improve robustness to packet loss. Clause X.2.3 provides examples of how the `PLR_adapt` attribute and `ALR` parameter (i.e., with application layer redundancy) is used in SDP.

When using CMR to request application layer redundancy the following CMR code points shall be used only if the MTSI client receives the `ALR` parameter in the `PLR_adapt` attribute in either an SDP offer or SDP answer:

Table W.3.1: Code points for AMR

| CMR code | Application layer redundancy AMR request |
|----------|--|
| 9 | RED 2x4.75 |
| 10 | RED 2x5.15 |
| 11 | RED 2x5.9 |
| 12 | Not used |
| 13 | Not used |
| 14 | Not used |

Table W.3.2: Code points for AMR-WB

| CMR code | Application layer redundancy AMR-WB request |
|----------|---|
| 9 | RED 2x6.6 |
| 10 | RED 2x8.85 |
| 11 | RED 2x12.65 |
| 12 | Not used |
| 13 | Not used |
| 14 | Not used |

Table W.3.3: Code points for EVS

| CMR code | Application layer redundancy EVS request |
|----------|--|
| 111 0000 | RED 2x7.2-NB |
| 111 0001 | RED 2x8-NB |
| 111 0010 | RED 2x9.6-NB |
| 111 0011 | RED 2x13.2-NB |
| 111 0100 | RED 2x7.2-WB |
| 111 0101 | RED 2x8-WB |
| 111 0110 | RED 2x9.6-WB |
| 111 0111 | RED 2x13.2-WB |
| 111 1000 | RED 2x13.2 CAM WB |
| 111 1001 | RED 2x13.2 CAM SWB |
| 111 1010 | RED 2x9.6-SWB |
| 111 1011 | RED 2x13.2-SWB |
| 111 1100 | RED 2x6.6-IO |
| 111 1101 | RED 2x8.85-IO |
| 111 1110 | RED 2x12.65-IO |

To enable use of application layer redundancy the max-red SDP parameter should be selected in the SDP offer and answer to give at least some room for application layer redundancy adaptation and in accordance with clauses 6.2.2.2 and 6.2.2.3. The recommended redundancy offset for application layer redundancy should be 3 frames.

W.4 Negotiation of End-to-End and Uplink/Downlink PLR

W.4.1 General

In addition to the negotiated codecs and codec modes, the end-to-end quality and robustness to packet loss of a service can depend on the MTSI client implementation of functions such as packet loss concealment (PLC) and de-jitter buffer management (JBM).

A set of end-to-end and link-by-link PLR parameters is specified to provide the following:

- enable MTSI clients to indicate for each RTP payload type the effective maximum tolerable PLR resulting from the codec, codec mode, packet loss concealment, de-jitter buffer management and other implementation considerations.

- enable MTSI clients to negotiate with each other and suggest to the PCRF/PCFs how the maximum tolerable end-to-end PLR budget can be distributed across the uplink and downlink in a media transport path. In the absence of any additional negotiation or information from the terminals, the PCRF/PCFs may default to distributing the maximum tolerable end-to-end PLR equally (50-50) across the uplink and downlink. The link-specific PLR parameters enable the terminals to suggest to the PCRF/PCFs an unequal distribution, e.g. 60% of the maximum end-to-end PLR allowed on the uplink and 40% allowed on the downlink before the terminals are handed off.
- indicate to the PCRFs/PCFs what PLR values may be used on the local uplink and downlink to the MTSI clients.

W.4.2 Offering MTSI Client

SDP PLR attributes and parameters are defined for use with each payload type. For each RTP payload type, an offering MTSI client supporting the CHEM feature may include the following SDP attribute in its SDP offer:

Name: MAXimum-e2e-PLR(maximum end-to-end PLR of the media decoder in the MTSI client)

Value: MAX-e2e-PLR-value

Usage Level: media

Charset Dependent: no

Syntax:

MAX-e2e-PLR-value = payload-type SP maxe2e-PLR [“:”maxDL-PLR] [“/”maxUL-PLR]

payload-type = zero-based-integer

maxe2e-PLR = plr-value

maxDL-PLR = plr-value

maxUL-PLR = plr-value

plr-value = integer

: integer taken from IETF RFC 4566

The IANA registration information for the MAXimum-e2e-PLR SDP attribute is provided in Annex M.10.

The maxe2e-PLR represents the maximum end-to-end packet loss rate that can be properly received by the media decoder in the offering MTSI client (including effects of codec mode, packet loss concealment, de-jitter buffering, etc...) indicated by the RTP payload type number payload-type (as used in an “m=” line).

The maxDL-PLR parameter indicates the maximum packet loss rate that the offering MTSI client is able to handle on its local downlink and shall not exceed the maxe2e-PLR value in the SDP offer. If the parameter is not included then the default value from the offering MTSI client is set to ½ of the maxe2e-PLR value included in the SDP offer.

The maxUL-PLR parameter indicates the maximum packet loss rate that the offering MTSI client is able to handle on its local uplink. If the parameter is not included then the default value from the offering MTSI client is set to ½ of maxe2e-PLR value the answering MTSI client will include in its corresponding SDP answer.

If maxe2e-PLR is not included in the SDP answer then its value is set to a recommended value for the codec and mode (e.g., obtain value from Annex Y if available).

The plr-value represents 1/100th of a percent (i.e. 10E-4) of packet loss as an integer.

The following table W.4.2-1 summarizes the SDP offer parameters.

Table W.4.2-1 SDP offer attributes and parameters

| Parameter | Requirement | Default Value if not included in SDP offer | Limit | Limit Condition |
|------------|--------------------------------------|---|---|-----------------|
| maxe2e-PLR | Included when attribute is supported | A recommended value for the codec and mode (e.g., obtain value from Annex X if available) | | |
| maxDL-PLR | Optional | $\frac{1}{2}$ maxe2e-PLR value in SDP offer | No greater than maxe2e-PLR value in SDP offer | Always |
| maxUL-PLR | Optional | $\frac{1}{2}$ maxe2e-PLR value in SDP answer | | |

Clause X.2.4 provides examples of how the MAXimum-e2e-PLR attribute is used in the SDP offer.

W.4.3 Answering MTSI Client

For each RTP payload type, an answering MTSI client supporting the CHEM feature may include the SDP attribute from sub-clause W.4.2 in its SDP answer with the following semantics of the parameters.

The maxe2e-PLR represents the maximum end-to-end packet loss rate that can be properly received by the media decoder in the answering MTSI client (including effects of codec mode, packet loss concealment, de-jitter buffering, etc...) indicated by the RTP payload type number payload-type (as used in an “m=” line).

If maxe2e-PLR is not included in the SDP offer then its value is set to a recommended value for the codec and mode (e.g., obtain value from Annex Y if available).

The maxDL-PLR parameter included in the SDP answer indicates the maximum packet loss rate that the answering MTSI client is able to handle on its local downlink and shall not exceed the maxe2e-PLR value in the SDP answer.

If the maxUL-PLR value included in the SDP offer is no greater than $\frac{1}{2}$ the maxe2e-PLR value included by the answering MTSI client in the SDP answer, then maxDL-PLR in the SDP answer shall be set to no greater than (maxe2e-PLR value included in the SDP answer minus maxUL-PLR value included in the SDP offer). Otherwise (the maxUL-PLR value included in the SDP offer is greater than $\frac{1}{2}$ the maxe2e-PLR value included in the SDP answer) the maxDL-PLR in the SDP answer should be set to no greater than (maxe2e-PLR value included in the SDP answer minus maxUL-PLR value included in the SDP offer). If the answerer sets the maxDL-PLR in the SDP answer to be greater than (maxe2e-PLR value included in the SDP answer minus maxUL-PLR value included in the SDP offer), the answerer shall not set maxDL-PLR greater than $\frac{1}{2}$ maxe2e-PLR value included in the SDP answer.

If the maxDL-PLR parameter is not included in the SDP answer then the default value from the answering MTSI client is set to $\frac{1}{2}$ of the maxe2e-PLR value included in the SDP answer.

The maxUL-PLR parameter included in the SDP answer indicates the maximum packet loss rate that the answering MTSI client is able to handle on its local uplink and shall not exceed the maxe2e-PLR value in the SDP offer.

If the maxDL-PLR value included in the SDP offer is no greater than $\frac{1}{2}$ the maxe2e-PLR value included by the offering MTSI client in the SDP offer, then maxUL-PLR in the SDP answer shall be set to no greater than (maxe2e-PLR value included in the SDP offer minus maxDL-PLR value included in the SDP offer). Otherwise (if the maxDL-PLR value included in the SDP offer is greater than $\frac{1}{2}$ the maxe2e-PLR value included in the SDP offer), then maxUL-PLR in the SDP answer should be set to no greater than (maxe2e-PLR value included in the SDP offer minus maxDL-PLR value included in the SDP offer). If the answerer sets the maxUL-PLR in the SDP answer to be greater than (maxe2e-PLR value included in the SDP offer minus maxDL-PLR value included in the SDP offer), the answerer shall not set maxUL-PLR greater than $\frac{1}{2}$ maxe2e-PLR value included in the SDP offer.

If the maxUL-PLR parameter is not included then the default value from the answering MTSI client is set to $\frac{1}{2}$ of the maxe2e-PLR value included in the SDP offer.

The plr-value represents $1/100^{\text{th}}$ of a percent (i.e. $10\text{E-}4$) of packet loss as an integer.

The table W.4.3-1 below summarizes the usage of the SDP answer attributes and parameters.

Table W.4.3-1 Usage of the SDP answer attributes and parameters

| Attribute/ Parameter | Requirement | Default Value if not included in SDP Answer | Limit Condition | Limit |
|-------------------------|--------------------------------------|---|--|---|
| maxe2e-PLR | Included when attribute is supported | A recommended value for the codec and mode (e.g., obtain value from Annex X if available) | | |
| maxDL-PLR | Optional | maxe2e-PLR value included in the SDP answer minus maxUL-PLR value included in the SDP offer. If maxUL-PLR is not included in the SDP offer then the default value is $\frac{1}{2}$ maxe2e-PLR value in the SDP answer. | maxUL-PLR value included in the SDP offer is no greater than $\frac{1}{2}$ maxe2e-PLR value included in the SDP answer | maxe2e-PLR value included in the SDP answer minus maxUL-PLR value included in the SDP offer |
| | | | maxUL-PLR value included in the SDP offer is greater than $\frac{1}{2}$ maxe2e-PLR value included in the SDP answer | $\frac{1}{2}$ maxe2e-PLR value in the SDP answer |
| maxUL-PLR | Optional | maxe2e-PLR value included in the SDP offer minus maxDL-PLR value included in the SDP offer. If maxDL-PLR is not included in the SDP offer then the default value is $\frac{1}{2}$ maxe2e-PLR value in the SDP offer. | maxDL-PLR value included in the SDP offer is no greater than $\frac{1}{2}$ maxe2e-PLR value included in the SDP offer | maxe2e-PLR value included in the SDP offer minus maxDL-PLR value included in the SDP offer |
| | | | maxDL-PLR value included in the SDP offer is greater than $\frac{1}{2}$ maxe2e-PLR value included in the SDP offer | $\frac{1}{2}$ maxe2e-PLR value in the SDP offer |

Clause X.2.4 provides examples of how the MAXimum-e2e-PLR attribute is used in the SDP answer.

Annex X (Informative) : Example Maximum Packet Loss Rate (Max. PLR) Values for Setting CHEM Handover Thresholds

X.1 Maximum Packet Loss Rate (Max. PLR) for Speech

Based on the 3GPP EVS Selection and Characterization results that included AMR-WB, AMR-WB with G718IO, and EVS codec, this clause provides an example set of Max. PLR operating points that the terminal may indicate to the PCRF/PCF.

X.1.1 Max. PLR recommendation without Application Layer Redundancy

Table X.1 provides example Maximum PLR operating points based on the EVS Selection and Characterization experiment results in TR 26.952 [168] and TR 26.959 [169].

Table X.1: Example Maximum End-to-end Packet Loss Rate (PLR) per link for AMR-WB and EVS

| Codec | Robustness Parameter | Maximum End-to-end Packet Loss Rate |
|----------------------------------|----------------------|-------------------------------------|
| AMR-WB | Normal | 1.5% |
| AMR-WB/G718 IO, EVS AMR-WB IO | Medium | 3% |
| EVS WB, SWB | High | 6% |
| EVS WB, SWB Channel Aware | Extreme High | 9% |

X.1.2 Max. PLR recommendation with Application Layer Redundancy

Application layer redundancy can work in conjunction with any of the aforementioned codec modes in Table X.1.

Table X.2 provides example Maximum PLR operating points with and without application layer redundancy applicable to EVS codec based on informal objective and subjective results in Annex A of TR 26.959 [169].

The example in Table X.2 includes 100% application layer redundancy with offset 2, resulting in (2 x Bitrate). It should be noted that the relationship to path loss when operating at twice the bit rate (with 100% application layer redundancy) is not accounted in the Max. PLR recommendation in Table X.2.

Table X.2: Example Max. End-to-end Packet Loss Rate (PLR) with application layer redundancy for EVS codec

| Codec | Robustness Parameter | Maximum End-to-end Packet Loss Rate |
|--|----------------------|-------------------------------------|
| No application layer redundancy, EVS at bitrate of R kb/s | - | X % |
| With 100% application layer redundancy, EVS at bitrate of 2xR kb/s, Offset = 2. | - | X + (2 to 5) % |

X.2 SDP Examples of the CHEM Feature (informative)

X.2.1 General

The following examples illustrate the use of the SDP attributes and parameters specified for the CHEM feature.

X.2.2 Example of Adaptation to Packet Losses without Application Layer Redundancy

The example in Table X.2.2-1 demonstrates how, as specified in clause W.2, the receiver in the offering MTSI client supports and offers to request adaptation to different codec configurations to provide different levels of packet loss robustness without using application layer redundancy.

Table X.2.2-1: SDP offer supporting adaptation to packet loss without using application layer redundancy

| SDP offer |
|--|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 105 106 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=200 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=200; octet-align=1 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=200 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=200; octet-align=1 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 a=rtpmap:106 telephone-event/8000 a=fmtp:106 0-15 a=ptime:20 a=maxptime:240 a=PLR_adapt </pre> |

The example in Table X.2.2-2 is one possible response to the offer in Table X.2.2-1 and demonstrates how, as specified in clause W.2, the receiver in the answering MTSI client also supports and negotiates requesting adaptation to different codec configurations to provide different levels of packet loss robustness without using application layer redundancy. During the ensuing session both the offerer MTSI client and answerer MTSI client request robustness adaptation

without application layer redundancy. The PCRF/PCFs should choose to use more robust handover thresholds that do not rely on application layer redundancy in both the uplink and downlink directions.

Table X.2.2-2: SDP answer supporting adaptation to packet loss without using application layer redundancy

| SDP answer |
|---|
| <pre>m=audio 49152 RTP/AVP 97 105 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 aptime:20 a=maxptime:240 a=PLR_adapt</pre> |

The example in Table X.2.2-3 is another possible response to the offer in Table X.2.2-1 and demonstrates how, as specified in clause W.2, the receiver in the answering MTSI client does not support requesting adaptation to different codec configurations to provide different levels of packet loss robustness. In the ensuing session, media robustness adaptation is not enabled in either direction of media transmission so the PCRF/PCFs should not use more robust handover thresholds in either the uplink or downlink direction.

Table X.2.2-3: SDP answer not supporting adaptation to packet loss

| SDP answer |
|--|
| <pre>m=audio 49152 RTP/AVP 97 105 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 aptime:20 a=maxptime:240</pre> |

X.2.3 Example of Adaptation to Packet Losses with Application Layer Redundancy

The example in Table X.2.3-1 demonstrates how, as specified in clause W.3, the receiver in the offering MTSI client supports and offers to request adaptation to different codec configurations to provide different levels of packet loss robustness using application layer redundancy.

Table X.2.3-1: SDP offer supporting adaptation to packet loss using application layer redundancy and the in-band RTP CMR code points specified in clause W.3

| SDP offer |
|--|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 105 106 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=200 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=200; octet-align=1 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=200 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=200; octet-align=1 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 a=rtpmap:106 telephone-event/8000 a=fmtp:106 0-15 a=ptime:20 a=maxptime:240 a=PLR_adapt:ALR </pre> |

The example in Table X.2.3-2 is one possible response to the offer in Table X.2.3-1 and demonstrates how, as specified in clause W.3, the receiver in the answering MTSI client also supports and negotiates requesting adaptation to different codec configurations to provide different levels of packet loss robustness using application layer redundancy. During the ensuing session both the offerer MTSI client and answerer MTSI client request robustness adaptation and can use the in-band RTP CMR codepoints specified in clause W.3 to request application layer redundancy. The PCRF/PCFs should choose to use more robust handover thresholds (both using application layer redundancy or not) in both the uplink and downlink directions.

Table X.2.3-2: SDP answer supporting adaptation to packet loss using application layer redundancy and the in-band RTP CMR code points specified in clause W.3

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVP 97 105 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 a=ptime:20 a=maxptime:240 a=PLR_adapt:ALR </pre> |

The example in Table X.2.3-3 is another possible response to the offer in Table X.2.3-1 and demonstrates how, as specified in clause W.2 and clause W.3, the receiver in the answering MTSI client supports requesting adaptation to different codec configurations but not using application layer redundancy. During the ensuing session both the offerer MTSI client and answerer MTSI client request robustness adaptation without application layer redundancy. The PCRF/PCFs should choose to use more robust handover thresholds that do not rely on application layer redundancy in both the uplink and downlink directions.

Table X.2.3-3: SDP answer supporting adaptation to packet loss without use of application layer redundancy

| SDP answer |
|--|
| <pre> m=audio 49152 RTP/AVP 97 105 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 aptime:20 a=maxptime:240 a=PLR_adapt </pre> |

X.2.4 Example of Maximum End-to-End Packet Loss Rate

The example in Table X.2.4-1 demonstrates how, as specified in clause W.4.2, the media receiver in the offering MTSI terminal receiver can handle up to 3.50% end-to-end PLR for the EVS codec, proposes to use 1.50% PLR on its downlink and 2.00% for its uplink, and can handle up to 2.00% end-to-end PLR for the AMR-WB codec, proposes to use 0.75% PLR on its downlink and 1.25% for its uplink. The example does not use application layer redundancy but could easily apply to sessions using application layer redundancy by including the ALR parameter in the PLR_adapt attribute as specified in clause W.3.

Table X.2.4-1: SDP offer with maximum end-to-end PLR attribute and parameters

| SDP offer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 98 99 100 101 105 106 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:98 AMR-WB/16000/1 a=fmtp:98 mode-change-capability=2; max-red=200 a=rtpmap:99 AMR-WB/16000/1 a=fmtp:99 mode-change-capability=2; max-red=200; octet-align=1 a=rtpmap:100 AMR/8000/1 a=fmtp:100 mode-change-capability=2; max-red=200 a=rtpmap:101 AMR/8000/1 a=fmtp:101 mode-change-capability=2; max-red=200; octet-align=1 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 a=rtpmap:106 telephone-event/8000 a=fmtp:106 0-15 aptime:20 a=maxptime:240 a=PLR_adapt a=MAXimum-e2e-PLR:97 350:150/200 a=MAXimum-e2e-PLR:98 200:75/125 a=MAXimum-e2e-PLR:99 200:75/125 </pre> |

The example in Table X.2.4-2 is one possible response to the offer in Table X.2.4-1 and demonstrates how, as specified in clause W.4.3, the receiver in the answering MTSI terminal supports up to 4.00% end-to-end PLR for the EVS codec, proposes to use 2.00% PLR on both its downlink and uplink. This leaves 1.50% for the downlink and 2.00% for the uplink to the offering MTSI terminal which matches what was proposed in the SDP offer.

