Wi-Fi Display
Technical Specification
Version 2.1

Document revision history

Version	Date YYYY-MM-DD	Remarks
1.0	2012-08-24	Public release version
1.1	2014-04-24	Public release for HDCPv2.2 updates.
2.0	2017-04-21	Release 2 final version.
2.1	2017-07-31	Corrected heading errors in sections 4.3 and 4.4. Removed references to Miracast.

Table of contents

1 INTRODUCTION 11
1.1 Scope 11
1.2 References 11
2 DEFINITIONS AND ACRONYMS 15
2.1.1 Shall/should/may/might word usage 15
2.1.2 Definitions 15
2.1.3 Abbreviations and acronyms 16
3 WFD ARCHITECTURE AND REQUIREMENTS 20
3.1 WFD Source, Primary Sink, Secondary Sink and WFD Session 20
3.1.1 WFD Source 21
3.1.2 WFD Sink(s) 21
3.1.3 Requirements for WFD Devices under Coupled Sink Operation 23
3.1.4 WFD Session 23
3.2 WFD Connection Topology 24
3.2.1 Wi-Fi P2P 24
3.2.2 TDLS 25
3.2.3 Wi-Fi Infrastructure 25
3.3 Functions and services 25
3.3.1 Basic Wi-Fi functions and services 25
3.3.2 Wi-Fi Display specific functions and services 26
3.4 Encoder/decoder characteristics 27
3.4.1 Audio 27
3.4.2 Video and Image Codecs 29
3.4.3 Stereoscopic 3D video 34
4 WFD SESSION FUNCTIONAL DESCRIPTION AND PROCEDURES 35
4.1 Reference Model 35
4.2 WFD connection setup, WFD session establishment and management functions 36
4.3 WFD R1 Discovery 37
4.3.1 WFD R1 Service Discovery 37
4.3.2 WFD R1 Device Discovery 37
4.4 WFD R2 Discovery 38
4.4.1 WFD R2 Service Discovery 38
4.4.2 WFD R2 Device Discovery 39
4.5 WFD Connection Setup 40
4.5.1 Connectivity Scheme Resolution 40
4.5.2 Establishing a WFD Connection using Wi-Fi P2P 41
4.5.3 Establish a WFD Connection using TDLS 43
4.5.4 Establishing a TCP connection 43
4.6 WFD Capability Negotiation 44
4.7 Link content protection setup 45
4.8 WFD session establishment 47
4.8.1 WFD Session between WFD R2 devices over a BSS infrastructure connection 49
4.8.2 WFD Session between a WFD R2 Device and a WFD R1 Device 51
4.9 Coupled Sink Operation 51
4.9.1 Primary and Secondary Sink Coupling 52
4.9.2 WFD Device Discovery 54
4.9.3 WFD Service Discovery 54
4.9.4 WFD Device Pairing 54
4.9.5 WFD capability negotiation 55
4.9.6 WFD Session establishment 55
4.10 AV streaming and control 55
4.10.1 Time synchronization 55
4.10.2 AV streaming 56
4.10.3 AV Encoding Rate Control 58

4.10.4 AV Session Control ... 60
4.10.5 WFD video recovery ... 60
4.10.6 Switching of Transport Protocol in a WFDS Session ... 60
4.11 User Input Back Channel ... 63
4.11.1 UIBC Data Encapsulation ... 64
4.11.2 UIBC Establishment and Maintenance ... 65
4.11.3 UIBC Input Body ... 66
4.12 WFD Session and WFD Connection Termination ... 70
4.12.1 Termination of a WFD Connection Using Wi-Fi P2P ... 70
4.12.2 Termination of a WFD Connection Using TDLS ... 70
4.12.3 Termination of WFD connection Using BSS infrastructure ... 70
4.13 Persistent WFD Groups ... 70
4.13.1 Persistent WFD Group over Wi-Fi P2P ... 71
4.13.2 Persistent WFD Group over TDLS ... 71
4.14 WFD MAC Procedures -- Concurrency ... 71
4.14.1 Concurrent WLAN access with Wi-Fi P2P ... 71
4.14.2 Concurrent WLAN access with TDLS ... 71
4.15 WFD Standby ... 71
4.16 Direct Streaming and Transport of Auxiliary Streams ... 73
4.16.1 Overview ... 74
4.16.2 Capability Negotiation and Session Setup ... 74
4.16.3 AV Streaming Switch ... 76
4.16.4 Auxiliary Stream over MPEG2-TS ... 78
4.17 Backward Compatibility ... 78
4.17.1 Device/Service Discovery ... 78
4.17.2 WFD Connection Establishment ... 79
4.17.3 WFD Capability Negotiation ... 79
4.17.4 WFD Session Establishment ... 79