Table X.2.4-2: SDP answer with maximum end-to-end PLR attribute and parameters agreeing to what was proposed in the SDP offer

| SDP answer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 105 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 aptime:20 a=maxptime:240 a=PLR_adapt a=MAXimum-e2e-PLR:97 400:200/200 </pre> |

The example in Table X.2.4-3 is another possible response to the offer in Table X.2.4-1 and demonstrates how, as specified in clause W.4.3, the receiver in the answering MTSI terminal supports up to 3.50% end-to-end PLR for the EVS codec, proposes to use 1.75% PLR on both its downlink and uplink. This leaves 1.75% for each of the uplink and downlink to the offering MTSI terminal which is higher than the downlink PLR (i.e., the answerer is being more generous to the offerer) and lower than the uplink PLR (i.e., the answerer not agreeing to let the offerer be greedy) proposed in the SDP offer.

Table X.2.4-3: SDP answer with maximum end-to-end PLR attribute and parameters not agreeing to what was proposed in the SDP offer

| SDP answer |
|---|
| <pre> m=audio 49152 RTP/AVP 97 105 b=AS:50 b=RS:0 b=RR:2500 a=rtpmap:97 EVS/16000/1 a=fmtp:97 br=5.9-24.4; bw=nb-swb; max-red=200 a=rtpmap:105 telephone-event/16000 a=fmtp:105 0-15 aptime:20 a=maxptime:240 a=PLR_adapt a=MAXimum-e2e-PLR:97 350:175/175 </pre> |

Annex Y (normative): Immersive Teleconferencing and Telepresence for Remote Terminals (ITT4RT)

Y.1 General

The MTSI terminal may support the Immersive Teleconferencing and Telepresence for Remote Terminals (ITT4RT) feature as defined in this clause. MTSI clients supporting the ITT4RT feature shall be referred to as ITT4RT clients.

ITT4RT functionality for MTSI enables support of an immersive experience for remote terminals joining teleconferencing and telepresence sessions. It addresses scenarios with two-way audio and one-way immersive 360-degree video, e.g., a remote single user wearing an HMD participating in a conference will send audio and optionally 2D video (e.g., of a presentation, screen sharing and/or a capture of the user itself), but receives stereo or immersive voice/audio and immersive 360-degree video captured by an omnidirectional camera in a conference room connected to a fixed network.

Since immersive 360-degree video support for ITT4RT is unidirectional, ITT4RT clients supporting immersive 360-degree video are further classified into two types to distinguish between the capabilities for sending or receiving immersive video: (i) ITT4RT-Tx client, which is an ITT4RT client only capable of sending immersive 360-degree video, and (ii) ITT4RT-Rx client, which is an ITT4RT client only capable of receiving immersive 360-degree video. Such a classification does not apply for ITT4RT clients supporting immersive speech/audio, since the support for immersive speech/audio is expected to be bi-directional. It should also be noted that a terminal containing ITT4RT-Tx or ITT4RT-Rx client capabilities may also contain further MTSI client capabilities to support bi-directional 2D video.

MTSI gateways supporting ITT4RT functionality are referred to as ITT4RT MRF, which is an ITT4RT client implemented by functionality included in the ITT4RT MRF. An ITT4RT MRF supporting immersive 360-degree video contains both ITT4RT-Tx and ITT4RT-Rx clients.

Y.2 Architecture and Interfaces

Definitions, reference and coordinate systems, video signal representation and audio signal representation as described in clause 4.1 of TS 26.118 [180] are applicable.

Figure Y.1 provides a possible sender architecture that produces the RTP streams containing 360-degree video and immersive speech/audio as applicable to an ITT4RT client in terminal. VR content acquisition includes capture of 360-degree video and immersive speech/audio, as well as other relevant content such as overlays. Following VR content pre-processing and encoding of 360-degree video and immersive speech/audio components, the corresponding elementary streams are generated. For 360-degree projected video, pre-processing may include video stitching, rotation or other translations, and the pre-processed 360-degree video is then passed into the projection functionality in order to map 360-degree video into 2D textures using a mathematically specified projection format. Optionally, the resulting projected video may be further mapped region-wise onto a packed video. For 360-degree fisheye video, circular videos captured by fisheye lenses are not stitched, but directly mapped onto a 2D texture, without the use of the projection and region-wise packing functionalities (as described in clause 4.3 of ISO/IEC 23090-2 [179]). In this case, pre-processing may include arranging the circular images captured by fisheye lenses onto 2D textures, and the functionality for projection and mapping is not needed. For audio, no stitching process is needed, since the captured signals are inherently immersive and omnidirectional. Followed by the HEVC/AVC encoding of the 2D textures and EVS encoding of immersive speech/audio along with the relevant immersive media metadata (e.g., SEI messages), the consequent video and audio elementary streams are encapsulated into respective RTP streams and transmitted.

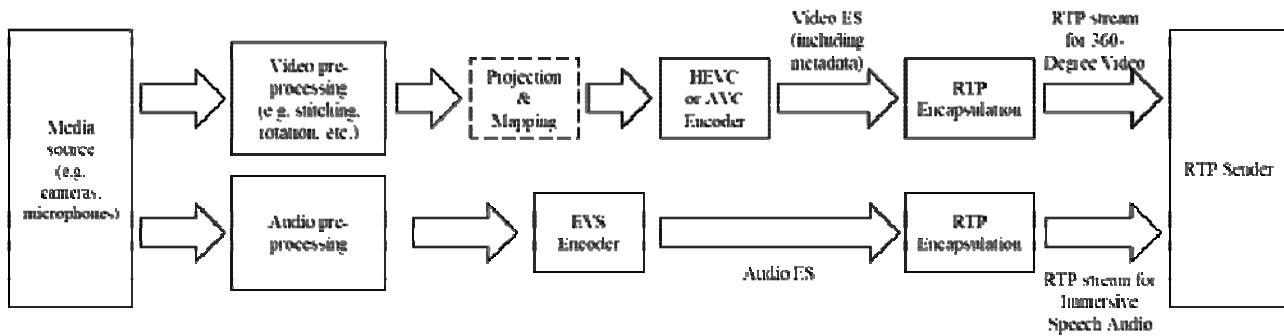


Figure Y.1: Reference sender architecture for ITT4RT client in terminal

Figure Y.2 provides an overview of a possible receiver architecture that reconstructs the 360-degree video and immersive speech/audio in an ITT4RT client in terminal. Note that this figure does not represent an actual implementation, but a logical set of receiver functions. Based on one or more received RTP media streams, the UE parses, possibly decrypts and feeds the elementary video stream into the HEVC/AVC decoder and the speech/audio stream into the EVS decoder. The HEVC/AVC decoder obtains the decoder output signal, referred to as the "2D texture", as well as the decoder metadata. Likewise, the EVS decoder output signal contains the immersive speech/audio. The decoder metadata for video contains the Supplemental Enhancement Information (SEI) messages, i.e., information carried in the omnidirectional video specific SEI messages, to be used in the rendering phase. In particular, the decoder metadata may be used by the Texture-to-Sphere Mapping function to generate a 360-degree video (or part thereof) based on the decoded output signal, i.e., the texture. The viewport is then generated from the 360-degree video signal (or part thereof) by taking into account the pose information from sensors, display characteristics as well as possibly other metadata.

For 360-degree video, the following components are applicable:

- The RTP stream contains an HEVC or an AVC bitstream with omnidirectional video specific SEI messages. In particular, the omnidirectional video specific SEI messages as defined in ISO/IEC 23008-2 [119] and ISO/IEC 14496-10 [24] may be present.
- The video elementary stream(s) are encoded following the requirements in clause Y.3

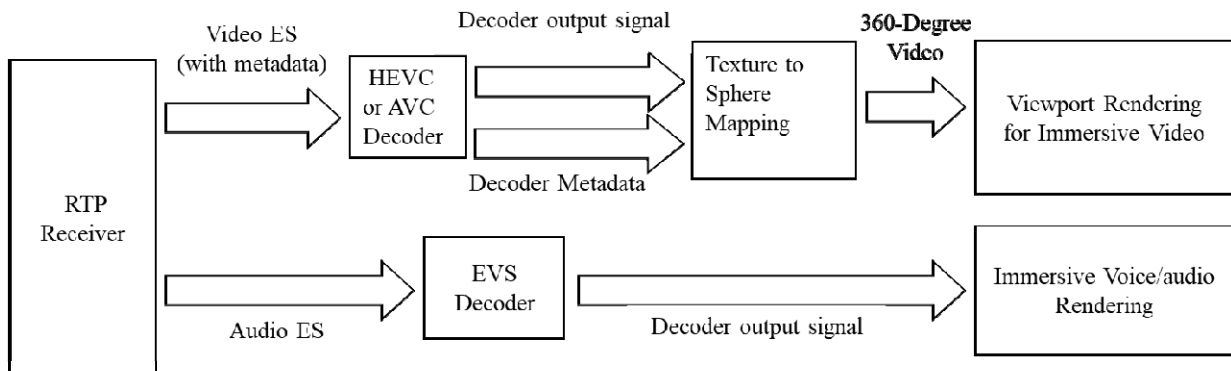


Figure Y.2: Reference receiver architecture for ITT4RT- client in terminal

Figure Y.2: Reference receiver architecture for ITT4RT- client in terminal The output signal, i.e., the decoded picture or "texture", is then rendered using the Decoder Metadata information in relevant SEI messages contained in the video elementary streams as well as the relevant information signalled at the RTP/RTCP level (in the viewport-dependent case). The Decoder Metadata is used when performing rendering operations such as region-wise unpacking, projection de-mapping and rotation for 360-degree projected video, or fisheye video information for 360-degree fisheye video) toward creating spherical content for each eye. Details of such sample location remapping process operations are described in clause D.3.41.7 of ISO/IEC 23008-2 [119].

Viewport-dependent 360-degree video processing could be supported for both point-to-point conversational sessions and multiparty conferencing scenarios and can be achieved by sending from the ITT4RT-Rx client RTCP feedback messages with viewport information and then encoding and sending the corresponding viewport by the ITT4RT-Tx client or by the ITT4RT-MRF. This is expected to deliver resolutions higher than the viewport independent approach

for the desired viewport. The transmitted RTP stream from the ITT4RT-Tx client or ITT4RT-MRF may also include the information on the region of the 360-degree video encoded in higher quality as the video generated, encoded and streamed by the ITT4RT-Tx client may cover a larger area than the desired viewport. Viewport-dependent processing is realized via RTP/RTCP based protocols that are supported by ITT4RT clients. The use of RTP/RTCP based protocols for viewport-dependent processing is further described in clause Y.7.2.

Y.3 Immersive 360-Degree Video Support

ITT4RT-Rx clients in terminals offering video communication shall support decoding capabilities based on:

- H.264 (AVC) [24] Constrained High Profile, Level 5.1 with the following additional restrictions and requirements on the bitstream:
 - the maximum VCL Bit Rate is constrained to be 120 Mbps with `cpbBrVclFactor` and `cpbBrNalFactor` being fixed to be 1250 and 1500, respectively.
 - the bitstream does not contain more than 10 slices per picture.
- H.265 (HEVC) [119] Main 10 Profile, Main Tier, Level 5.1.

In addition, ITT4RT-Rx clients in terminals may support:

- H.265 (HEVC) [119] Screen-Extended Main 10 Profile, Main Tier, Level 5.1.
- H.265 (HEVC) [119] Screen-Extended Main 4:4:4 10 Profile, Main Tier, Level 5.1.

ITT4RT-Tx clients in terminals offering video communication shall support encoding up to the maximum capabilities (e.g., color bit-depth, luma samples per second, luma picture size, frames per second) compatible with decoders compliant with the following on the bitstream:

- H.264 (AVC) [24] Constrained High Profile, Level 5.1 with the following additional restrictions and requirements:
 - the maximum VCL Bit Rate is constrained to be 120 Mbps with `cpbBrVclFactor` and `cpbBrNalFactor` being fixed to be 1250 and 1500, respectively.
 - the bitstream does not contain more than 10 slices per picture.
- H.265 (HEVC) [119] Main 10 Profile, Main Tier, Level 5.1.

In addition, ITT4RT-Tx clients in terminals may support:

- H.265 (HEVC) [119] Screen-Extended Main 10 Profile, Main Tier, Level 5.1.
- H.265 (HEVC) [119] Screen-Extended Main 4:4:4 10 Profile, Main Tier, Level 5.1.

Hence, for a Bitstream conforming to the H.264 (AVC) [24] Constrained High Profile, Level 5.1 delivered from an ITT4RT-Tx client to the ITT4RT-Rx client, the following restrictions apply:

- The `profile_idc` shall be set to 100 indicating the High profile.
- The `constraint_set0_flag`, `constraint_set1_flag`, `constraint_set2_flag` and `constraint_set3_flag` shall all be set to 0, and `constraint_set4_flag` and `constraint_set5_flag` shall be set to 1.
- The value of `level_idc` shall not be greater than 51 (corresponding to the level 5.1) and should indicate the lowest level to which the Bitstream conforms.

Furthermore, for a Bitstream conforming to the H.265 (HEVC) [119] Main 10 Profile, Main Tier, Level 5.1 delivered from an ITT4RT-Tx client to the ITT4RT-Rx client, the following restrictions apply:

- The `general_profile_idc` shall be set to 2 indicating the Main10 profile.
- The `general_tier_flag` shall be set to 0 indicating the Main tier.

- The value of `level_idc` shall not be greater than 153 (corresponding to the Level 5.1) and should indicate the lowest level to which the Bitstream conforms.

For 360-degree video delivery across ITT4RT clients, the following components are applicable:

- The RTP stream shall contain an HEVC or an AVC bitstream with possible presence of omnidirectional video specific SEI messages. In particular, the omnidirectional video specific SEI messages as defined in clause D.2.41 of ISO/IEC 23008-2 [119] or ISO/IEC 14496-10 [24] may be present for the respective HEVC or AVC bitstreams.
- The video elementary stream(s) shall be encoded following the requirements in the Omnidirectional Media Format (OMAF) specification ISO/IEC 23090-2 [179], clauses 10.1.2.2 (viewport-independent case) or 10.1.3.2 (viewport-dependent case) for HEVC bitstreams and clause 10.1.4.2 for AVC bitstreams. Furthermore, the general video codec requirements for AVC and HEVC in clause 5.2.2 of TS 26.114 also apply.

ITT4RT-Rx clients are expected to be able to process the VR metadata carried in SEI messages for rendering 360-degree video according to the relevant processes. Relevant SEI messages contained in the elementary stream(s) with decoder rendering metadata may include the following information for the relevant processes as per clause D.3.41 of ISO/IEC 23008-2 [119] and ISO/IEC 14496-10 [24]:

- Projection mapping information (indicating the projection format in use, e.g., Equirectangular projection (ERP) or Cubemap projection (CMP)), for the projection sample location remapping process as specified in clauses 7.5.1.3 and 5.2 of ISO/IEC 23090-2 [179]
- Region-wise packing information (carrying region-wise packing format indication, any coverage restrictions or padding/guard region information in the packed picture), for the inverse processes of the region-wise packing as specified in clauses 7.5.1.2 and 5.4 of ISO/IEC 23090-2 [179]
- Sphere rotation information (indicating the amount of sphere rotation, if any, applied to the sphere signal before projection and region-wise packing at the encoder side), for the coordinate axes conversion process as specified in clause 5.3 of ISO/IEC 23090-2 [179]
- Frame packing arrangement (indicating the frame packing format for stereoscopic content), for the processes as specified in D.3.16 of ISO/IEC 23008-2 [119]
- Fisheye video information (indicating that the picture is a fisheye video picture containing a number of active areas captured by fisheye camera lens), for the fisheye sample location remapping process as specified in clause D.3.41.7.5 of ISO/IEC 23008-2 [119]

The exchange of SEI messages carrying VR metadata for rendering 360-degree video or fisheye video shall be performed using bitstream-level signalling as follows.

SEI messages shall be present in the respective video elementary streams corresponding to the HEVC or AVC bitstreams carrying 360-degree video or fisheye video from the ITT4RT-Tx client to the ITT4RT-Rx client, as per ISO/IEC 23008-2 [119] or ISO/IEC 14496-10 [24]. As expressed more clearly below, the mandatory inclusion of the specific SEI messages in the bitstream by the ITT4RT-Tx client and their decoder and rendering processing by the ITT4RT-Rx client is conditional upon successful SDP-based negotiation of the corresponding 360-degree video or fisheye video capabilities.

In particular, the ITT4RT-Tx client supporting 360-degree video for viewport-independent processing shall signal in the bitstream the equirectangular projection SEI message (`payloadType` equal to 150) to the ITT4RT-Rx client, with the `erp_guard_band_flag` set to 0.

If viewport-dependent processing (VDP) capability is successfully negotiated by the ITT4RT-Tx client and ITT4RT-Rx client for the exchange of 360-degree video, then, the ITT4RT-Tx client shall signal in the bitstream to the ITT4RT-Rx client either:

- the equirectangular projection SEI message (`payloadType` equal to 150) with the `erp_guard_band_flag` set to 0, or
- the cubemap projection SEI message (`payloadType` equal to 151).

In order to optimize the spatial resolution of specific viewports, the ITT4RT-Tx client and ITT4RT-Rx client may negotiate the use of region-wise packing as part of the exchange of 360-degree video. If this is the case, the region-wise

packing SEI message (payloadType equal to 155) shall also be signalled by the ITT4RT-Tx client to the ITT4RT-Rx client in the bitstream.

If stereoscopic video capability is successfully negotiated by the ITT4RT-Tx client and ITT4RT-Rx client as part of the exchange of 360-degree video, then the frame packing arrangement SEI message (payloadType equal to 45) shall also be signalled by the ITT4RT-Tx client to the ITT4RT-Rx client in the bitstream, with the following restrictions:

- The value of `frame_packing_arrangement_cancel_flag` is equal to 0.
- The value of `frame_packing_arrangement_type` is equal to 4.
- The value of `quincunx_sampling_flag` is equal to 0.
- The value of `spatial_flipping_flag` is equal to 0.
- The value of `field_views_flag` is equal to 0.
- The value of `frame0_grid_position_x` is equal to 0.
- The value of `frame0_grid_position_y` is equal to 0.
- The value of `frame1_grid_position_x` is equal to 0.
- The value of `frame1_grid_position_y` is equal to 0.

Furthermore, ITT4RT-Tx clients supporting 360-degree fisheye video shall signal the fisheye video information SEI message (payloadType equal to 152) to the ITT4RT-Rx clients in the bitstream.

The bitstream delivered from an ITT4RT-Tx client to the ITT4RT-Rx client shall contain the corresponding SEI message and ITT4RT-Rx client shall process the VR metadata carried in the signalled SEI message(s) for rendering 360-degree video (provided the successful SDP-based negotiation of the corresponding 360-degree video or fisheye video capabilities associated with the SEI messages).

Y.4 Immersive Voice/Audio Support

ITT4RT-Rx and ITT4RT-Tx clients in terminals offering speech communication shall support super-wideband communication as specified in clause 5.2.1.

Y.5 Overlay Support

An overlay is defined as a piece of visual media, rendered over omnidirectional video or on the viewport. An ITT4RT-Tx client supporting the 'Overlay' feature shall add real-time overlays on top of a 360-degree background and offer this capability in the SDP as part of the the initial offer-answer negotiation. An overlay in ITT4RT is characterized by the overlay source and its rendering configuration. ITT4RT supports both 2D and spherical overlays.

An overlay source specifies the image or video to be used as the content for the overlay. The overlay source is a bitstream that may be delivered as an overlay video encoded and delivered as an RTP stream.

ITT4RT clients supporting the 'Overlay' feature have the following requirements:

- an ITT4RT-Tx client may be capable to provide a bitstream consisting of an overlay with a video source that conforms to the requirements of Y.3 or the video bitstream requirements of MTSI.
- an ITT4RT-Tx client may be capable of providing HEVC encoded images/image sequence that conform to HEVC bitstream requirements of Y.3. The signalling for such a stream shall comply to 6.2.11 and 7.4.8.
- an ITT4RT-Rx client may be capable to decode and render a 360-degree video bitstream and further decode and render one or more overlay streams on top of the 360-degree video.

The rendering configuration defines how the overlay is to be rendered in the spherical scene. It is possible for an overlay to be delivered without any rendering configuration, in which case, the ITT4RT-Rx client should render the

content as per the default configuration described in section Y.6.4.3. Alternatively, the rendering configuration for one or more overlays may be negotiated during session establishment to ensure uniform operation for different receivers.

Y.6 Media configuration

Y.6.1 General

Based on the architecture described in clause Y.2, an SDP framework for immersive video and immersive voice/audio exchange for ITT4RT is presented to negotiate codec support, SEI messages for decoder rendering metadata, as well as RTP/RTCP signaling necessary for viewport dependent processing.

The SDP attributes *3gpp_360video*, *3gpp_fisheye*, *3gpp_overlay* shall be used to indicate respectively a 360-degree projected video stream, a 360-degree fisheye video stream, and a spherical overlay. ITT4RT-Tx clients that support both 360-degree projected video and 360-degree fisheye video may include both *3gpp_360video* and *3gpp_fisheye* attributes as alternatives in the SDP offer, but an ITT4RT-Rx client willing to receive 360-degree video shall include only one attribute (either *3gpp_360video* or *3gpp_fisheye*, based on support or selection) in the SDP answer. The *3gpp_overlay* attribute may be included in the SDP answer independent on whether projected or fisheye video is selected, since spherical overlays are applicable to both types of 360-degree video streams. The detailed definition and usage of these SDP attributes are presented in the clauses below.

Y.6.2 Main 360-degree video

Y.6.2.1 General

A new SDP attribute *3gpp_360video* is defined with the following ABNF syntax:

```
att-field =          "3gpp_360video"
att-value =          PT [optparams]
PT =                 1*DIGIT
optparams =          [SP FOV] [SP FOV_CENTER] [SP "Stereo"] [SP VDP] [SP viewportfb_trigger]
FOV =                "fov=" 1*(fovset)
fovset =             "[x=" azimuthrange ",y=" elevationrange "]"
azimuthrange =      deg0to360
elevationrange =    deg0to180

FOV_CENTER =        "fov_center=[x=" centerazimuth ",y=" centerelevation "]"
centerazimuth =     degminus180to180
centerelevation =   degminus90to90

VDP =               "VDP" [SP SLVL] [SP Projection] [SP PPM] SP viewport_ctrl SP viewport_size
SLVL =              "VL"/"VL,SL"/"SL"
Projection =        "projection=" proj-type *(", " proj-type)
PPM =               "ppm=" ppm-list
viewport_ctrl =     "viewport_ctrl=" vc-value *2(", " [SP] vc-value)
viewport_size =     "viewport=" azimuthrange "x" elevationrange
;sub-rules of Projection
proj-type =         "ERP"/"CMP"
;sub-rules of PPM
ppm-list =          ppm-value *("/" ppm-value)
ppm-value =         "1"/"2"/packing
packing =           "[" PPWHQ ", " PPHHQ ", " TRHQ ", " PPWLQ ", " PPHLQ ", " TRLQ "]"
PPWHQ =             pos-integer
PPHHQ =             pos-integer
TRHQ =              transform-value
PPWLQ =             pos-integer
PPHLQ =             pos-integer
TRLQ =              transform-value
transform-value =   "0"/"1"/"2"/"3"/"4"/"5"/"6"/"7"
; transform values as per Table Y.6.1

;sub-rules of viewport_ctrl
vc-value =          "0"/"1"/"2"
; viewport control values as per Table Y.6.2

viewportfb_trigger =D_spherical/("[ D_azimuth ", " D_elevation "]")
D_spherical =       deg0to180minus1
D_azimuth =         degminus180to180
```

```

D_elevation =      degminus90to90

;generic sub-rules
deg0to360 =        "0"
                  / POS-DIGIT *6DIGIT
                  / "1" 7DIGIT
                  / "2" ("0"/"1"/"2") 6DIGIT
                  / "23" ("0"/"1"/"2"/"3"/"4") 5DIGIT
                  / "235" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7"/"8") 4DIGIT
                  / "2359" ("0"/"1") 3DIGIT
                  / "23592" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7"/"8") 2DIGIT
                  / "235929" ("0"/"1"/"2"/"3"/"4"/"5") DIGIT
                  / "23592960"
                  ; 0 to 23 592 960, inclusive
deg0to180 =        "0"
                  / POS-DIGIT *6DIGIT
                  / "10" 6DIGIT
                  / "11" ("0"/"1"/"2"/"3"/"4"/"5"/"6") 5DIGIT
                  / "117" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7"/"8") 4DIGIT
                  / "1179" ("0"/"1"/"2"/"3"/"4"/"5") 3DIGIT
                  / "11796" ("0"/"1"/"2"/"3") 2DIGIT
                  / "117964" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7") 1DIGIT
                  / "11796480"
                  ; 0 to 11 796 480, inclusive
degminus180to180 = "0"
                  / ["-"] POS-DIGIT *6DIGIT
                  / ["-"] "10" 6DIGIT
                  / ["-"] "11" ("0"/"1"/"2"/"3"/"4"/"5"/"6") 5DIGIT
                  / ["-"] "117" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7"/"8") 4DIGIT
                  / ["-"] "1179" ("0"/"1"/"2"/"3"/"4"/"5") 3DIGIT
                  / ["-"] "11796" ("0"/"1"/"2"/"3") 2DIGIT
                  / ["-"] "117964" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7") 1DIGIT
                  / "-11796480"
                  ; -11 796 480 to 11 796 479, inclusive
degminus90to90 =  "0"
                  / ["-"] POS-DIGIT *5DIGIT
                  / ["-"] ("1"/"2"/"3"/"4") 6DIGIT
                  / ["-"] "5" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7") 5DIGIT
                  / ["-"] "58" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7"/"8") 4DIGIT
                  / ["-"] "589" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7") 3DIGIT
                  / ["-"] "5898" ("0"/"1") 2DIGIT
                  / ["-"] "58982" ("0"/"1"/"2"/"3") DIGIT
                  / ["-"] "5898240"
                  ; -5 898 240 to 5 898 240, inclusive
deg0to180minus1 = "0"
                  / POS-DIGIT *6DIGIT
                  / "10" 6DIGIT
                  / "11" ("0"/"1"/"2"/"3"/"4"/"5"/"6") 5DIGIT
                  / "117" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7"/"8") 4DIGIT
                  / "1179" ("0"/"1"/"2"/"3"/"4"/"5") 3DIGIT
                  / "11796" ("0"/"1"/"2"/"3") 2DIGIT
                  / "117964" ("0"/"1"/"2"/"3"/"4"/"5"/"6"/"7") 1DIGIT
                  ; 0 to 11 796 479, inclusive
pos-integer =     POS-DIGIT * DIGIT
POS-DIGIT =       %x31-39          ; 1-9

```

NOTE: If the SDP negotiations become too complex, defining profiles can be considered.

The semantics of the above attribute and parameters is provided below. Unsupported parameters of the 3gpp_360video attribute may be ignored. The payload type is the RTP payload type number of the media stream associated with the 3gpp_360video attribute.

An ITT4RT client supporting the 3gpp_360video attribute shall support the following procedures:

- when sending an SDP offer, the ITT4RT client includes the 3gpp_360video attribute in the media description for video in the SDP offer,
- when sending an SDP answer, the ITT4RT client includes the 3gpp_360video attribute in the media description for video in the SDP answer if the 3gpp_360video attribute was received in an SDP offer,
- after successful negotiation of the 3gpp_360video attribute in the SDP, for the video streams based on the HEVC or AVC codec, the ITT4RT clients exchange an RTP-based video stream containing an HEVC or AVC bitstream with omnidirectional video specific SEI messages as defined in Annex Y.3.

An ITT4RT client supporting the `3gpp_360video` attribute supporting use of viewport-dependent processing (VDP) shall include the `VDP` parameter in the SDP offer and answer. Depending on the value indicated by the `VDP` parameter, the ITT4RT client shall further support the following procedures:

- the RTCP feedback (FB) message described in Annex Y.7.2 of type 'Viewport' to carry requested viewport information during the RTP streaming of media (signalled from the ITT4RT-Rx client to the ITT4RT-Tx client).

An ITT4RT client shall not include `VDP` parameter in the SDP answer if the SDP offer contains the `3gpp_360video` attribute without the `VDP` parameter.

An ITT4RT-Tx client that supports VDP may use viewport margins to maintain consistent quality during small head motion and also to reduce the need for frequent viewport updates. Viewport margins can be extended on all or some sides of the viewport and may be at the same quality (or resolution) as the viewport or at a quality (or resolution) lower than the viewport but higher than the background. Viewport margins may be extended around the viewport evenly or unevenly depending on head motion or network quality.

Y.6.2.2 Projection

An ITT4RT- client supporting the `3gpp_360video` attribute with VDP supporting projection may include the `Projection` parameter indicating the types of projection (e.g. ERP, CMP) it prefers (in the order of preference) in the SDP. An ITT4RT client may respond to an SDP offer with multiple options indicated in the `Projection` parameter with the agreed option. An ITT4RT-Tx client is not required to provide the preferred form of projection indicated by an ITT4RT-Rx client but may do so when possible.

The ABNF syntax is defined as follows:

Y.6.2.3 Field Of View (FOV)

An ITT4RT-Tx client may support sending a limited 360-degree video.

- i. An ITT4RT-Tx client supporting the `3gpp_360video` attribute capable of sending a limited 360-degree video shall include the parameter `FOV` in its SDP offer to indicate the *cfov* (*Capture FoV*) as the extent (range) of the 360-degree video with respect to the unit sphere. The range is expressed in units of 2^{-16} degrees with an *x* parameter for azimuth range and a *y* parameter for elevation range, sent as a comma-separated tuple. The values for azimuth range shall be in the range of 0 to $360 * 2^{16}$ (i.e., 23 592 960), inclusive, and the values for elevation range shall be in the range of 0 to $180 * 2^{16}$ (i.e., 11 796 480), inclusive. In the absence of *cfov*, the default value of *x* and *y* are 360 and 180 degrees, respectively.
- ii. An ITT4RT-Rx client supporting the `3gpp_360video` attribute capable that wants to receive a limited 360-degree video shall include the parameter `FOV` in its SDP offer/answer to indicate the *pfov* (*Preferred FoV*), where *pfov* \leq *cfov* in one or both the *x* and *y* dimensions when *cfov* is known. The *pfov* range is expressed in units of 2^{-16} degrees with an *x* parameter for azimuth range and a *y* parameter for elevation range, sent as a comma-separated tuple. The values for azimuth range shall be in the range of 0 to $360 * 2^{16}$ (i.e., 23 592 960), inclusive, and the values for elevation range shall be in the range of 0 to $180 * 2^{16}$ (i.e., 11 796 480), inclusive. In the absence of *pfov*, the default value of *x* and *y* are 360 and 180 degrees, respectively.
- iii. An ITT4RT-Tx client that has received an SDP offer from an ITT4RT-Rx client with the parameter `FOV` shall include in its SDP answer the parameter `FOV` to indicate the range of the 360-degree video it will provide. The value is the same as the `FOV` in the SDP offer or different based on the ITT4RT-Tx client capabilities.

An ITT4RT client supporting the `3gpp_360video` attribute with the `FOV` parameter may include the parameter `FOV_CENTER` in the SDP. `FOV_CENTER` is expressed as a comma-separated tuple (*x*,*y*), where *x* is the azimuth (in units of 2^{-16} degrees) and *y* is the elevation (in units of 2^{-16} degrees) with respect to the global coordinates such that the range defined `FOV` bypasses through the coordinates defined by `FOV_CENTER`. The values for azimuth shall be in the range of $-180 * 2^{16}$ (i.e., -11 796 480) to $180 * 2^{16} - 1$ (i.e., 11 796 479), inclusive, and the values for elevation shall be in the range of $-90 * 2^{16}$ (i.e., -5 898 240) to $90 * 2^{16}$ (i.e., 5 898 240), inclusive. The *imageattr* attribute indicates the resolution of the delivered content based on the *cfov* and *pfov* options.

Y.6.2.4 Picture Packing

An ITT4RT client supporting mixed-quality tiled encoding, mixed-resolution tiled encoding and/or a 360-degree low-quality background frame-packed with an overlapping high-quality viewport shall include the PPM parameter in the 3gpp_360video attribute of the SDP offer.

A list of all supported options as defined by ppm-list in Annex Y.6.2.1 shall be included in the SDP offer, where:

- A ppm-value of 1 indicates mixed-quality tiled encoding
- A ppm-value of 2 indicates mixed-resolution tiled encoding
- A ppm-value set to the comma-separated list 'packing' indicates low-quality viewport-independent background 360-degree video frame-packed with a high-quality viewport (possibly with margins) such that the two regions have overlapping content

An ITT4RT client that receives an SDP offer with a ppm-list of more than one ppm-value shall include only one preferred/supported ppm-value in the SDP answer. An ITT4RT-Rx client that includes the PPM parameter in its SDP offer with the ppm-value set to 'packing' (as defined above in the PPM syntax) shall set all values of the 'packing' to zero. An ITT4RT-Tx client that receives the PPM parameter in an SDP offer with the ppm-value set to 'packing' (as defined above in the PPM syntax) shall set all values of the 'packing' appropriately in the response.

Tiled encoding may be used to deliver the full 360-degree video or a high-quality video which includes the viewport and may include viewport margins.

The ppm-value 'packing' consists of the following six fields:

- PPWHQ defines packed_picture_width of the high-quality region in pixels
- PPHHQ defines packed_picture_height of the high-quality region in pixels
- TRHQ defines transform operations applied on the high-quality region.
- PPWLQ defines packed_picture_width of the low-quality region in pixels
- PPHLQ defines packed_picture_height of the low-quality region in pixels
- TRLQ defines transform operations applied on the low-quality region

The transform operations have a value of 0-7 as defined in Table Y.6.1:

Table Y.6.1: Transform values

| Transform value | Transform operation |
|-----------------|---|
| 0 | no transform |
| 1 | mirrored horizontally |
| 2 | rotation by 180 degrees (counter-clockwise) |
| 3 | rotation by 180 degrees (counter-clockwise) before mirroring horizontally |
| 4 | rotation by 90 degrees (counter-clockwise) before mirroring horizontally |
| 5 | rotation by 90 degrees (counter-clockwise) |
| 6 | rotation by 270 degrees (counter-clockwise) before mirroring horizontally |
| 7 | rotation by 270 degrees (counter-clockwise) |

An ITT4RT-Rx client shall render the high-quality viewport region where these two regions are overlapping. The PPM parameter for defining the HQ and LQ regions should be used when the information remains constant during the session. When the packed regions are not overlapping, the high-quality and low-quality regions do not need to be explicitly defined and SEI messages for region-wise packing may be used instead of the SDP PPM parameter.

NOTE: The size of the viewport and fov attributes define the size of projected regions. [It should be considered if the two values should be included explicitly here].

Y.6.2.5 Viewport Control

An ITT4RT client that supports the 3gpp_360video with VDP shall in its SDP offer include the parameter `viewport_ctrl` with one or more of the following control options:

- *device_controlled* if ITT4RT-Tx client will provide VDP based on the requested viewport indicated by the RTCP feedback (FB) message type ‘Viewport’ sent by the corresponding ITT4RT-Rx client.
- *recommended_viewport* if ITT4RT-Tx client will provide VDP with the help of a recommendation/prediction engine.
- *presenter_viewport* if ITT4RT-Tx will provide VDP based on the viewport of an ITT4RT-Rx client other than the one the SDP offer is being sent to.

Table Y.6.2 provides a mapping between viewport control values and viewport control options.

Table Y.6.2: Viewport control values

| Viewport control value | Viewport control option |
|------------------------|-----------------------------------|
| 0 | <code>device_controlled</code> |
| 1 | <code>recommended_viewport</code> |
| 2 | <code>presenter_viewport</code> |

Multiple options are provided as a comma-separated list. An ITT4RT client that receives an SDP offer with multiple `viewport_ctrl` options may include its preferred `viewport_ctrl` option in the SDP answer. If no options are given in the answer, the sender shall use the first option in the list. If the `recommended_viewport` is successfully negotiated as `viewport_ctrl`, the ITT4RT-Rx client should not use viewport prediction when sending the RTCP feedback (FB) message type ‘Viewport’ to avoid any conflicts with the prediction engine of the ITT4RT-Tx client.

Y.6.2.6 Overlay and 360-degree video

An ITT4RT client that sends an SDP message with at least one 360-degree video/audio and at least one overlay shall include in SDP the attribute `itt4rt_group` before any media lines. The `itt4rt_group` attribute is used to group 360-degree media and overlay media using the mid attribute and the syntax for the SDP attribute is:

```
a=itt4rt_group: <group-1> / ... / <group-N>
```

where `<group-X>` shall include at least one mid associated with 360-degree media and at least one mid associated with an overlay as defined by the mid attribute in the corresponding media description.

The ABNF syntax for this attribute is the following:

```
att-field = "itt4rt_group"
att-value = rest-group *[" / " rest-group]
rest-group = 2*(SP identification-tag)
            ; identification-tag is defined in RFC 5888
```

An ITT4RT-Tx client and an ITT4RT-Rx client may negotiate the overlays that can be associated with the 360-degree video offered by the ITT4RT-Tx client using the `itt4rt_group` attribute. An ITT4RT client shall indicate in an offer the overlays to be grouped with the 360-degree video using the `itt4rt_group` attribute. The overlays that are acceptable shall be retained in the answer and the ones that are not acceptable shall be removed. An ITT4RT-Tx client may offer overlay configuration options using the `3gpp_overlay` attribute based on the list of media lines (i.e., potential overlay sources) provided in the `itt4rt_group` attribute in an SDP offer initiated by an ITT4RT-Rx client. The `3gpp_overlay` attribute is offered in an SDP renegotiation.

The order of the media included in the `itt4rt_group` indicates the synchronization source with the first media always being the synchronization anchor when synchronization is required.

NOTE: 2D video received from an ITT4RT-Rx clients may be offered as an overlay by the ITT4RT MRF to other ITT4RT-Rx clients. The ITT4RT MRF (acting as the ITT4RT-Tx client) would be the source of overlay media in this case.

Y.6.2.7 Viewport Size

An ITT4RT client that includes the `3gpp_360video` with the VDP parameter shall also include in SDP the parameter `viewport_size` to indicate the size of the device viewport using the azimuth and elevation ranges expressed in degrees. An ITT4RT-Tx client may include the `viewport_size` of the ITT4RT-Rx client when this is known (e.g., in response to an SDP offer from an ITT4RT-Rx client) or include "`viewport=0x0`" and the value can be ignored by the ITT4RT-Rx client.

NOTE: The viewport size defines the size of the viewport of the ITT4RT-Rx client UE. An ITT4RT-Tx client provides VDP based on this viewport size. The capture and preferred FOV, on the other hand, defines the range of the 360-degree content, which should be larger than the viewport size and can be negotiated even for viewport-independent processing.

Y.6.2.8 Viewport-only VDP and Image attributes

An ITT4RT-Tx client supporting VDP may deliver only the viewport or viewport with viewport margins and not the full captured/preferred field-of-view of the 360-degree video. If the viewport region (with or without a viewport margin) is extracted from a projected picture (e.g., ERP), the resolution would change depending on where the viewport is located on the picture. To avoid this, the ITT4RT-Tx client may rotate the desired viewport region to the centre of the ERP before cropping it to the desired size as indicated by `imageattr`. The delivered bitstream shall contain the rotation SEI and the region-wise packing SEI message if the ITT4RT-Rx client is expected to do sphere-locked rendering by reversing the rotation of the received image before rendering.

An ITT4RT client may support viewport-locked VDP for delivering the viewport region only. A viewport-locked VDP bit stream should include only the viewport region and should not include rotation SEI messages. An ITT4RT client that supports viewport-locked VDP shall include in its SDP offer the parameter `SLVL` as defined in Annex Y.6.2.1.