5 FRAME FORMATS AND SERVICE DISCOVERY INFORMATION ... 80
5.1 WFD Information Element ... 80
5.1.1 WFD IE Format ... 80
5.1.2 WFD Device Information Subelement ... 81
5.1.3 Associated BSSID Subelement ... 83
5.1.4 Coupled Sink Information Subelement ... 83
5.1.5 WFD Video Formats Subelement ... 83
5.1.6 WFD 3D Video Formats Subelement ... 89
5.1.7 WFD Audio Formats Subelement ... 89
5.1.8 WFD Extended Capability Subelement ... 91
5.1.9 Local IP Address Subelement ... 92
5.1.10 WFD Session Information Subelement ... 92
5.1.11 Alternative MAC Address Subelement ... 93
5.1.12 WFD R2 Device Information Subelement ... 94
5.2 Management Frames ... 94
5.2.1 Beacon Frame Format ... 95
5.2.2 Probe Request Frame Format ... 95
5.2.3 Probe Response Frame ... 96
5.2.4 Association/Reassociation Request Frame ... 96
5.2.5 Association/Reassociation Response Frame ... 97
5.2.6 P2P Public Action Frames ... 97
5.2.7 Service Discovery Action Frames ... 101
5.3 WFD R2 Command ... 101
5.4 WFD R2 Service Discovery Information ... 101
5.4.1 WFD R2 Video Formats ... 102
5.4.2 WFD R2 Aux Stream Formats ... 109
5.4.3 WFD R2 Audio Formats ... 110
5.4.4 WFD R2 Extended Capabilities bitmap ... 113
5.5 TXT Record ... 114

6 RTSP BASED WFD CONTROL PLANE ... 116
6.1 RTSP data structures ... 116
6.1.1 ABNF Definitions ... 116
6.1.2 wfd-audio-codecs ... 116
6.1.3 wfd-video-formats ... 117
6.1.4 wfd-3d-formats ... 118
6.1.5 wfd-content-protection ... 118
6.1.6 wfd-display-edid ... 118
6.1.7 wfd-coupled-sink ... 120
6.1.8 wfd-trigger-method ... 120
6.1.9 wfd-presentation-url ... 120
6.1.10 wfd-client-rtp-ports ... 121
6.1.11 wfd-route ... 122
6.1.12 wfd-I2C ... 122
6.1.13 wfd-av-format-change-timing ... 122
6.1.14 wfd-preferred-display-mode ... 123
6.1.15 wfd-uibc-capability ... 123
6.1.16 wfd-uibc-setting ... 123
6.1.17 wfd-standby-resume-capability ... 123
6.1.18 wfd-standby ... 124
6.1.19 wfd-connector-type ... 124
6.1.20 wfd-idr-request ... 125
6.1.21 wfd2-audio-codecs ... 125
6.1.22 wfd2-video-formats ... 125
6.1.23 wfd2-aux-stream-formats ... 126
6.1.24 wfd2-direct-streaming-mode ... 127
6.1.25 wfd2-transport-switch ... 127
6.1.26 wfd2-buffer-length ... 127
6.1.27 wfd2-audio-playback-status ... 128
6.1.28 wfd2-video-playback-status ... 128
6.1.29 wfd2-cta-datablock-collection ... 129
6.1.30 wfd2-cta-audio-infoframe ... 131
6.2 WFD RTSP methods ... 131
6.2.1 WFD RTSP OPTIONS ... 131
6.2.2 GET_PARAMETER ... 133
6.2.3 SET_PARAMETER ... 133
6.2.4 SETUP ... 134
6.2.5 PLAY ... 134
6.2.6 PAUSE ... 135
6.2.7 TEARDOWN ... 135
6.3 RTSP Parameters ... 135
6.4 RTSP Messages ... 137
6.4.1 RTSP M1 Message ... 139
6.4.2 RTSP M2 Message ... 139
6.4.3 RTSP M3 Message ... 140
6.4.4 RTSP M4 Message ... 140
6.4.5 RTSP M5 Message ... 142
6.4.6 RTSP M6 Message ... 143
6.4.7 RTSP M7 Message ... 143
6.4.8 RTSP M8 Message ... 144
6.4.9 RTSP M9 Message ... 144
6.4.10 RTSP M10 Message ... 145
6.4.11 RTSP M11 Message ... 145
6.4.12 RTSP M12 Message ... 145
6.4.13 RTSP M13 Message ... 145
6.4.14 RTSP M14 Message ... 146
6.4.15 RTSP M15 Message ... 146
6.4.16 RTSP M16 Message ... 147