The value "SL" refers to sphere-locked rendering, which requires the receiver to render the received picture according to the global coordinate axes. The value "VL" refers to viewport-locked rendering, which require the receiver to render the received picture such that the center of the received picture is aligned to the center of the current viewport.

An ITT4RT client that supports only viewport-locked VDP shall include "VL" in its SDP offer. An ITT4RT client that supports both viewport-locked VDP and a sphere-locked type of VDP shall include "VL,SL". An ITT4RT client that receives an SDP offer with "VL" shall include it in its response if it supports viewport-locked VDP and chooses to use it. An ITT4RT client that receives an SDP offer with "VL" shall remove the VDP parameter in its response if it does not support viewport-locked VDP or does not wish to use it; the 360-degree video is then delivered using viewport-independent processing. An ITT4RT client that receives an SDP offer with "SL" shall include it in its response if it supports sphere-locked VDP and chooses to use it. An ITT4RT client that receives an SDP offer with "SL" shall remove the VDP parameter in its response if it does not support sphere-locked VDP or does not choose to use it; the 360-degree video is then delivered using viewport-independent processing. An ITT4RT client that receives an SDP offer with "VL,SL" shall include either "SL" or "VL" in the SDP response based on its preferred mode. Alternatively, if it does not support nor choose either VL or SL it shall remove the VDP from the `3gpp_360video` attribute in the response.

Y.6.2.9 Viewport Feedback Trigger

An ITT4RT client include the parameter `viewportfb_trigger` in the `3gpp_360video` attribute to define the minimum view port change to initiate an early or event-based RTCP feedback. The syntax for `viewportfb_trigger` is defined in Annex Y.6.2.1. `D_azimuth` and `D_elevation` are the minimum number of degrees that the viewport may change in the horizontal or vertical direction, respectively. The value `D_spherical` is the minimum spherical distance in degrees between the center of the old and the new viewport. The values for `D_spherical` shall be in the range 0 to $180 * 2^{16} - 1$ (i.e., 11 796 479). Spherical distance between the centre of a first viewport (x_1, y_1) and second viewport (x_2, y_2) , is calculated as:

$$\alpha = \sin^2\left(\frac{y_1 - y_2}{2}\right) + \cos(y_1) \times \cos(y_2) \times \sin^2\left(\frac{x_1 - x_2}{2}\right)$$

$$c = 2 \times \arctan2(\sqrt{\alpha}, \sqrt{1 - \alpha})$$

where x_1 and x_2 is the azimuth in radians and y_1 and y_2 is the elevation in radians. The value c is in radians and must be converted to degrees for use with D .

The viewport feedback trigger value is estimated by the ITT4RT-Tx client based on the viewport margin configuration it intends to use. An ITT4RT-Rx client supporting RTCP viewport feedback shall use periodic RTCP viewport feedback. The frequency of the periodic feedback should be such that it does not exceed the allocated RTCP bandwidth as defined in RFC 4585. An ITT4RT-Rx client may use immediate/early RTCP feedback in addition to the periodic feedback as long as the allocated RTCP bandwidth requirements are met. An ITT4RT-Tx may define a viewport feedback trigger value for an early/immediate feedback and signal this value to the ITT4RT-Rx client in the SDP. The ITT4RT-Tx client should select a threshold value that is suitable for the margin configuration that it intends to use for that stream. The threshold value should be defined within the viewport margin region such that the ITT4RT-Tx client would update the high-quality region (viewport and viewport margin) if the viewport breaches this threshold.

If an ITT4RT-Rx client does not have the capability to provide an RTCP viewport feedback at the viewport feedback threshold value provided by the ITT4RT-Tx client in an SDP offer, it may respond with the the minimum threshold value it can support. The ITT4RT-Tx client may adjust its viewport margin configuration based on the threshold value in the answer. If an ITT4RT-Rx client only supports periodic feedback, it shall remove the `viewportfb_trigger` parameter from the response.

An ITT4RT-Rx client that supports a viewport feedback trigger shall include the parameter `viewportfb_trigger` with the minimum threshold value it can support in an SDP offer. The ITT4RT-Tx client may remove the parameter if it does not support this value or respond with an acceptable value that is equal or higher than the one in the ITT4RT-Rx's offer.

If both sides acknowledge the support of `viewportfb_trigger`, the ITT4RT-Rx client shall use event-driven/early viewport feedback in addition to periodic feedback. If `viewportfb_trigger` is not defined by the ITT4RT-Tx client, the ITT4RT-Rx client may still use immediate/early feedback. An ITT4RT-Rx client may use the velocity of the viewport during head motion and the viewport margin (if known) to trigger an immediate feedback. Alternatively, it may use the spherical distance between the viewport in the last feedback and the current viewport to trigger an immediate feedback. The spherical distance can be selected based on viewport margins (if known). An ITT4RT-Rx client may suppress an immediate/early feedback if the time to the next periodic viewport feedback is less than an application-defined threshold.

Y.6.3 Still Background

Still image backgrounds may be supported by ITT4RT clients. The format and signaling shall follow the static image format and signaling as defined in clauses 5.2.4, 6.2.11, and 7.4.8. An ITT4RT-Tx client should send the image/image sequence as a video bitstream if still images are not supported.

The signaling in clause Y.6.2 shall apply to indicate that the still background is 360 degree. The `3gpp_360video` or `3gpp_fisheye` attribute shall be used for that purpose.

Y.6.4 Overlays

Y.6.4.1 General

ITT4RT clients supporting the 'Overlay' feature may define an overlay source and overlay configuration in the SDP. An ITT4RT client that supports overlays shall support a video or image stream indicated by a media line in the SDP as the source of an overlay.

Y.6.4.2 Visual Media

Any visual media that is defined with the 'm=video' line, includes the attribute *mid* and does not have the attribute *a=3gpp_360video* in the SDP may be rendered as an overlay by an ITT4RT-Rx client. An ITT4RT client shall include the attribute *mid* in the overlay media description. If an overlay is to be associated with a particular overlay configuration, the *mid* shall be used to associate the overlay media description to the the overlay configuration, which is later described in clause Y.6.4.3.

Visual media may consist of an encoded video bitstream or HEVC encoded images/image sequences, both of which are defined with the 'm=video' line in SDP. ITT4RT clients that support images shall use the "a=imageseq" attribute as defined in clause 6.2.11.

Y.6.4.3 Overlay Configuration

Y.6.4.3.1 General

ITT4RT clients may support the following types of rendering for overlays, as defined in the OMAF specification [179] clause 7.14.

- viewport-relative overlay, specifying that the overlay is displayed on a rectangular area at an indicated position relative to the top-left corner of the viewport;
- [sphere-relative projected omnidirectional overlay, specifying that the overlay is displayed on a sphere surface at an indicated position within or on the unit sphere];
- sphere-relative 2D overlay, specifying that the overlay is displayed on a plane at an indicated position within the unit sphere.

An ITT4RT client supporting overlays may include in its SDP media description the attribute `3gpp_overlay` to define one or more parameters for configuring the rendering properties of an overlay. The `3gpp_overlay` attribute has the following syntax:

```
att-field = "3gpp_overlay"
att-value = [free_overlay SP] overlay_id SP type SP overlay_config
           [SP overlay_info] [SP overlay_overlap_flag]

free_overlay =          "free_overlay=" BIT
overlay_id =           token
                       ; token is defined in RFC 4566
type =                 BIT
overlay_config =       (sph_rel_overlay_config / vp_rel_overlay_config)
overlay_info =         "overlay_info=" 5BIT
overlay_overlap_flag = "overlap=" BIT

;sub-rules for sphere_relative_overlay_config
sph_rel_overlay_config = [" Overlay_azimuth "," Overlay_elevation "," Overlay_tilt
                        "," Overlay_azimuth_range "," Overlay_elevation_range
                        "," Overlay_rot_yaw "," Overlay_rot_pitch "," Overlay_rot_roll
                        "," region_depth_minus1 "," timeline_change_flag "]"
Overlay_azimuth =      degminus180to180
Overlay_elevation =   degminus90to90
Overlay_tilt =         degminus180to180
Overlay_azimuth_range = deg0to360
Overlay_elevation_range = deg0to180
Over_rot_yaw =        degminus180to180
Over_rot_pitch =      degminus90to90
Over_rot_roll =       degminus180to180
region_depth_minus1 = pctin2tominus16
timeline_change_flag = BIT

;sub-rules for viewport_relative_overlay_config
vp_rel_overlay_config = [" Overlay_rect_left_percent "," Overlay_rect_top_percent
                        "," Overlay_rect_width_percent "," Overlay_rect_height_percent
                        "," Relative_disparith_flag "," Disparity
                        "," media_alignment "," layering_order "," opacity
                        "," overlay_priority "]"
Overlay_rect_left_percent = pctIn2ToMinus16
Overlay_rect_top_percent =  pctIn2ToMinus16
Overlay_rect_width_percent = pctIn2ToMinus16
Overlay_rect_height_percent = pctIn2ToMinus16
Relative_disparith_flag =   BIT
Disparity =                 (Disparity_in_percent / Disparity_in_pixels)
Disparity_in_percent =      pctIn2ToMinus16
Disparity_in_pixels =       integer
media_alignment =           "0"/pos-integer
layering_order =            pos-integer
opacity =                   "0"/pos-integer
overlay_priority =          pos-integer

;generic sub-rules
integer =                   "0"/["-"]pos-integer
pctin2tominus16 =          "0"
                           / POS-DIGIT *4DIGIT
                           / "6" ("0"/"1"/"2"/"3"/"4") 3DIGIT
```

```

/ "65" ("0"/"1"/"2"/"3"/"4") 2DIGIT
/ "655" ("0"/"1"/"2") 1DIGIT
/ "6553" ("0"/"1"/"2"/"3"/"4"/"5")
; 0 to 65 535, inclusive
; pos-integer, degminus180to180, degminus90to90, deg0to360 and deg0to180 are from Y.6.2.1

```

The `3gpp_overlay` attribute is included as part of the 360-degree video media description. More than one `3gpp_overlay` attribute may exist as part of the media description if more than one overlay is to be configured. The `overlay_id` in `3gpp_overlay` is set to the mid of the media description of the overlay source for which the configuration is provided. An ITT4RT-Tx client may include more than one `3gpp_overlay` attributes with the same non-zero `overlay_id` (associated with an overlay source) in an SDP offer. The overlay configurations are listed in order of preference. An ITT4RT-Rx client that receives an SDP offer with multiple `3gpp_overlay` attributes with the same non-zero `overlay_id` shall reply with only one acceptable `3gpp_overlay` line for one acceptable overlay source. If an overlay source is rejected, all `3gpp_overlay` attribute lines with the `overlay_id` associated with that source are excluded in the SDP answer.

An ITT4RT-Tx client may set the `free_overlay` flag to indicate that no media source is associated with this overlay configuration by the ITT4RT-Tx client and the corresponding overlay region in the video may be used by the ITT4RT-Rx client to overlay any media including external media sources (e.g., a text notification from an external application, an external video overlay or a source describing an occlude-free region that any overlay on top would not result in any significant loss of information). The `overlay_id` in `3gpp_overlay` in case of overlays with the `free_overlay` flag with a value of 1 shall not be equal to any mid in the SDP media description. In case of `overlay_id` which is not associated with an overlay source (to identify an external source overlay and/or an occlude-free region), the `free_overlay` flag shall be included with a value equal to 1. If the `free_overlay` flag is not present or the `free_overlay` flag has a value of 0, the `overlay_id` shall be a value which corresponds to at least one mid of the media description of the overlay source. An ITT4RT-Tx client may include more than one `3gpp_overlay` attributes with the same non-zero `overlay_id` and with the `free_overlay` flag set to 1 in an SDP offer. The implementation details for use of overlay region with the `free_overlay` flag set to 1 is left to the application.

An ITT4RT-Rx client that sends an SDP offer to receive 360-degree media and overlay source media shall not include the `3gpp_overlay` attribute. An ITT4RT-Tx client that receives such an offer may initiate an SDP renegotiation offer to configure the overlays for the overlay sources as indicated by the `itt4rt_group` attribute.

An ITT4RT client that receives an SDP offer including overlay sources without the `3gpp_overlay` attribute may decline the offer and send a renegotiation offer with the `3gpp_overlay` attribute.

The `type` shall have the value '0' for viewport-relative overlays and '1' for sphere-relative overlays. Depending on the value of `type`, the `3gpp_overlay` attribute may further include the corresponding configuration information `sph_rel_overlay_config` (`type` = '1') or the `vp_rel_overlay_config` (`type` = '0').

A set of flags describing interactivity of the overlay may be included in the optional `overlay_info` parameter defined in Annex Y.6.4.3.4.

NOTE 1: Other overlay definitions in OMAF are not excluded from ITT4RT. Which overlay definition(s) from OMAF are adopted for overlays in ITT4RT is currently TBD. Optional user interactivity flags such as for defining moveability, resizing of the overlay may be added later to the `3gpp_overlay` parameter.

NOTE 2: There should be a default configuration for overlay when explicit configuration is not provided to ensure that multiple receivers have similar experience.

Y.6.4.3.2 Sphere-relative Overlay Configuration

An ITT4RT client supporting the `3gpp_overlay` attribute to configure a sphere-relative overlay shall set parameter `type` = '1' and additionally include the parameter `sph_rel_overlay_config` defined as follows:

```

sph_rel_overlay_config = [Overlay_azimuth, Overlay_elevation, Overlay_tilt, Overlay_azimuth_range,
Overlay_elevation_range, Overlay_rot_yaw, Overlay_rot_pitch, Overlay_rot_roll, region_depth_minus1,
timeline_change_flag]

```

- `Overlay_azimuth`: Specifies the azimuth angle of the centre of the overlay region on the unit sphere in units of 2^{-16} degrees relative to the global coordinate axes.
- `Overlay_elevation`: Specifies the elevation angle of the centre of the overlay region on the unit sphere in units of 2^{-16} degrees relative to the global coordinate axes.

- `Overlay_tilt`: Specifies the tilt angle of the offered overlay region, in units of 2^{-16} degrees, relative to the global coordinate axes.
- `Overlay_azimuth_range`: Specifies the azimuth range of the region corresponding to the 2D plane on which the overlay is rendered through the centre point of the overlay region in units of 2^{-16} degrees.
- `Overlay_elevation_range`: Specifies the elevation range of the offered region corresponding to the 2D plane on which the overlay is rendered through the centre point of the overlay region in units of 2^{-16} degrees.
- `Overlay_rot_yaw`, `Overlay_rot_pitch`, and `Overlay_rot_roll` specify the rotation of the 2D plane on which the overlay is rendered. Prior to rendering the 2D plane, it may be rotated as specified by `overlay_rot_yaw`, `overlay_rot_pitch` and `overlay_rot_yaw` and placed on a certain distance as specified by `region_depth_minus1`. The rotations are relative to the coordinate system as specified in clause 5.1 of ISO/IEC 23090-2 in which the origin of the coordinate system is in the centre of the overlay region, the X axis is towards the origin of the global coordinate axes, the Y axis is towards the point on the plane that corresponds to `cAzimuth1` in Figure 7-4 of ISO/IEC 23090-2, and the Z axis is towards the point on the plane that corresponds to `cElevation2` in Figure 7-4 of ISO/IEC 23090-2. `overlay_rot_yaw` expresses a rotation around the Z axis, `overlay_rot_pitch` rotates around the Y axis, and `overlay_rot_roll` rotates around the X axis. Rotations are extrinsic, i.e., around X, Y, and Z fixed reference axes. The angles increase clockwise when looking from the origin towards the positive end of an axis. The rotations are applied starting from `overlay_rot_yaw`, followed by `overlay_rot_pitch`, and ending with `overlay_rot_roll`.
- `region_depth_minus1` - indicates the depth (z-value) of the region on which the overlay is to be rendered. The depth value is the norm of the normal vector of the overlay region. `region_depth_minus1 + 1` specifies the depth value relative to a unit sphere in units of 2^{-16} .
- `timeline_change_flag` equal to '1' specifies that the overlay content playback shall pause if the overlay is not in the user's current viewport, and when the overlay is back in the user's viewport the overlay content playback shall resume with the global presentation timeline of the content. The content in the intermediate interval is skipped. `timeline_change_flag` equal to '0' specifies that the overlay content playback shall pause if the overlay is not in the user's current viewport, and when the overlay is back in the user's viewport the overlay content playback resumes from the paused sample. This prevents loss of any content due to the overlay being away from the user's current viewport.

Y.6.4.3.3 Viewport-relative Overlay Configuration

An ITT4RT client supporting the `3gpp_overlay` attribute to configure a viewport-relative overlay shall set parameter `type = '0'` and additionally include the parameter `vp_rel_overlay_config` defined as follows:

```
vp_rel_overlay_config = [Overlay_rect_left_percent, Overlay_rect_top_percent,
Overlay_rect_width_percent, Overlay_rect_height_percent, Relative_disparity_flag, Disparity,
media_alignment, layering_order, opacity, overlay_priority]
```

- `Overlay_rect_left_percent`: Specifies the x-coordinate of the left corner of the rectangular region of the overlay to be rendered on the viewport in per cents relative to the width of the viewport. The values are indicated in units of 2^{-16} in the range of 0 (indicating 0%), inclusive, up to but excluding 65536 (that indicates 100%).
- `Overlay_rect_top_percent`: Specifies the y-coordinate of the top corner of the rectangular region of the overlay to be rendered on the viewport in per cents relative to the height of the viewport. The values are indicated in units of 2^{-16} in the range of 0 (indicating 0%), inclusive, up to but excluding 65536 (that indicates 100%).
- `Overlay_rect_width_percent`: Specifies the width of the rectangular region of the overlay to be rendered on the viewport in per cents relative to the width of the viewport. The values are indicated in units of 2^{-16} in the range of 0 (indicating 0%), inclusive, up to but excluding 65536 (that indicates 100%).
- `Overlay_rect_height_percent`: Specifies the height of the rectangular region of the overlay to be rendered on the viewport in per cents relative to the height of the viewport. The values are indicated in units of 2^{-16} in the range of 0 (indicating 0%), inclusive, up to but excluding 65536 (that indicates 100%).

NOTE 1: The size of overlay region over the viewport changes according to the viewport resolution and aspect ratio. However, the aspect ratio of the overlaid media is not intended to be changed.

- **Relative_disparity_flag**: Indicates whether the disparity is provided as **Disparity_in_percent** which is a percentage value of the width of the display window for one view (when the value is equal to 1) or as **Disparity_in_pixels** which is a number of pixels (when the value is equal to 0).
- **Disparity_in_percent**: Specifies the disparity, in units of 2^{-16} , as a fraction of the width of the display window for one view. The value may be negative, in which case the displacement direction is reversed. This value is used to displace the region to the left on the left eye view and to the right on the right eye view. This applies for the case when there is a monoscopic overlay and stereoscopic background visual media.
- **Disparity_in_pixels**: Indicates the disparity in pixels. The value may be negative, in which case the displacement direction is reversed. This value is used to displace the region to the left on the left eye view and to the right on the right eye view. This applies for the case when there is a monoscopic overlay and stereoscopic background visual media.
- **Media_alignment**: Specifies the default intended scaling of the overlay source depending on the dimensions of the specified rectangular region and the intended placement of the scaled overlay source relative to the specified rectangular region.
- **Layering_order**: Indicates the default layering order among the overlays that are relative to the viewport, and separately among each set of overlays that have the same depth. Viewport-relative overlays are overlaid on top of the viewport in descending order of **layering_order**, i.e., an overlay with a smaller **layering_order** value shall be in front of an overlay with a greater **layering_order** value. The layering order for overlays of the 360-degree video should be decided by the ITT4RT-Tx client.
- **Opacity**: Indicates an integer value that specifies the default opacity that is to be applied for the overlay and assigned by the ITT4RT-Tx client. Value 0 is fully transparent, and value 100 is fully opaque with a linear weighting between the two extremes. Values greater than 100 are reserved.
- **Overlay_priority**: Indicates which overlay should be prioritized in the case the ITT4RT-Rx client does not have enough decoding capacity to decode all overlays. A lower **overlay_priority** indicates higher priority. The value of **overlay_priority**, when present, shall be equal to 0 for overlays that are essential for displaying. More than one overlay may have the same **overlay_priority** and an ITT4RT-Rx client that does not support all overlays with the same priority may choose any subset of these.