6.5 RTSP Timeout ... 147
6.5.1 WFD keep-alive ... 147
6.5.2 Timeout before RTSP Procedure ... 148
6.6 RTSP Syntax remark ... 148
6.6.1 CSeq ... 148
6.6.2 Delimiter for parameters ... 148
6.6.3 Message header ... 149
6.6.4 Content-Encoding ... 149
6.6.5 Case Sensitivity ... 149
6.6.6 Content-Length and Content-Type ... 149

7 REMOTE I2C READ/WRITE MESSAGING TRANSACTION ... 150
7.1 Remote_I2C_Read_Request ... 151
7.2 Remote_I2C_Write_Request ... 151
7.3 Remote_I2C_Read_Reply_Ack ... 152
7.4 Remote_I2C_Read_Reply_Nak ... 152
7.5 Remote_I2C_Write_Reply_Ack ... 152
7.6 Remote_I2C_Write_Reply_Nak ... 153

8 PREFERRED DISPLAY MODE ... 154

APPENDIX A MPEG SYSTEM LAYER (INFORMATIVE) ... 155

APPENDIX B MPEG-TS PARAMETERS FOR AUDIO, VIDEO AND AUXILIARY ELEMENTARY STREAMS (NORMATIVE) 156
B.1 Encapsulation of MPEG2-TS into RTP Packets ... 163
B.2 PES Structure and Blending Operation for Auxiliary Streams ... 164
B.2.1 Blending Operation ... 167
B.2.2 Carriage of JPEG Image over MPEG2-TS Transport Stream ... 169
B.2.3 Carriage of PNG Image over MPEG2-TS Transport Stream ... 170
B.3 WFD R2 LPCM Formats and Packet Structures ... 172
B.3.1 PES Structure for LPCM Streams ... 172
B.3.2 Speaker Location Names used for Multichannel Audio Streaming ... 175
B.3.3 Multi-channel Audio Formats Information ... 177
B.4 DTS-HD Audio Bitstream Constraints ... 177
B.4.1 DTS-HD MPEG-TS Parameters and Requirements ... 177
B.4.2 DTS-HD Elementary Stream Encapsulation ... 178

APPENDIX C RECOMMENDATIONS FOR SATISFYING THE HDCP 2.X LOCALITY CHECK (INFORMATIVE) ... 179

APPENDIX D H.264 AND H.222 USAGE DETAIL (NORMATIVE) ... 180
D.1 Slice Usage ... 180
D.1.1 Multiple Slices in a Picture ... 180
D.2 Frame Packing Arrangement SEI ... 180
D.3 SEI and VUI ... 182
D.4 H.222 usage ... 182
D.4.1 PTS/DTS for video stream ... 182
D.4.2 PAT/PMT ... 182
D.4.3 Program and program element descriptors in PMT ... 183

APPENDIX E RTSP MESSAGE EXAMPLES (INFORMATIVE) ... 185
E.1 From RTSP M1 to M7 without errors ... 185
E.2 RTSP M4 with error case ... 186
E.3 Example RTSP Messages for TCP Mode ... 187