NOTE 2: Other overlay definitions in OMAF are not excluded from ITT4RT. Which overlay definition(s) from OMAF are adopted for overlays in ITT4RT is currently TBD.

NOTE 3: For both of the above overlay types from OMAF, incorporate from the spec further details about the order of operations for overlay rendering (in particular order of translation and rotation).

Y.6.4.3.4 Overlay info parameter

The parameter **overlay_info** is a bit field consisting of flags describing the type of overlay interactivity recommended:

overlay_info= 'overlay_info='b4b3b2b1b0

- b0 = '0': changing the position of the overlay is not recommended, overlay position is fixed
b0 = '1': changing the position of the overlay is allowed, ITT4RT-Rx client may change position
- b1 = '0': switching the overlay on/off is not recommended
b1 = '1': switching the overlay on/off is allowed, ITT4RT-Rx client may switch overlay off
- b2 = '0': rotating the overlay is not recommended
b2 = '1': rotating the overlay is allowed
- b3 = '0': resizing the overlay is not recommended
b3 = '1': resizing the overlay is allowed
- b4 = '0': changing the opacity of the overlay is not recommended
b4 = '1': changing the opacity of the overlay is allowed

Y.6.4.3.5 Additional Overlay Configuration

An ITT4RT client supporting the `3gpp_overlay` attribute to configure a sphere-relative overlay or viewport-relative overlay may include the following additional parameter for overlay support:

`overlay_overlap_flag`: Indicates if the ITT4RT-Rx client is allowed to overlap overlays from the ITT4RT-Tx client. If set to 1, the ITT4RT-Rx client may overlap overlays shared by the ITT4RT-Tx client.

Y.6.4.4 Captured Content Replacement

To prevent the degradation of presentation material (e.g., slides, screen share, video, notes) that may be captured from a display (screen or projector) with a 360-degree camera, the captured content in the 360-degree video can be replaced with the original presentation material. Such replacement implies decoding the content (360-degree video and presentation material), identifying the position of the presentation material in the 360-degree video, replacing the captured presentation content at the display coordinates in the 360-degree video, and finally encoding the new 360-degree video (i.e., with the same encoding parameters as the original 360-degree video). The replacement could either be performed in the ITT4RT-Tx client in terminal which is sending the 360-degree video or in the ITT4RT-MRF.

When replacement is to be performed, the availability of the original presentation content must be signalled by the source of the content to the client performing the replacement (that is, the ITT4RT-Tx client in terminal or the ITT4RT-MRF) using the SDP attribute `"a=content:slides"` [81] which may include different content, for example slides, screen share, video, notes. The client performing the replacement shall determine an appropriate configuration for performing the content replacement in the 360-degree video, unless overlay parameters are given by the source of the original presentation content (e.g., configuration in terms of sphere-relative overlay coordinates as defined in Annex Y.6.4.3.2).

When the SDP negotiation is initiated by the ITT4RT-Tx client in terminal, the ITT4RT-Tx client in terminal shall include the attribute `"a=3gpp_360video_replacement"` in its SDP offer to indicate that the content captured in the 360-degree video can be replaced. If the ITT4RT-MRF supports content replacement and receives an SDP offer with the attribute `"a=3gpp_360video_replacement"`, then the ITT4RT-MRF shall include the attribute `"a=3gpp_360video_replacement"` in its SDP answer and shall perform content replacement.

If the ITT4RT-Tx client in terminal includes the attribute `"a=3gpp_360video_replacement"` in its SDP offer but does not receive the attribute in the SDP answer (that is, replacement is not supported in the ITT4RT-MRF) then the ITT4RT-Tx client in terminal may send the original presentation content using a different process than ITT4RT-MRF replacement (e.g., the presentation can be sent as an overlay as defined in Annex Y.6.4., or inserted into the 360-degree video by the ITT4RT-Tx client in terminal as described above).

If the ITT4RT-MRF does not receive the attribute `"a=3gpp_360video_replacement"` in an SDP offer, it shall not perform any replacement and will not include the attribute in its SDP answer.

When replacement is to be performed by the ITT4RT-MRF and the SDP negotiation is initiated by the ITT4RT-MRF, the offer sent by the ITT4RT-MRF to the ITT4RT-Tx client in terminal shall include the attribute `"a=3gpp_360video_replacement"`. If the ITT4RT-Tx client in terminal accepts the offer by the MRF to perform replacement, the ITT4RT-Tx client in terminal shall include the attribute `"a=3gpp_360video_replacement"` in the SDP answer and the ITT4RT-MRF shall perform content replacement.

If the ITT4RT-MRF does not receive the attribute `"a=3gpp_360video_replacement"` in the SDP answer of the ITT4RT-Tx in terminal (i.e., the content captured in the 360-degree video cannot be replaced), the ITT4RT-MRF shall not perform any replacement.

If the ITT4RT-MRF does not support content replacement, it shall not include the attribute `"a=3gpp_360video_replacement"` in an SDP offer, it will not perform any replacement, and the ITT4RT-Tx client in terminal may send the original presentation content using a different process (e.g., the presentation can be sent as an overlay as defined in Annex Y.6.4, or inserted into the 360-degree video by the ITT4RT-Tx client in terminal as described above). In the case that the ITT4RT-MRF does not send the attribute `"a=3gpp_360video_replacement"` in an offer, the ITT4RT-Tx client in terminal shall not send the attribute `"a=3gpp_360video_replacement"` in an answer.

After an accepted offer/answer between ITT4RT-Tx in terminal and ITT4RT-MRF with both offer and answer including the attribute `"a=3gpp_360video_replacement"`, the ITT4RT-MRF shall perform content replacement once the original presentation content is available from the source of the content and the replacement configuration is determined.

If the replacement configuration of the content is analysed and determined by the ITT4RT-Tx in terminal, the client shall include the configuration as sphere-relative overlay coordinates (defined in Annex Y.6.4.3.2) in the SDP offer/answer while negotiating the stream with the ITT4RT-MRF. If the sphere-relative overlay coordinates are not signalled in the SDP offer/answer by the ITT4RT-Tx, the ITT4RT-MRF shall analyse and determine an appropriate configuration for performing the content replacement in the 360-degree video.

The ABNF syntax for the replacement attribute is as follows:

```
att-field = "3gpp_360video_replacement"
att-value = [sph_rel_overlay_config]
```

Y.6.5 Fisheye Video

Y.6.5.1 Identifying the 360-degree fisheye video stream

The SDP attribute *3gpp_fisheye* is used to indicate a 360-degree fisheye video stream.

The semantics of the above attribute and parameters is provided below.

ITT4RT clients supporting 360-degree fisheye video shall support the *3gpp_fisheye* attribute and shall support the following procedures:

- when sending an SDP offer, the ITT4RT-Tx client includes the *3gpp_fisheye* attribute in the media description for video in the SDP offer
- when sending an SDP answer, the ITT4RT-Rx client includes the *3gpp_fisheye* attribute in the media description for video in the SDP answer if the *3gpp_fisheye* attribute was received in an SDP offer
- after successful negotiation of the *3gpp_fisheye* attribute in the SDP, the MTSI clients exchange an RTP-based video stream containing an HEVC or AVC bitstream with fisheye omnidirectional video specific SEI messages as defined in clause Y.3

ITT4RT-Tx clients that support both 360-degree projected video and 360-degree fisheye video may include both *3gpp_360video* and *3gpp_fisheye* attributes as alternatives in the SDP offer, but an ITT4RT-Rx client shall include only one attribute (either *3gpp_360video* or *3gpp_fisheye*, based on support or selection) in the SDP answer.

Y.6.5.2 360-degree fisheye video SDP attribute parameters

Media-line level parameters are defined in order to aid session establishment between the ITT4RT-Tx and ITT4RT-Rx clients for 360-degree fisheye video, as well as to describe the fisheye video stream as identified by the *3gpp_fisheye* attribute.

The syntax for the SDP attribute is:

```
a=3gpp_fisheye: <fisheye> <fisheye-img> <maxpack>
```

- Total number of fisheye circular videos at the capturing terminal.

Depending on the camera configuration of the sending terminal, the 360-degree fisheye video may be comprised of multiple different fisheye circular videos, each captured through a different fisheye lens.

- *<fisheye>*: this parameter inside an SDP offer sent by an ITT4RT-Tx client indicates the total number of fisheye circular videos output by the camera configuration at the terminal.
- Fisheye circular video static parameters.

In order to enable the quick selection of desired fisheye circular videos by the ITT4RT-Rx client during SDP negotiation, the following static parameters are defined for each fisheye circular video. These parameters are defined from the video bitstream fisheye video information SEI message as defined in ISO/IEC 23008-2 [119] and ISO/IEC 23090-2 [179].

- *<fisheye-img>* = *<fisheye-img-1>* ... *<fisheye-img-N>*
- *<fisheye-img-X>* = [*<id-X>* *<azi>* *<ele>* *<til>* *<fov>*] for $1 \leq X \leq N$ where:

- <id>: an identifier for the fisheye video.
 - <azi>, <ele>: azimuth and elevation indicating the spherical coordinates that correspond to the centre of the circular region that contains the fisheye video, in units of 2^{-16} degrees. The values for azimuth shall be in the range of $-180 * 2^{16}$ (i.e., -11 796 480) to $180 * 2^{16} - 1$ (i.e., 11 796 479), inclusive, and the values for elevation shall be in the range of $-90 * 2^{16}$ (i.e., -5 898 240) to $90 * 2^{16}$ (i.e., 5 898 240), inclusive.
 - <til>: tilt indicating the tilt angle of the sphere regions that corresponds to the fisheye video, in units of 2^{-16} degrees. The values for tilt shall be in the range of $-180 * 2^{16}$ (i.e., -11 796 480) to $180 * 2^{16} - 1$ (i.e., 11 796 479), inclusive.
 - <fov>: specifies the field of view of the lens that corresponds to the fisheye video in the coded picture, in units of 2^{-16} degrees. The field of view shall be in the range of 0 to $360 * 2^{16}$ (i.e., 23 592 960), inclusive.
- Stream packing of fisheye circular videos

Depending on the terminal device capabilities and bandwidth availability, the packing of fisheye circular videos within the stream can be negotiated between the sending and receiving terminals.

- <maxpack>: this parameter inside an SDP offer indicates the maximum supported number of fisheye videos which can be packed into the video stream by the ITT4RT-Tx client. The value of this parameter inside an SDP answer indicates the number of fisheye videos to be packed, as selected by the ITT4RT-Rx client.

The ABNF syntax for this attribute is the following:

```
att-field   = "3gpp_fisheye"
att-value  = [SP fisheye] SP fisheye-img SP maxpack

fisheye   = pos-integer

fisheye-img = 1*fisheye-img-X
fisheye-img-X = "[" "id=" idvalue "," "azi=" azivalue "," "ele=" elevalue "," "til=" tilvalue
                "," "fov=" fovvalue "]"
;sub-rules for fisheye-img-X
idvalue   = byte-string ; byte-string defined by RFC 4566
azivalue  = degminus180to180
elevalue  = degminus90to90
tilvalue  = degminus180to180
fovvalue  = deg0to360

maxpack   = pos-integer

;pos-integer, degminus180to180, degminus90to90 and deg0to360 are from Y.6.2.1
An example SDP offer is shown in table Y.6.5.2-1.
```

Table Y.6.5.2-1: Example SDP offer with 360-degree fisheye video attribute parameters

| SDP offer |
|---|
| <pre>m=video 49154 RTP/AVP 99 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:10000 b=RS:0 b=RR:2500 a=rtmpmap:99 H265/90000 a=fmtp:99 profile-id=1; level-id=93; a=3gpp_fisheye: 2 [id=1,azi=0,ele=0,til=0,fov=11796480] [id=2,azi=11796479,ele=0,til=0,fov=11796480] 2 a=sendonly</pre> |

As an example, a receiving terminal which only receives 360-degree fisheye video (and possibly sends a 2D video to the sender) replies with an SDP answer containing only the selected fisheye videos equal to the number as selected by the value of *maxpack* in the corresponding m-line, which is set to *recvonly*.

Y.6.5.3 Viewport dependent delivery of fisheye video

By exposing the coverage information of each fisheye circular video using the parameters in section Y.6.5.2, the collective multitude of which makes up the whole 360-degree video, a ITT4RT-Rx client can opt to select only the required fisheye circular videos needed to render the current viewport of the user.

Through the parameters defined in section Y.6.5.2, a ITT4RT-Rx client can select the desired fisheye packing configuration of the video stream during SDP negotiation, as well as the initial desired fisheye videos using the *id* parameter.

Once a session is established, dynamic delivery of the desired fisheye videos depending the ITT4RT-Rx client user's viewport can be enabled using RTCP-based signalling, specifically with the RTCP feedback message with type "Viewport" as defined in Y.7.2.

Y.6.6 Camera Calibration for Network-based Stitching

Network-based stitching in the context of ITT4RT refers to generation of 360-degree videos in the ITT4RT MRF based on 2D video captures received from MTSI clients. This clause describes SDP-based signalling of camera calibration parameters for this purpose using the *a=3gpp-camera-calibration* attribute and SDP-based grouping of the corresponding 2D video captures using the *a=stitch_group* attribute.

The SDP syntax for *a=3gpp-camera-calibration* is defined with the following semantics (detailed ABNF presented at the end of the clause):

```
3gpp-camera-calibration = "a=3gpp-camera-calibration:" [SP "Param 1" SP "Param 2" SP ..... SP "Param K"]
```

where "Param 1", ..., "Param K" express the set of intrinsic and extrinsic camera parameters as specified below.

If the ITT4RT-Tx client in the ITT4RT MRF intends to perform network-based stitching to generate 360-degree video from a particular set of 2D video captures received from an MTSI sender, it shall use the SDP session-level attribute *a=stitch_group* before any media lines that correspond to the particular 2D video captures during the SDP negotiation of the corresponding media. Likewise, an MTSI sender capable of capturing 2D videos for 360-degree video generation shall use the session-level *a=stitch_group* attribute in the SDP before any media lines that correspond to the particular 2D video captures. The *a=stitch_group* attribute is used to group the corresponding to-be-stitched 2D video captures using the *mid* attribute as defined according to the ABNF below:

```
att-field = "stitch_group"
att-value = mid * [SP mid]
mid = token
; token is defined in RFC 4566
```

The *mid* attribute with the appropriate value as defined in the other parts of the SDP shall be included in the media description for the relevant 2D video captures when the *a=stitch_group* attribute is used. Furthermore, for each of these 2D video captures, the MTSI sender shall also include the SDP attribute *3gpp-camera-calibration* in the SDP under the relevant *m=* line for that particular video to signal the relevant camera calibration information. The order of the media included in the *a=stitch_group* indicates the synchronization source with the first media always being the synchronization anchor when synchronization is required.

More specifically, detailed camera calibration parameters based on ISO/IEC 23008-2 [3] are provided as follows, considering the multi-view acquisition information SEI message for HEVC. With these specifications, a 3-dimensional world point, $wP = [x\ y\ z]$ is mapped to a 2-dimensional camera point, $cP[i] = [u\ v\ 1]$, for the *i*-th camera according to:

$$s * cP[i] = A[i] * R^{-1}[i] * (wP - T[i]) \quad (\text{eqn. Y.6.6.1})$$

where $A[i]$ denotes the intrinsic camera parameter matrix, $R^{-1}[i]$ denotes the inverse of the rotation matrix $R[i]$, $T[i]$ denotes the translation vector, and *s* (a scalar value) is an arbitrary scale factor chosen to make the third coordinate of $cP[i]$ equal to 1.

Equation Y.6.6.1 can be extended to incorporate the entrance pupil variation to correct the incidence ray of $cP[i] = [u\ v\ 1]$ such that it always passes through the camera optical center, thereby removing distortion. The resulting entrance pupil coefficients $E[i]$ may be incorporated into Equation Y.6.6.1 as

$$s * cP[i] = A[i] * R^{-1}[i] * ((wP + E) - T[i]) \quad (\text{eqn. Y.6.6.2})$$

where $wP + E[i] = [x \ y \ z + E]$, $E = e1 * \theta^3 + e2 * \theta^5 + e3 * \theta^7 + e4 * \theta^9$, θ is the incidence angle pertaining to each ray formed by the pixel $cP[i] = [u \ v \ 1]$, and $[e1, e2, e3, e4]$ are entrance pupil coefficients. In addition, the accuracy of these entrance pupil parameters have an influence of the accuracy of estimated extrinsic parameters and thus improve the future imaging tasks. If not available, vector E is considered as 0 and a fallback to eqn. Y.6.6.1 is expected.

Accordingly, the following intrinsic camera parameters can be signalled in the SDP for each 2D video capture using the `a=3gpp-camera-calibration` attribute:

`focalLengthX[i]` specifies the focal length of the i -th camera in the horizontal direction as a signed floating-point number.

`focalLengthY[i]` specifies the focal length of the i -th camera in the vertical direction as a signed floating-point number.

`principalPointX[i]` specifies the principal point of the i -th camera in the horizontal direction as a signed floating-point number.

`principalPointY[i]` specifies the principal point of the i -th camera in the vertical direction as a signed floating-point number.

`skewFactor[i]` specifies the skew factor of the i -th camera as a signed floating-point number.

The intrinsic matrix $A[i]$ for i -th camera is represented by:

$$\begin{bmatrix} \text{focalLengthX}[i] & \text{skewFactor}[i] & \text{principalPointX}[i] \\ 0 & \text{focalLengthY}[i] & \text{principalPointY}[i] \\ 0 & 0 & 1 \end{bmatrix}$$

It is possible that the intrinsic camera parameters are equal for all of the cameras. In that case, only one set of values based on the above parameters would need to be signalled, e.g., via SDP signalling at the session level.

Furthermore, the following extrinsic camera parameters can be signalled in the SDP for each camera as per ISO/IEC 23008-2 [3]:

`rE[i][j][k]` specifies the (j, k) component of the rotation matrix for the i -th camera as a signed floating-point number.

The rotation matrix $R[i]$ for i -th camera is represented as follows:

$$\begin{bmatrix} rE[i][0][0] & rE[i][0][1] & rE[i][0][2] \\ rE[i][1][0] & rE[i][1][1] & rE[i][1][2] \\ rE[i][2][0] & rE[i][2][1] & rE[i][2][2] \end{bmatrix}$$

`tE[i][j]` specifies the j -th component of the translation vector for the i -th camera as a signed floating-point number.

The translation vector $T[i]$ for the i -th camera is represented by:

$$\begin{bmatrix} tE[i][0] \\ tE[i][1] \\ tE[i][2] \end{bmatrix}$$

For the i -th camera, $E[i][j]$ specifies the j -th component of the entrance pupil coefficient $[e1, e2, e3, e4]$ where $j=1, \dots, 4$. The parameters are represented as a signed floating-point number, as per eqn (2) above.

The syntax for the "`a=3gpp-camera-calibration`" attribute shall conform to the following ABNF:

`att-field` = "3gpp-camera-calibration"

`att-value` = PT 1*WSP attr-list

PT = 1*DIGIT / "*"

```

attr-list = ( set *(1*WSP set) ) / "*"
           ; WSP and DIGIT defined in [RFC5234]
;sub-rules for set
set = "[" "focalLengthX=" sfloatvalue ",focalLengthY=" sfloatvalue
      ",skewFactor=" sfloatvalue ",principalPointX=" sfloatvalue
      ",principalPointY=" sfloatvalue ",rotation00=" sfloatvalue
      ",rotation01=" sfloatvalue ",rotation02=" sfloatvalue ",rotation10="
sfloatvalue ",rotation11=" sfloatvalue ",rotation12=" sfloatvalue
      ",rotation20=" sfloatvalue ",rotation21=" sfloatvalue ",rotation22="
sfloatvalue ",translation0=" sfloatvalue ",translation1="
sfloatvalue ",translation2=" sfloatvalue ",epupil1=" sfloatvalue
      ",epupil2=" sfloatvalue ",epupil3=" sfloatvalue ",epupil4="
sfloatvalue "]"
sfloatvalue = [sign] sizevalue [ "." 6*DIGIT ]
sign = "-"
sizevalue = POS-DIGIT *5DIGIT
           ; POS-DIGIT is defined in Y.6.2.1

```

Y.6.7 Support for Stream Pausing/Resuming

An ITT4RT-Tx client shall use the `a=rtcp-fb ccm pause` attribute and parameter values as specified in [43] and [156] to indicate the capability to support receiving and acting on PAUSE and RESUME requests targeted for RTP streams it sends. The optional parameter setting of `a=rtcp-fb ccm pause config=3` could be used by the ITT4RT-Tx client to indicate that it will only receive and react to PAUSE and RESUME requests but will not send them.

An ITRT4RT-Rx client shall use the `a=rtcp-fb ccm pause` attribute and parameter values as specified in [43] and [156] to indicate the capability to support sending PAUSE and RESUME requests targeted for RTP streams it receives. The optional parameter setting of `a=rtcp-fb ccm pause config=2` could be used by the ITT4RT-Rx client to indicate that it will only send PAUSE and RESUME requests but does not support receiving these requests.

Y.6.8 Multiple 360-degree videos

An ITT4RT conference may contain multiple 360-degree videos which originate from multiple conference rooms at the conference location, or from remote participants. When multiple 360-degree videos are present in an ITT4RT conference, an ITT4RT MRF shall negotiate an SDP session with every remote participant.

In the SDP offer from the ITT4RT MRF, 360-degree video is identified by either the `a=3gpp_360video` or `a=3gpp_fisheye` media line attributes. When multiple 360-degree videos are present in the SDP offer, the ITT4RT MRF shall include the `a=content` attribute under the media lines for 2D or 360-degree video originating from the conference location. For media streams originating from the main default conference room, the content attribute is set to `a=content:main`. For media streams originating from other conference rooms, the content attribute is set to `a=content:alt`. 2D and 360-degree video from remote participants shall not include the `a=content` attribute under their corresponding media lines.

When there are multiple 360-degree videos from multiple sources available to the ITT4RT MRF, the ITT4RT MRF may include the `'itt4rt_group'` attribute (as defined in Y.6.2.6) to define one or more restricting groups, each group containing at least one mid associated with a 360-degree video media line, and at least one mid associated with an overlay.

On receipt of an SDP offer containing multiple 360-degree videos from the ITT4RT MRF, an ITT4RT-Rx client shall select to receive only one 360-degree video media together with possible 2D video media from other sources, rejecting the other 360-degree video media.

Example SDP offers for multiple 360-degree video with and without group restrictions are shown in clause Y.8.

Y.6.8.2 Excluding other participants' overlays

When an ITT4RT-Tx client in terminal sends a 360-degree video media stream to the MRF, it may include an attribute "a= no_other_overlays", which indicates that the MRF shall not group the 360-degree media stream from that ITT4RT-Tx client with overlay media streams from other ITT4RT clients. In this case, the MRF shall group the 360-degree video media stream and one or more overlays of that ITT4RT-Tx client in a separate <rest-group> in the itt4rt_group attribute when describing them to any ITT4RT-Rx client.

The ABNF syntax for this attribute is the following:

```
att-field = "no_other_overlays"
```

NOTE: If multiple itt4rt_group are created, an ITT4RT-Rx client in terminal would need to re-negotiate the session to switch to media streams from other itt4rt_group. However, doing so may add further burden on the signaling nodes.

Y.6.9 Scene Description-Based Overlays

Y.6.9.1 General

ITT4RT clients that support the "Overlay" feature may support the scene description as defined in [183] for signaling the overlay configuration.

If scene description-based overlays are supported, the following subset of the MPEG-I scene description extensions and features shall be supported:

- The MPEG_media extension: used to reference the media streams.
- The MPEG_accessor_timed and the MPEG_buffer_circular: used to bind timed media.
- The MPEG_texture_video: used to define video textures for the overlay and the 360 video.
- The scene description update mechanism as defined in clause 5.2.4 of [183].

If scene description-based overlays are used in an ITT4RT session with multiple participants, then the ITT4RT MRF shall be used for the session and shall own the scene description.

If scene description-based overlays are used, then the ITT4RT-TX client in the ITT4RT MRF shall:

- Create a sphere or cubemap mesh node (depending on the selected projection) in the scene description for each 360 video stream in the ITT4RT session. The source of the node's texture shall reference the ITT4RT media stream of the corresponding 360 video as signaled by the SDP.
- Create a rectangular or spherical mesh node in the scene description for each overlay stream in the ITT4RT session. The source of the node's texture shall reference the media stream of the corresponding overlay stream as signaled by the SDP.
- The location of the overlay shall be indicated by the transformation of the corresponding overlay node in the scene description.

NOTE: In a scene description-based overlay solution, the scene camera corresponds the viewer's position and it tracks the user's 3DoF movements. The camera's projection determines the field of view of the user.

The URL format as specified in 23090-14 Annex C shall be used to reference media streams in the ITT4RT session.

For participants that support scene description, the overlay information and positioning that is provided as part of the scene description shall take precedence over any information provided as part of the 3gpp_overlay attribute.

An ITT4RT-Tx client in terminal that offers overlays may select to signal the overlay either through the `3gpp_overlay` attribute or through a scene update that adds the overlay node. The scene update mechanism is described in [183]. In case the ITT4RT-Tx uses the `3gpp_overlay` attribute to describe its overlays, the ITT4RT-Tx client in the ITT4RT MRF shall generate the scene description or scene description update document that signals the presence and position of that overlay.

Y.6.9.2 Offer/Answer Negotiation

An ITT4RT-Tx client that desires to use scene description-based overlays, shall offer a data channel with a data channel indicating the “mpeg-sd” sub-protocol. The ITT4RT-Rx client in the MRF that supports scene-based overlays may answer by accepting the scene description data channel.

If the offer is accepted, the ITT4RT MRF shall generate and send the scene description to the offerer upon establishment of the data channel.

If the ITT4RT MRF receives an offer that does not contain a data channel with the “mpeg-sd” sub-protocol, it shall assume that the offering ITT4RT client does not support scene description-based overlays. In such case, the answering ITT4RT MRF shall not add a data channel with the “mpeg-sd” sub-protocol and may describe any overlays using the `3gpp_overlay` attribute.

Y.6.9.3 SDP Signaling

An ITT4RT-Tx in the ITT4RT MRF that supports scene description-based overlays, shall support MTSI data channel media and act as a DCMTSI client. The stream id of the data channel with the sub-protocol “mpeg-sd” shall be in the range allocated for bootstrap data channels, i.e. below 1000, excluding values in Table 6.2.10.1-2. A single data channel with sub-protocol “mpeg-sd” shall be present in the offer/answer SDP. If multiple data channels with the “mpeg-sd” sub-protocol are detected, the one with the lowest stream ID shall be used. The scene description data channel shall be configured as ordered, reliable, with normal SCTP multiplexing priority.

When scene description-based overlays are offered, the ITT4RT-Tx in the ITT4RT MRF shall offer a data channel with a stream id that indicates the “mpeg-sd” subprotocol in the `dcmap` attribute. The “mpeg-sd” messages shall be JSON formatted in UTF-8 coding without BOM.

Scene description-based overlay descriptions, including complete scene descriptions and scene updates, shall be delivered through the same data channel.

Y.7 Media transport

Y.7.1 RTP

RTP signaling impacts are FFS.

Y.7.2 RTCP

When viewport-dependent processing (VDP) of 360-degree video is successfully negotiated between the ITT4RT-Tx client and ITT4RT-Rx client as per the SDP-based procedures described in clause Y.6.2, the RTCP feedback (FB) message type ‘Viewport’ shall carry requested viewport information during the RTP transmission of media signalled from the ITT4RT-Rx client to the ITT4RT-Tx client.

The signalling of ‘Viewport’ requests shall use RTCP feedback messages as specified in IETF 4585. The RTCP feedback message is identified by PT (payload type) = PSFB (206) which refers to payload-specific feedback message. FMT (feedback message type) shall be set to the value assigned by IANA for Viewport feedback messages. The IANA registration information for the FMT value for ‘Viewport’ is provided in Annex R.3 of TS 26.114. The RTCP feedback method may involve signalling of viewport information in both of the immediate feedback and early RTCP modes.

The FCI (feedback control information) format for Viewport shall be as follows. The FCI shall contain exactly one viewport. The signalled desired viewport information in the RTCP feedback message for ‘Viewport’ is composed of the following parameters (as defined as shape type value equal to 0 in clause 7.5.6 of ISO/IEC 23090-2 [179]):

Y.8 SDP Examples (informative)

An example SDP offer for multiple 360-degree video is shown in table Y.8.1.

Table Y.8.1: Example SDP offer with multiple 360-degree video

| SDP offer |
|---|
| <pre> /*1st conference room @ conference location*/ m=video 49144 RTP/AVP 98 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:950 b=RS:0 b=RR:5000 a=content:main /*indicates this m-line is from the main source from the conference location*/ a=rtpmap:98 H265/90000 /*360 video*/ a=3gpp_360video:98... a=rtpmap:100 H265/90000 /*2D video*/ /*2nd conference room @ conference location*/ m=video 49154 RTP/AVP 102 104 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:950 b=RS:0 b=RR:5000 a=content:alt /*indicates this m-line is from an alt source from the conference location*/ a=rtpmap:102 H265/90000 /*360 video*/ a=3gpp_360video:102... a=rtpmap:104 H265/90000 /*2D video*/ /*Capture of a remote participant*/ m=video 49164 RTP/AVP 106 108 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:950 b=RS:0 b=RR:5000 a=rtpmap:106 H265/90000 /*2D video*/ a=rtpmap:108 H265/90000 /*360 video*/ a=3gpp_360video:108... </pre> |

An example SDP offer for multiple 360-degree video with group restrictions is shown in table Y.8.2.

Table Y.8.2: Example SDP offer with multiple 360-degree video containing group restrictions

| SDP offer |
|---|
| <pre> a=itt4rt_group: D G H / E F H / H G I /*create multiple groups at MRF, each group between a different 360 video from a certain source, and 2D videos from other sources/ /*360 video of 1st conference room @ conference location*/ m=video 49144 RTP/AVP 98 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:950 b=RS:0 b=RR:5000 a=content:main /*indicates this m-line is from the main source from the conference location*/ a=rtpmap:98 H265/90000 /*360 video*/ a=3gpp_360video:98... a=mid=D /*2D video of 1st conference room @ conference location*/ m=video 49154 RTP/AVP 100 a=tcap:1 RTP/AVPF a=pcfg:1 t=1 b=AS:950 </pre> |

```

b=RS:0
b=RR:5000
a=content:main /*indicates this m-line is from the main source from the conference
location*/
a=rtpmap:100 H265/90000 /*2D video*/
a=3gpp_overlay:100... /*optional attribute*/
a=mid=E

/*360 video of 2nd conference room @ conference location*/
m=video 49164 RTP/AVP 102
a=tcap:1 RTP/AVPF
a=pcfg:1 t=1
b=AS:950
b=RS:0
b=RR:5000
a=content:alt /*indicates this m-line is from an alt source from the conference
location*/
a=rtpmap: 102 H265/90000 /*360 video*/
a=3gpp_360video:98...
a=mid=F

/*2D video of 2nd conference room @ conference location*/
m=video 49174 RTP/AVP 104
a=tcap:1 RTP/AVPF
a=pcfg:1 t=1
b=AS:950
b=RS:0
b=RR:5000
a=content:alt /*indicates this m-line is from an alt source from the conference
location*/
a=rtpmap:100 H265/90000 /*2D video*/
a=3gpp_overlay:104... /*optional attribute*/
a=mid=G

/*2D video capture of a remote participant*/
m=video 49164 RTP/AVP 106
a=tcap:1 RTP/AVPF
a=pcfg:1 t=1
b=AS:950
b=RS:0
b=RR:5000
a=rtpmap:106 H265/90000 /*2D video*/
a=3gpp_overlay:100... /*optional attribute*/
a=mid=H

/*3D capture of a remote participant*/
m=video 49164 RTP/AVP 108
a=tcap:1 RTP/AVPF
a=pcfg:1 t=1
b=AS:950
b=RS:0
b=RR:5000
a=rtpmap:108 H265/90000 /*360 video*/
a=3gpp_360video:108...
a=mid=I

```

Y.9 Recommended audio mixing gains

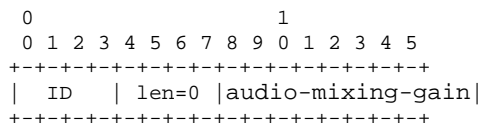
An ITT4RT-Tx client may specify recommended gains for mixing of its transmitted audio streams and update these recommended gains during a session. An ITT4RT-Rx client may or may not use such recommended mixing gains to scale the audio streams prior to mixing.

An ITT4RT-Tx client may for example send the recommended mixing gains r_0, r_1, \dots, r_N for the audio sources a_0 (360 video) and a_1, a_2, \dots, a_N (overlay videos) of that sender to recommend a mix at the receivers to be $r_0*a_0+r_1*a_1+\dots+r_N*a_N$.

If an ITT4RT-Rx client negotiated to receive recommended audio mixing gains and the ITT4RT-Tx client chooses to send these mixing gains, the ITT4RT-Tx client shall indicate each audio mixing gain value to the ITT4RT-Rx client using RTP header-extensions (see clause Y.9.1).

Y.9.1 RTP header extension for audio mixing gain

The format for sending a recommended audio mixing gain using the RTP header extension shall be as follows:



Audio mixing gain using One-Byte Header Format

The 4-bit ID is the local identifier of this element. The 4-bit length is the number, minus one, of data bytes of this header extension element following the one-byte header. The URI for declaring the audio mixing gain header extension in a Session Description Protocol (SDP) extmap attribute [95] and mapping it to a local identifier is:

```
urn:3gpp:audio-mixing-gain
```

The audio mixing gain is expressed in dB as a signed integer in the range "-127" to "0" (hence the numerical values directly represent the gain in dB). A value of "-128" indicates muting the channel. The meaning of positive values other than 0 is undefined and shall be ignored if received.

An ITT4RT-Tx client may repeat the header extension over multiple RTP packets to improve the likelihood of successful transmission as described in [RFC 8285]. The number of header extension transmissions (for the same recommended mixing gain) should therefore depend on the probability of delivery.

Annex Z (informative): Change history

| Change history | | | | | | | |
|----------------|-------|-----------|------|-----|--|-------|-------|
| Date | TSG # | TSG Doc. | CR | Rev | Subject/Comment | Old | New |
| 2007-03 | 35 | SP-070110 | | | Approved at TSG SA #35 | 2.0.0 | 7.0.0 |
| 2007-06 | 36 | SP-070318 | 0001 | 2 | Addition of non-compound RTCP | 7.0.0 | 7.1.0 |
| 2007-06 | 36 | SP-070318 | 0002 | 1 | Addition of DTMF | 7.0.0 | 7.1.0 |
| 2007-06 | 36 | SP-070318 | 0005 | 2 | Video QoS Profile | 7.0.0 | 7.1.0 |
| 2007-06 | 36 | SP-070318 | 0006 | 1 | Correction of the reference to the AMR/AMR-WB RTP payload format | 7.0.0 | 7.1.0 |
| 2007-06 | 36 | SP-070318 | 0007 | 1 | Video rate adaptation in MTSI | 7.0.0 | 7.1.0 |
| 2007-06 | 36 | SP-070318 | 0008 | 1 | Improved Video support for MTSI | 7.0.0 | 7.1.0 |
| 2007-09 | 37 | SP-070631 | 0009 | 1 | Correction of List of Abbreviations, Section 13, and Annex F | 7.1.0 | 7.2.0 |
| 2007-09 | 37 | SP-070631 | 0010 | | Clarification text for Media Synchronization Info | 7.1.0 | 7.2.0 |
| 2007-12 | 38 | SP-070763 | 0012 | 1 | Correction of minor errors | 7.2.0 | 7.3.0 |
| 2007-12 | 38 | SP-070763 | 0013 | 1 | Correction of references in MTSI | 7.2.0 | 7.3.0 |
| 2007-12 | 38 | SP-070763 | 0014 | 1 | Synchronization information | 7.2.0 | 7.3.0 |
| 2007-12 | 38 | SP-070763 | 0015 | | Encapsulation parameters | 7.2.0 | 7.3.0 |
| 2007-12 | 38 | SP-070763 | 0016 | 1 | Video codec in MTSI | 7.2.0 | 7.3.0 |
| | | | | | Inclusion of attachments | 7.3.0 | 7.3.1 |
| 2008-03 | 39 | SP-080005 | 0017 | 2 | Correction of the SDP example | 7.3.1 | 7.4.0 |
| 2008-03 | 39 | SP-080005 | 0018 | 1 | Alignment of SDP examples | 7.3.1 | 7.4.0 |
| 2008-03 | 39 | SP-080005 | 0020 | 1 | Codec mode requests | 7.3.1 | 7.4.0 |
| 2008-03 | 39 | SP-080005 | 0022 | 1 | Correction to Figure 10.1 | 7.3.1 | 7.4.0 |
| 2008-06 | 40 | SP-080248 | 0019 | 1 | DTMF clarifications | 7.4.0 | 7.5.0 |
| 2008-06 | 40 | SP-080249 | 0024 | 2 | Introduction of network preference management object | 7.4.0 | 7.5.0 |
| 2008-06 | 40 | SP-080248 | 0025 | | max-red alignment and clarifications | 7.4.0 | 7.5.0 |
| 2008-06 | 40 | SP-080248 | 0026 | 1 | Addition of QoS mapping for interactive bearer for real-time text | 7.4.0 | 7.5.0 |
| 2008-06 | 40 | SP-080248 | 0027 | 2 | Clarifications on bandwidths in SDP | 7.4.0 | 7.5.0 |
| 2008-06 | 40 | SP-080248 | 0028 | 1 | Codec Control Messages (CCM) update | 7.4.0 | 7.5.0 |
| 2008-06 | 40 | SP-080248 | 0029 | 1 | Terminology improvements | 7.4.0 | 7.5.0 |
| 2008-06 | 40 | SP-080248 | 0030 | | 3G-324M inter-working in GERAN/UTRAN CS | 7.4.0 | 7.5.0 |
| 2008-06 | 40 | SP-080248 | 0031 | 1 | ptime and encapsulation alignment | 7.4.0 | 7.5.0 |
| 2008-09 | 41 | SP-080469 | 0032 | 1 | Introduction of SDP Capability Negotiation | 7.5.0 | 7.6.0 |
| 2008-09 | 41 | SP-080469 | 0033 | 2 | Clarifications on the usage of AVPF NACK and PLI | 7.5.0 | 7.6.0 |
| 2008-09 | 41 | SP-080469 | 0035 | 1 | Transmission of H.264 parameter sets | 7.5.0 | 7.6.0 |
| 2008-09 | 41 | SP-080469 | 0036 | | Clarification of H.263 support in MTSI | 7.5.0 | 7.6.0 |
| 2008-09 | 41 | SP-080476 | 0034 | 1 | Improvements to MTSI 3G324M interworking | 7.6.0 | 8.0.0 |
| 2008-12 | 42 | SP-080679 | 0039 | 2 | MTSI QoE configuration and reporting | 8.0.0 | 8.1.0 |
| 2008-12 | 42 | SP-080673 | 0041 | 1 | MTSI QoE configuration and reporting | 8.0.0 | 8.1.0 |
| 2008-12 | 42 | SP-080673 | 0043 | 1 | Negotiated GBR and AS bandwidths | 8.0.0 | 8.1.0 |
| 2008-12 | 42 | SP-080673 | 0045 | 1 | Additional SDP examples for SDP answers | 8.0.0 | 8.1.0 |
| 2008-12 | 42 | SP-080673 | 0047 | | Initial codec mode | 8.0.0 | 8.1.0 |
| 2008-12 | 42 | SP-080679 | 0048 | 2 | RTP/RTCP symmetric media stream | 8.0.0 | 8.1.0 |
| 2009-03 | 43 | SP-090013 | 0049 | 1 | SDP Offer/Answer Examples for Video Sessions using "a=imageattr" | 8.1.0 | 8.2.0 |
| 2009-03 | 43 | SP-090005 | 0051 | 1 | Session setup clarifications | 8.1.0 | 8.2.0 |
| 2009-03 | 43 | SP-090012 | 0052 | 1 | Inter-working enhancements | 8.1.0 | 8.2.0 |
| 2009-03 | 43 | SP-090013 | 0053 | 1 | Dynamic video rate adaptation | 8.1.0 | 8.2.0 |
| 2009-03 | 43 | SP-090012 | 0054 | | MTSI QoE Corrections | 8.1.0 | 8.2.0 |
| 2009-03 | 43 | SP-090012 | 0055 | 1 | Adding second stream of same media type | 8.1.0 | 8.2.0 |
| 2009-03 | 43 | SP-090005 | 0057 | 2 | Session setup clarifications | 8.1.0 | 8.2.0 |
| 2009-03 | | | | | Attachments added to the .zip file | 8.2.0 | 8.2.1 |
| 2009-06 | 44 | SP-090255 | 0059 | | MTSI QoE Correction | 8.2.1 | 8.3.0 |
| 2009-06 | 44 | SP-090255 | 0060 | | Cross reference correction | 8.2.1 | 8.3.0 |
| 2009-06 | 44 | SP-090258 | 0061 | 2 | Packetization for E-UTRAN | 8.2.1 | 8.3.0 |
| 2009-06 | 44 | SP-090246 | 0063 | | Correction of unclear packetization recommendation for unknown RAT | 8.2.1 | 8.3.0 |
| 2009-06 | 44 | SP-090246 | 0065 | 1 | Corrections to example in clause A.6 | 8.2.1 | 8.3.0 |
| 2009-06 | 44 | SP-090246 | 0067 | 1 | Correction to comments in sub-clause A.3.1a | 8.2.1 | 8.3.0 |
| 2009-06 | 44 | SP-090246 | 0069 | 1 | Correction of inconsistent packetization requirements | 8.2.1 | 8.3.0 |
| 2009-06 | 44 | SP-090246 | 0071 | 1 | Corrections to examples in clauses A.7 and A.8 | 8.2.1 | 8.3.0 |
| 2009-06 | 44 | SP-090246 | 0073 | 1 | Correction to CTM Interworking | 8.2.1 | 8.3.0 |
| 2009-09 | 45 | SP-090566 | 0076 | | MSRP corrections | 8.3.0 | 8.4.0 |
| 2009-09 | 45 | SP-090566 | 0077 | | QoE Corrections for MTSI | 8.3.0 | 8.4.0 |
| 2009-09 | 45 | SP-090566 | 0083 | 1 | Session setup clarifications | 8.3.0 | 8.4.0 |
| 2009-09 | 45 | SP-090566 | 0085 | 1 | Correction of OMA-DM MO DTD DFFormat in 3GPP MTSINP MO | 8.3.0 | 8.4.0 |

| Change history | | | | | | | |
|----------------|-------|-----------|------|-----|--|--------|--------|
| Date | TSG # | TSG Doc. | CR | Rev | Subject/Comment | Old | New |
| 2009-09 | 45 | SP-090571 | 0074 | 1 | Maintaining temporalspatial tradeoff during video adaptation / equalization of font and size for MO DDF specifications | 8.4.0 | 9.0.0 |
| 2009-09 | 45 | SP-090573 | 0081 | 1 | Initial Codec Mode Selection for AMR and AMR-WB | 8.4.0 | 9.0.0 |
| 2009-12 | 46 | SP-090716 | 0075 | 4 | Replacement of non-compound RTCP with Reduced-Size RTCP | 9.0.0 | 9.1.0 |
| 2009-12 | 46 | SP-090704 | 0082 | 6 | Adding Support for Explicit Congestion Notification | 9.0.0 | 9.1.0 |
| 2009-12 | 46 | SP-090708 | 0088 | 1 | Managing MTSI Media Adaptation | 9.0.0 | 9.1.0 |
| 2009-12 | 46 | SP-090716 | 0089 | 1 | MTSI DDF for QoE | 9.0.0 | 9.1.0 |
| 2009-12 | 46 | SP-090716 | 0091 | 1 | Variable encoding of video to facilitate quality-recovery techniques | 9.0.0 | 9.1.0 |
| 2010-03 | 47 | SP-100020 | 0094 | 2 | UE behaviour on the receipt of ECN-CE | 9.1.0 | 9.2.0 |
| 2010-03 | 47 | SP-100020 | 0095 | 1 | OMA-DM configuration parameters for ECN-triggered adaptation | 9.1.0 | 9.2.0 |
| 2010-03 | 47 | SP-100020 | 0096 | 2 | ECN adaptation example | 9.1.0 | 9.2.0 |
| 2010-03 | 47 | SP-100019 | 0101 | 1 | H324M interworking procedures in decomposed MGCF and IM-MGW | 9.1.0 | 9.2.0 |
| 2010-03 | 47 | SP-100028 | 0102 | 1 | MTSI Client Global Text Telephony negotiation procedures | 9.1.0 | 9.2.0 |
| 2010-06 | 48 | SP-100300 | 0103 | 4 | Work Split of 3GPP MTSINP and MTSIMA MOs | 9.2.0 | 9.3.0 |
| 2010-06 | 48 | SP-100296 | 0107 | 1 | Correction of AVP-AVPF negotiation procedure | 9.2.0 | 9.3.0 |
| 2010-12 | 50 | SP-100784 | 0111 | 3 | Correction of AMR and AMR-WB codec mode adaptation in the beginning of the session | 9.3.0 | 9.4.0 |
| 2010-12 | 50 | SP-100784 | 0113 | | Removal of ECN Nonce | 9.3.0 | 9.4.0 |
| 2011-03 | 51 | SP-110036 | 0115 | 3 | Corrections for ECN for speech | 9.4.0 | 9.5.0 |
| 2011-03 | 51 | SP-110040 | 0117 | 1 | Improved interoperability with non-MTSI AVP AMR clients | 9.4.0 | 9.5.0 |
| 2011-03 | 51 | SP-110036 | 0121 | 3 | Corrections related to MTSI gateway with split architecture | 9.4.0 | 9.5.0 |
| 2011-03 | 54 | SP-110036 | 0122 | | ECN corrections related to MTSI gateway with split architecture | 9.4.0 | 9.5.0 |
| 2011-03 | 51 | SP-110036 | 0124 | 1 | Correction of adaptation examples | 9.4.0 | 9.5.0 |
| 2011-03 | 51 | SP-110041 | 0108 | 10 | Video Adaptation Parameters for ECN | 9.5.0 | 10.0.0 |
| 2011-03 | 51 | SP-110041 | 0112 | 6 | Introduction of ECN for speech in HSPA/UTRA and for video in UTRA/HSPA/E-UTRA | 9.5.0 | 10.0.0 |
| 2011-03 | 51 | SP-110041 | 0114 | 5 | QoS profiles for MBR>GBR bearers | 9.5.0 | 10.0.0 |
| 2011-03 | 51 | SP-110047 | 0123 | 2 | Video coding enhancements in MTSI | 9.5.0 | 10.0.0 |
| 2011-06 | 52 | SP-110308 | 0125 | | Video Codec Enhancement to MTSI | 10.0.0 | 11.0.0 |
| 2011-09 | 53 | SP-110546 | 0129 | | Correction of the reference to the "a=imageattr" attribute | 11.0.0 | 11.1.0 |
| 2011-09 | 53 | SP-110553 | 0134 | 2 | Corrections related to unsupported video codecs | 11.0.0 | 11.1.0 |
| 2011-09 | 53 | SP-110551 | 0146 | | Correction of functional requirements for jitter buffer management | 11.0.0 | 11.1.0 |
| 2011-09 | 53 | SP-110545 | 0152 | | Clarification on SDP negotiation for video | 11.0.0 | 11.1.0 |
| 2011-09 | 53 | SP-110553 | 0154 | 2 | Example Configurations for H.264 AVC Level 1.2 | 11.0.0 | 11.1.0 |
| 2011-09 | 53 | SP-110552 | 0156 | 2 | Signalling applicable AMR/AMR-WB mode set in SDP answer | 11.0.0 | 11.1.0 |
| 2011-11 | 54 | SP-110791 | 0135 | 2 | Clarification of the methods to compute b=AS for speech | 11.1.0 | 11.2.0 |
| 2011-11 | 54 | SP-110786 | 0144 | 1 | Correction to H.264 RTP Payload Format Reference | 11.1.0 | 11.2.0 |
| 2011-11 | 54 | SP-110791 | 0157 | | Correction in the description of MTSINP example including H.263 | 11.1.0 | 11.2.0 |
| 2011-11 | 54 | SP-110791 | 0160 | 2 | Correction in ECN-related nodes of MTSIMA | 11.1.0 | 11.2.0 |
| 2011-11 | 54 | SP-110791 | 0163 | 2 | Correction in the usage of mode-set with MTSIMA | 11.1.0 | 11.2.0 |
| 2011-11 | 54 | SP-110797 | 0165 | | Correction of Sequence Parameter Set NAL unit values in the examples of SDP offers and answers for H.264 video | 11.1.0 | 11.2.0 |
| 2011-11 | 54 | SP-110797 | 0167 | | On correction to H.264 frame reordering usage | 11.1.0 | 11.2.0 |
| 2011-11 | 54 | SP-110790 | 0178 | | Update of reference for ECN | 11.1.0 | 11.2.0 |
| 2012-03 | 55 | SP-120024 | 0175 | 1 | Increasing optional codec level of H.264 | 11.2.0 | 11.3.0 |
| 2012-03 | 55 | SP-120027 | 0180 | 1 | Clarifying the usage of "a=imageattr" | 11.2.0 | 11.3.0 |
| 2012-03 | 55 | SP-120018 | 0186 | | Correcting order of SDP lines | 11.2.0 | 11.3.0 |
| 2012-03 | 55 | SP-120021 | 0191 | 2 | Correction of figure for RTP media and session set-up signaling paths | 11.2.0 | 11.3.0 |
| 2012-06 | 56 | SP-120220 | 0132 | 3 | Correction of mapping b=AS when MBR can be greater than GBR | 11.3.0 | 11.4.0 |
| 2012-06 | 56 | SP-120218 | 0194 | | IP Version Node for 3GPP MTSINP MO | 11.3.0 | 11.4.0 |
| 2012-06 | 56 | SP-120218 | 0197 | 2 | Bandwidth information for T.140 real-time text | 11.3.0 | 11.4.0 |
| 2012-06 | 56 | SP-120217 | 0204 | 1 | Correction to H.264 RTP Payload Format References | 11.3.0 | 11.4.0 |
| 2012-06 | 56 | SP-120226 | 0212 | 2 | Reasons for Recommending a Subset of AMR/AMR-WB Codec Modes | 11.3.0 | 11.4.0 |
| 2012-06 | 56 | SP-120220 | 0214 | 1 | Computation of MBR/GBR for AMR and AMR-WB | 11.3.0 | 11.4.0 |
| 2012-06 | 56 | SP-120218 | 0216 | 1 | SID Packetizing Procedures | 11.3.0 | 11.4.0 |
| 2012-09 | 57 | SP-120508 | 0199 | 5 | On support of fax | 11.4.0 | 11.5.0 |
| 2012-09 | 57 | SP-120496 | 0208 | 3 | Corrections and Clarifications to RTCP Bandwidth Modifiers | 11.4.0 | 11.5.0 |
| 2012-09 | 57 | SP-120502 | 0222 | 1 | QCI for T.140 | 11.4.0 | 11.5.0 |
| 2012-09 | 57 | SP-120500 | 0224 | 1 | Clarifying ECN Support of UTRAN and Video | 11.4.0 | 11.5.0 |
| 2012-09 | 57 | SP-120502 | 0226 | 1 | Replacing PDP Context Activation | 11.4.0 | 11.5.0 |
| 2012-09 | 57 | SP-120508 | 0227 | 1 | Clarifying SDP Attributes | 11.4.0 | 11.5.0 |
| 2012-09 | 57 | SP-120497 | 0230 | | Update of reference for ECN for RTP | 11.4.0 | 11.5.0 |
| 2012-09 | 57 | SP-120508 | 0231 | | Correcting b=AS for SID | 11.4.0 | 11.5.0 |
| 2012-10 | | | | | Inclusion of missing attachments | 11.5.0 | 11.5.1 |
| 2012-12 | 58 | SP-120765 | 0237 | 2 | On Coordination of Video Orientation | 11.5.1 | 12.0.0 |
| 2013-03 | 59 | SP-130013 | 0238 | 1 | Negotiation of AVPF and CCM feedback messages for video | 12.0.0 | 12.1.0 |

| Change history | | | | | | | |
|----------------|-------|-----------|------|-----|---|---------|---------|
| Date | TSG # | TSG Doc. | CR | Rev | Subject/Comment | Old | New |
| 2013-03 | 59 | SP-130021 | 0240 | 2 | Session setup for asymmetric video | 12.0.0 | 12.1.0 |
| 2013-03 | 59 | SP-130022 | 0241 | 2 | Non-CVO operation | 12.0.0 | 12.1.0 |
| 2013-03 | 59 | SP-130022 | 0243 | 3 | Enabling Higher Granularity Rotation in CVO | 12.0.0 | 12.1.0 |
| 2013-03 | 59 | SP-130023 | 0244 | 2 | Video Intra-Refreshing using SIP INFO | 12.0.0 | 12.1.0 |
| 2013-03 | 59 | SP-130022 | 0245 | 1 | Clarifying asymmetric operation in CVO | 12.0.0 | 12.1.0 |
| 2013-06 | 60 | SP-130188 | 0237 | 1 | Camera and Flip bits semantic clarifications | 12.1.0 | 12.2.0 |
| 2013-06 | 60 | SP-130188 | 0246 | 1 | Correction of text regarding order when compensating for flip and rotation | 12.1.0 | 12.2.0 |
| 2013-06 | 60 | SP-130181 | 0250 | 1 | Correction on ECN Capability Indication in the SDP | 12.1.0 | 12.2.0 |
| 2013-06 | 60 | SP-130191 | 0253 | 1 | Correcting Re-use of H.264 Parameter Set Identifier | 12.1.0 | 12.2.0 |
| 2013-06 | 60 | SP-130191 | 0254 | | Missing IANA registration information for two SDP attributes | 12.1.0 | 12.2.0 |
| 2013-09 | 61 | SP-130355 | 0255 | 1 | Procedures for selecting the media sending rate | 12.2.0 | 12.3.0 |
| 2013-09 | 61 | SP-130352 | 0257 | 2 | Complementing Definition of CVO Angle | 12.2.0 | 12.3.0 |
| 2013-09 | 61 | SP-130351 | 0259 | | Correction of DTMF handling | 12.2.0 | 12.3.0 |
| 2013-09 | 61 | SP-130352 | 0260 | | Correction to CVO | 12.2.0 | 12.3.0 |
| 2013-12 | 62 | SP-130568 | 0262 | 1 | Correction to references | 12.3.0 | 12.4.0 |
| 2013-12 | 62 | SP-130561 | 0264 | 1 | Correction to audio channels requirements | 12.3.0 | 12.4.0 |
| 2013-12 | 62 | SP-130578 | 0265 | 1 | Computation of video b=AS | 12.3.0 | 12.4.0 |
| 2013-12 | 62 | SP-130577 | 0272 | | E2EMTSI Fixed-mobile interworking | 12.3.0 | 12.4.0 |
| 2014-03 | 63 | SP-140008 | 0273 | | IANA Registration Information for RTP Header Extensions | 12.4.0 | 12.5.0 |
| 2014-03 | 63 | SP-140014 | 0275 | 1 | Fixed-mobile interworking | 12.4.0 | 12.5.0 |
| 2014-03 | 63 | SP-140009 | 0276 | 1 | HEVC support | 12.4.0 | 12.5.0 |
| 2014-06 | 64 | SP-140216 | 0277 | 1 | Transmission of VPS, SPS and PPS | 12.5.0 | 12.6.0 |
| 2014-06 | 64 | SP-140211 | 0278 | 5 | SDP examples and QoS examples for H.265 (HEVC) | 12.5.0 | 12.6.0 |
| 2014-06 | 64 | SP-140215 | 0279 | 1 | Fixed-mobile interworking | 12.5.0 | 12.6.0 |
| 2014-06 | 64 | SP-140209 | 0280 | | Correction on Missing Reference in CVO Specification | 12.5.0 | 12.6.0 |
| 2014-06 | 64 | SP-140216 | 0282 | 1 | Adding QCI examples to QoS examples | 12.5.0 | 12.6.0 |
| 2014-09 | 65 | SP-140596 | 0283 | 4 | Requirements for end-to-end video rate adaptation | 12.6.0 | 12.7.0 |
| 2014-09 | 65 | SP-140454 | 0286 | 1 | Correction to GBR, MBR, and RTCP BW in Video QoS Profiles | 12.6.0 | 12.7.0 |
| 2014-09 | 65 | SP-140478 | 0287 | 1 | Video telephony robustness improvements | 12.6.0 | 12.7.0 |
| 2014-09 | 65 | SP-140477 | 0288 | 1 | Errors in the computation of b=AS for AMR and AMR-WB | 12.6.0 | 12.7.0 |
| 2014-09 | 65 | SP-140476 | 0289 | 1 | Clarifications and SDP Example on Removal of Media Components | 12.6.0 | 12.7.0 |
| 2014-09 | 65 | SP-140476 | 0290 | 1 | MTSI Open Offer-Answer Procedures | 12.6.0 | 12.7.0 |
| 2014-09 | 65 | SP-140468 | 0296 | 3 | Introducing EVS into MTSI | 12.6.0 | 12.7.0 |
| 2014-12 | 66 | SP-140722 | 0297 | 4 | Incorporating EVS into MTSI | 12.7.0 | 12.8.0 |
| 2014-12 | 66 | SP-140722 | 0300 | 1 | Resolving status for EVS in MTSI | 12.7.0 | 12.8.0 |
| 2015-03 | 67 | SP-150082 | 0302 | 2 | "max-red" and "channels" parameters for EVS Primary and AMR-WB IO modes | 12.8.0 | 12.9.0 |
| 2015-03 | 67 | SP-150089 | 0303 | 1 | Separation of video codec parameters in SDP | 12.8.0 | 12.9.0 |
| 2015-03 | 67 | SP-150093 | 0304 | 2 | Relocating MTSINP, MTSIQoE, and MTSIMA MOs as Attachments | 12.8.0 | 12.9.0 |
| 2015-03 | 67 | SP-150082 | 0305 | 1 | Correcting b=AS values | 12.8.0 | 12.9.0 |
| 2015-03 | 67 | SP-150080 | 0306 | 1 | Alignments to VoLTE (GSMA IR.92) | 12.8.0 | 12.9.0 |
| 2015-03 | 67 | SP-150082 | 0307 | 1 | Corrections on Introduction of EVS into MTSI | 12.8.0 | 12.9.0 |
| 2015-06 | 68 | SP-150199 | 0311 | | Clarification of usage of EVS parameters in SDP offer | 12.9.0 | 12.10.0 |
| 2015-06 | 68 | SP-150199 | 0314 | 2 | Computing b=AS of EVS AMR-WB IO | 12.9.0 | 12.10.0 |
| 2015-06 | 68 | SP-150207 | 0315 | | Integrating H.265 (HEVC) into 3GPP MTSINP MO | 12.9.0 | 12.10.0 |
| 2015-06 | 68 | SP-150209 | 0317 | 2 | On transport of media and mode adaptation signalling in MTSI deploying RFC 4867 | 12.9.0 | 12.10.0 |
| 2015-06 | 68 | SP-150209 | 0319 | 1 | RTCP Reporting Interval | 12.9.0 | 12.10.0 |
| 2015-06 | 68 | SP-150210 | 0318 | 4 | Video Region-of-Interest Signaling | 12.10.0 | 13.0.0 |
| 2015-09 | 69 | SP-150433 | 0320 | | Integrating EVS into 3GPP MTSINP MO | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150433 | 0321 | 1 | Computing b=AS of EVS Primary in Dual-mono | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150433 | 0322 | 2 | Configuring "channels" when "ch-send" and "ch-recv" are different | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150433 | 0323 | 1 | Accepting received RTCP-APP of EVS Primary Mode | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150435 | 0325 | 1 | Clarification on CVO and imageattr in MTSI | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150456 | 0328 | 4 | Mandating AVPF with RTCP-APP | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150433 | 0330 | 4 | SDP examples of evs-mode-switch parameter | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150451 | 0333 | 1 | Video telephony robustness improvements extensions | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150443 | 0337 | 2 | Corrections, Clarifications and Guidelines on ROI | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150454 | 0340 | 2 | Inclusion of H.264/AVC Constrained High Profile in MTSI | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150433 | 0346 | 3 | Correction of payload format usage | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150435 | 0348 | 1 | On the use of one byte RTP header extension for CVO | 13.0.0 | 13.1.0 |
| 2015-09 | 69 | SP-150451 | 0349 | 1 | Recommendations on Packet Loss handling mechanisms | 13.0.0 | 13.1.0 |
| 2015-12 | 70 | SP-150638 | 0326 | 5 | Correction on session negotiation for EVS | 13.1.0 | 13.2.0 |
| 2015-12 | 70 | SP-150638 | 0335 | 3 | EVS QoS Profiles | 13.1.0 | 13.2.0 |
| 2015-12 | 70 | SP-150638 | 0336 | 3 | SDP example for negotiating only SC-VBR of EVS | 13.1.0 | 13.2.0 |
| 2015-12 | 70 | SP-150657 | 0354 | 1 | Addition of QoS End-to-end Multimedia Telephony Service for IMS (MTSI) extensions | 13.1.0 | 13.2.0 |

| Change history | | | | | | | |
|----------------|-------|-----------|------|-----|--|--------|--------|
| Date | TSG # | TSG Doc. | CR | Rev | Subject/Comment | Old | New |
| 2015-12 | 70 | SP-150659 | 0355 | 2 | Introduction of Multistream Multiparty Conference Media Handling | 13.1.0 | 13.2.0 |
| 2015-12 | 70 | SP-150647 | 0356 | | Corrections, Clarifications and Guidelines on ROI | 13.1.0 | 13.2.0 |
| 2015-12 | 70 | SP-150638 | 0358 | 1 | Correction of codec preference order | 13.1.0 | 13.2.0 |
| 2016-03 | 71 | SP-160064 | 0350 | 2 | Correction of EVS SDP examples | 13.2.0 | 13.3.0 |
| 2016-03 | 71 | SP-160064 | 0352 | 2 | Handling mode-change-capability parameter for EVS AMR-WB IO | 13.2.0 | 13.3.0 |
| 2016-03 | 71 | SP-160064 | 0361 | | Correction of SDP example | 13.2.0 | 13.3.0 |
| 2016-03 | 71 | SP-160075 | 0362 | 1 | Correction of SDP handling | 13.2.0 | 13.3.0 |
| 2016-03 | 71 | SP-160074 | 0363 | 2 | Completion of QOSE2EMTSI | 13.2.0 | 13.3.0 |
| 2016-03 | 71 | SP-160064 | 0364 | | Correction of an EVS SDP example for fixed access | 13.2.0 | 13.3.0 |
| 2016-03 | 71 | SP-160077 | 0365 | 1 | Clarification on handling multiple rtpmaps | 13.2.0 | 13.3.0 |
| | | | | | | | |

| Change history | | | | | | | |
|----------------|---------|-----------|------|---------|-----|--|----------------|
| Date | Meeting | TDoc | CR | R ev | Cat | Subject/Comment | New version |
| 2016-06 | SA#72 | SP-160257 | 0338 | 6 | F | Integrating EVS into 3GPP MTSIMA MO | 13.4.0 |
| 2016-06 | SA#72 | SP-160259 | 0367 | 1 | A | RFC 7798: RTP Payload Format for HEVC for MTSI | 13.4.0 |
| 2016-06 | SA#72 | SP-160266 | 0368 | 2 | F | Correction to Speech Media Flow Information | 13.4.0 |
| 2016-06 | SA#72 | SP-160264 | 0369 | | F | Correction of SDP examples for MMCMH | 13.4.0 |
| 2016-06 | SA#72 | SP-160263 | 0370 | | F | Addition of information to support IANA registration of new SDP attribute | 13.4.0 |
| 2016-06 | SA#72 | SP-160263 | 0371 | | F | Correction of SDP attribute description and usage for QOSE2EMTSI | 13.4.0 |
| 2016-06 | SA#72 | SP-160264 | 0375 | | F | Removal of simulcast and pause-resume | 13.4.0 |
| 2016-06 | SA#72 | SP-160269 | 0374 | 3 | B | RTCP Bandwidth Recommendations | 14.0.0 |
| 2016-09 | SA#73 | SP-160585 | 0382 | 1 | A | Corrections related to recommended mode set for AMR and AMR-WB | 14.1.0 |
| 2016-09 | SA#73 | SP-160594 | 0384 | - | A | Clarifications on using imageattr | 14.1.0 |
| 2016-09 | SA#73 | SP-160597 | 0385 | 1 | B | Re-insertion of simulcast and pause-resume | 14.1.0 |
| 2016-09 | SA#73 | SP-160589 | 0388 | - | A | Correction of SDP example | 14.1.0 |
| 2016-12 | SA#74 | SP-160772 | 0390 | 1 | B | Compact Concurrent Codec Capabilities SDP Usage for MMCMH CCCEX and Session Initiation | 14.2.0 |
| 2016-12 | SA#74 | SP-160772 | 0392 | 2 | B | Common Codec Identification and Usage | 14.2.0 |
| 2016-12 | SA#74 | SP-160772 | 0393 | 1 | B | MSMTSI MRF handling with reduced m-lines and multi-codec switching | 14.2.0 |
| 2016-12 | SA#74 | SP-160772 | 0394 | | F | Correction of SDP examples for EVS in MMCMH | 14.2.0 |
| 2016-12 | SA#74 | SP-160772 | 0395 | 1 | C | QoS for MMCMH Sessions | 14.2.0 |
| 2017-03 | SA#75 | SP-170024 | 0389 | 2 | C | Examples of Using the Compact CCC SDP Attribute | 14.3.0 |
| 2017-03 | SA#75 | SP-170024 | 0396 | 1 | B | IANA Registration of ccc-list SDP attribute | 14.3.0 |
| 2017-03 | SA#75 | SP-170024 | 0397 | 1 | F | Correction to definition of ccc-list SDP attribute | 14.3.0 |
| 2017-03 | SA#75 | SP-170024 | 0398 | - | F | Corrections to response to SIP OPTIONS request for ccex | 14.3.0 |
| 2017-03 | SA#75 | SP-170024 | 0399 | 1 | F | Clarifications on MSMTSI client referencing MSMTSI MRF and MSMTSI client in terminal | 14.3.0 |
| 2017-03 | SA#75 | SP-170024 | 0403 | 1 | F | Requirement clarification in S.5.1 for MMCMH | 14.3.0 |
| 2017-03 | SA#75 | SP-170024 | 0404 | 1 | F | Clarification of Thumbnail Handling in MMCMH | 14.3.0 |
| 2017-03 | SA#75 | SP-170024 | 0405 | - | B | IANA Registration of application/ccex media subtype | 14.3.0 |
| 2017-06 | SA#76 | SP-170320 | 0409 | 1 | F | Clarification on Thumbnail Handling | 14.4.0 |
| 2017-06 | SA#76 | SP-170324 | 0410 | 1 | C | RAN bitrate recommendation | 14.4.0 |
| 2017-09 | SA#77 | SP-170600 | 0406 | 2 | A | Correction of IPv6 header size | 14.5.0 |
| 2017-09 | SA#77 | SP-170601 | 0411 | 1 | F | ANBR GBR clarification | 14.5.0 |
| 2017-09 | SA#77 | SP-170603 | 0412 | - | C | MTSI QoE enhancements for new codecs | 15.0.0 |
| 2017-12 | SA#78 | SP-170825 | 0414 | - | C | QoE Control Plane Enhancements | 15.1.0 |
| 2017-12 | SA#78 | SP-170822 | 0415 | 1 | A | Transport of DTMF events | 15.1.0 |
| 2017-12 | SA#78 | SP-170822 | 0420 | 1 | A | Clarification of DTMF Event Operation | 15.1.0 |
| 2017-12 | SA#78 | SP-170828 | 0421 | 1 | B | NO_REQ Interworking | 15.1.0 |
| 2018-03 | SA#79 | SP-180023 | 0426 | 3 | A | Misc Changes in Clause 10 | 15.2.0 |
| 2018-03 | SA#79 | SP-180025 | 0427 | - | F | Correction of QMC Capability Signalling | 15.2.0 |
| 2018-06 | SA#80 | SP-180262 | 0430 | - | A | Clarification of Bandwidth Calculations in the UE | 15.3.0 |
| 2018-06 | SA#80 | SP-180278 | 0431 | 2 | B | Video Codec Requirements for 5G MTSI terminals | 15.3.0 |
| 2018-06 | SA#80 | SP-180268 | 0432 | - | F | Clarification on usage of geographical filtering for QoE reporting | 15.3.0 |
| 2018-09 | SA#81 | SP-180651 | 0433 | 2 | B | Speech Codecs for 5G MTSI Clients | 15.4.0 |
| 2018-09 | SA#81 | SP-180651 | 0434 | 1 | B | Adding NR and ANBR support | 15.4.0 |
| 2018-09 | SA#81 | SP-180641 | 0435 | - | F | Corrections to QMC signalling | 15.4.0 |
| 2018-12 | SA#82 | SP-180975 | 0436 | 3 | B | Recommendations on Media Rate Adaptation | 16.0.0 |
| 2018-12 | SA#82 | SP-180975 | 0438 | 6 | B | MTSI Client Profiles | 16.0.0 |
| 2018-12 | SA#82 | SP-180981 | 0439 | 3 | F | Correction in SDP Example for imageattr | 16.0.0 |
| 2018-12 | SA#82 | SP-180977 | 0441 | 2 | B | Signaling of Delay Budget Information | 16.0.0 |
| 2018-12 | SA#82 | SP-180981 | 0442 | 1 | D | Correction to MTSI network preference management object tree | 16.0.0 |
| 2018-12 | SA#82 | SP-180977 | 0443 | 3 | B | Recommendations on use of RAN Delay Budget Reporting in MTSI | 16.0.0 |
| 2019-03 | SA#83 | SP-190034 | 0448 | 1 | C | DBI Signaling Recommendations | 16.1.0 |
| 2019-03 | SA#83 | SP-190032 | 0450 | - | A | ECN Support in NR | 16.1.0 |
| 2019-03 | SA#83 | SP-190038 | 0451 | 2 | F | Clarification on b=AS with evs-mode-switch parameter | 16.1.0 |
| 2019-03 | SA#83 | SP-190036 | 0454 | - | B | Addition of reference to Alt_FX_EVS implementation | 16.1.0 |
| 2019-03 | SA#83 | SP-190031 | 0459 | 1 | A | Correction to MTSI network preference management object tree for EVS | 16.1.0 |
| 2019-06 | SA#84 | SP-190339 | 0461 | 1 | B | Signaling of ANBR Capabilities | 16.2.0 |
| 2019-06 | SA#84 | SP-190339 | 0462 | 1 | C | Additional DBI Signaling Recommendations | 16.2.0 |
| 2019-06 | SA#84 | SP-190339 | 0463 | 1 | C | SDP Examples on DBI Signaling | 16.2.0 |
| 2019-06 | SA#84 | SP-190336 | 0467 | - | A | Update FLEX FEC Usage and Reference | 16.2.0 |
| 2019-06 | SA#84 | SP-190335 | 0472 | - | A | CVO Corrections | 16.2.0 |
| 2019-06 | SA#84 | SP-190337 | 0474 | 1 | A | Correction to H.264 CBP Level 1.2 Requirement | 16.2.0 |
| 2019-06 | SA#84 | SP-190339 | 0475 | - | F | Metrics collection and reporting | 16.2.0 |
| 2019-09 | SA#85 | SP-190647 | 0478 | 2 | A | ANBR implementation issues | 16.3.0 |
| 2019-09 | SA#85 | SP-190646 | 0484 | 1 | A | Updating reference to RFC 8627 | 16.3.0 |
| 2019-09 | SA#85 | SP-190652 | 0446 | 8 | B | Addition of Coverage and Handoff Enhancements for Multimedia | 16.3.0 |

| | | | | | | | |
|---------|--------------|-----------|------|---|---|--|--------|
| 2019-09 | SA#85 | SP-190934 | 0479 | 5 | F | Interfacing MTSI client with 3GPP L2 | 16.3.0 |
| 2019-09 | SA#85 | SP-190650 | 0480 | - | C | Updates on ANBR Capability Signaling | 16.3.0 |
| 2019-09 | SA#85 | SP-190652 | 0485 | 1 | B | Management Objects for CHEM feature | 16.3.0 |
| 2019-09 | SA#85 | SP-190652 | 0486 | 1 | B | SDP Examples for CHEM feature | 16.3.0 |
| 2019-09 | SA#85 | SP-190650 | 0491 | - | F | SDP Attributes' Mux Category | 16.3.0 |
| 2019-12 | SA#86 | SP-191000 | 0460 | 5 | F | Clarifications on EVS SID update | 16.4.0 |
| 2019-12 | SA#86 | SP-190996 | 0492 | - | B | References to CHEM SDP Examples | 16.4.0 |
| 2019-12 | SA#86 | SP-190996 | 0493 | 1 | B | Registration and Corrections of CHEM SDP Attributes | 16.4.0 |
| 2019-12 | SA#86 | SP-190995 | 0495 | - | B | Amended QoS Signaling | 16.4.0 |
| 2020-03 | SA#87-e | SP-200033 | 0496 | 2 | B | Addition of MTSI Data Channel Media | 16.5.0 |
| 2020-03 | SA#87-e | | | | | editorial - change history update | 16.5.1 |
| | | | | | | | |
| 2020-06 | SA#88-e | SP-200394 | 0497 | | F | Correction of 3gpp-qos-hint examples | 16.6.0 |
| 2020-06 | SA#88-e | SP-200397 | 0498 | | D | Editorial Improvements | 16.6.0 |
| 2020-06 | SA#88-e | SP-200597 | 0499 | 1 | F | Adding Media Feature Tag for IMS Data Channel | 16.6.0 |
| 2020-06 | SA#88-e | SP-200595 | 0500 | 1 | C | QoE Measurement Collection | 16.6.0 |
| 2020-07 | SA#88-e | | | | | Corrections (editorial) | 16.6.1 |
| 2020-09 | SA#89-e | SP-200665 | 0503 | 1 | F | Correction to Delay Budget Reporting for Coverage Enhancements in NR | 16.7.0 |
| 2020-09 | SA#89-e | SP-200665 | 0502 | - | D | Editorial Improvements | 16.7.0 |
| 2020-09 | SA#89-e | SP-200665 | 0504 | - | D | Update of references | 16.7.0 |
| 2020-09 | SA#89-e | SP-200804 | 0501 | - | C | Removing H.263 and MPEG-4 Visual from MTSI | 16.7.0 |
| 2020-12 | SA#90-e | SP-200931 | 0505 | - | D | CR on 26.114 for editorial improvements | 16.8.0 |
| 2021-01 | Post SA#90-e | | | | | Minor Editorial Corrections. However, this version has inadvertently missed out the .zip attachments - hence not to be used. | 16.8.1 |
| 2021-01 | Post SA#90-e | | | | | g82 to be used instead of g81 as it doesnot contain the .zip attachments. Also the auto-numbering of the tables in table T.7, clauses T.1.2 and S.4.1 are corrected. | 16.8.2 |
| 2021-04 | SA#91-e | SP-210035 | 0509 | 1 | A | IETF Reference Update | 16.9.0 |
| 2021-04 | SA-91-e | SP-210035 | 0510 | - | F | IETF Reference Update for Data Channel Media | 16.9.0 |
| 2021-04 | SA#91-e | SP-210035 | 0513 | - | A | IETF Reference Update for MMCMH | 16.9.0 |
| 2021-04 | SA#91-e | SP-210040 | 0514 | | B | Video Support for ITT4RT | 17.0.0 |
| 2021-06 | SA#92-e | SP-210530 | 0517 | - | A | Updates on data channel (Rel-17) | 17.1.0 |
| 2021-06 | SA#92-e | SP-210531 | 0518 | - | A | IETF Reference Update for MMCMH (Rel-17) | 17.1.0 |
| 2021-06 | SA#92-e | SP-210537 | 0519 | - | B | Change Request on ITT4RT features | 17.1.0 |
| 2021-09 | SA#93-e | SP-210821 | 0521 | 1 | | QoE configuration release | 17.2.0 |
| 2021-12 | SA#94-e | SP-211347 | 0522 | - | C | Add MMtel Call Setup Time | 17.3.0 |
| 2022-03 | SA#95-e | SP-220033 | 0524 | 1 | F | Add support of per-slice QoE measurement | 17.4.0 |
| 2022-03 | SA#95-e | SP-220033 | 0525 | 1 | F | Requirement on UE Behavior regarding QoE Measurement and Reporting | 17.4.0 |
| 2022-03 | SA#95-e | SP-220035 | 0526 | 2 | B | CR on ITT4RT Phase 2 Features | 17.4.0 |
| 2022-06 | SA#96 | SP-220599 | 0527 | 1 | F | Editorial and Still Images | 17.5.0 |
| 2022-06 | SA#96 | SP-220600 | 0528 | 3 | F | Support of NR QoE features | 17.5.0 |
| 2022-12 | SA#98-e | SP-221039 | 0539 | - | A | Use of ICE with IMS Data Channel | 17.6.0 |
| 2022-12 | SA#98-e | SP-221053 | 0533 | 2 | F | Corrections to TS 26.114 on ITT4RT | 17.6.0 |
| 2022-12 | SA#98-e | SP-221054 | 0532 | 2 | F | CR to TS 26.114 Add slice scope into the QoE configuration | 17.6.0 |
| 2023-03 | SA#99 | SP-230250 | 0543 | - | F | Corrections to QoE configuration and reporting | 17.7.0 |
| 2023-03 | SA#99 | SP-230261 | 0541 | 2 | F | [ITT4RT] IANA registration of RTCP feedback for Viewport | 17.7.0 |
| 2023-03 | SA#99 | SP-230261 | 0547 | 1 | F | CR on 26.114 for the registration of SDP attributes for ITT4RT | 17.7.0 |
| 2023-06 | SA#100 | SP-230551 | 0552 | - | F | [ITT4RT] Correction to SDP example for itt4rt_group | 17.8.0 |
| 2023-06 | SA#100 | SP-230551 | 0554 | 1 | F | CR on 26.114 for scene description-based overlays (Rel-17) | 17.8.0 |
| 2025-01 | SA#106 | SP-241749 | 0576 | | F | Correction of RTCP Viewport feedback format value | 17.9.0 |

History

| Document history | | |
|-------------------------|--------------|-------------|
| V17.4.0 | April 2022 | Publication |
| V17.5.0 | July 2022 | Publication |
| V17.6.0 | January 2023 | Publication |
| V17.7.0 | April 2023 | Publication |
| V17.8.0 | July 2023 | Publication |
| V17.9.0 | January 2025 | Publication |