APPENDIX F HDCP 2.X NOTATION MAPPING TO WFD DEVICE HDCP IMPLEMENTATION REVISION ... 191

APPENDIX G CTA AUDIO EXAMPLE (INFORMATIVE) ... 192

List of tables

Table 1. Definitions 15
Table 2. Abbreviations and acronyms 16
Table 3. Functions and services 26
Table 4. Optional audio CODEC formats 28
Table 5. List of supported audio CODEC formats for audio stream from Source to Sink in forward channel 28
Table 6. Wi-Fi Display H.264 profiles 29
Table 7. Video codecs and profiles applicable for transcoding mode 30
Table 8. Video codecs and profiles applicable for Non-Transcoding Mode 31
Table 9. Codecs and profiles applicable for auxiliary streams 31
Table 10. Supported video resolutions and frame rates 32
Table 11. Connectivity scheme resolution 41
Table 12. Connectivity resolution scheme when using Secondary Sinks 54
Table 13. Input Category Code 64
Table 14. Generic input message format 66
Table 15. Generic Input Type ID for user inputs of the Generic Category 66
Table 16. Describe Field of the Generic Input Message for Left Mouse Down/Touch Down 67
Table 17. Describe Field of the Generic Input Message for Left Mouse Move/Touch Move 67
Table 18. Describe Field of the Generic Input Message for Zoom 67
Table 19. Describe Field of the Generic Input Message for Vertical Scroll 68
Table 20. Describe Field of the Generic Input Message for Horizontal Scroll 68
Table 21. Describe Field of the Generic Input Message for Rotate 68
Table 22. HIDC message format 69
Table 23. HID Input Path 69
Table 24. HID Type 69
Table 25. WFD IE Format 80
Table 26. General Format of a WFD Subelement 80
Table 27. WFD Subelement ID Definitions 80
Table 28. WFD Device Information Subelement 81
Table 29. WFD Device Information Field 82
Table 30. Associated BSSID Subelement 83
Table 31. Coupled Sink Information Subelement 83
Table 32. Coupled Sink Status Bitmap 83
Table 33. WFD Video Formats Subelement 84
Table 34. Supported CEA Resolution/Refresh Rates 85
Table 35. Supported VESA Resolution/Refresh Rates 85
Table 36. Supported HH Resolutions/Refresh Rates 86
Table 37. Display Native Resolution Refresh Rate 87
Table 38. Profiles Bitmap 87
Table 39. Maximum H.264 Level Supported 88
Table 40. Slice Encoding parameters bitmap 88
Table 41. Video Frame Rate Control Support Bitmap 89
Table 42. WFD Audio Formats Subelement 89
Table 43. LPCM Modes bitmap 90
Table 44. AAC Codec bitmap 91
Table 45. AC3 Modes bitmap 91
Table 46. WFD Extended Capability Subelement 91
Table 47. WFD Extended Capabilities Bitmap 92
Table 48. Local IP Address Subelement 92
Table 49. WFD Session Information Subelement 93
Table 50. WFD Device Info Descriptor field 93
Table 51. Alternative MAC Address Subelement 94
Table 52. WFD R2 Device Information Subelement 94
Table 53. WFD R2 Device Information 94
Table 54. WFD Subelements in a Beacon Frame 95
Table 55. WFD Subelements in a Probe Request Frame 95
Table 56. Tunneled Probe Request/Response 95

Table 57. WFD Subelements in a Probe Response Frame ... 96
Table 58. WFD Subelements in an Association/Reassociation Request Frame ... 97
Table 59. WFD Subelements in an Association/Reassociation Response Frame ... 97
Table 60. WFD Subelements in a GO Negotiation Request Frame ... 98
Table 61. WFD Subelements in a GO Negotiation Response Frame ... 98
Table 62. WFD Subelements in a GO Negotiation Confirmation Frame ... 98
Table 63. WFD Subelements in a P2P Invitation Request Frame ... 99
Table 64. WFD Subelements in a P2P Invitation Response Frame ... 99
Table 65. WFD Subelements in a Provision Discovery Request Frame ... 100
Table 66. WFD Subelements in a Provision Discovery Response Frame ... 100
Table 67. WFD R2 Command ... 101
Table 68. WFD R2 TCP Connection Request Command ... 101
Table 69. WFD R2 Capability Definitions used for Service Discovery Information ... 101
Table 70. WFD R2 Video Formats Information ... 102
Table 71. Supported CEA Resolution/Refresh Rates ... 103
Table 72. Supported VESA Resolution/Refresh Rates ... 104
Table 73. Supported HH Resolutions/Refresh Rates ... 105
Table 74. Display Native Resolution Refresh Rate ... 106
Table 75. WFD Video Codec Info Descriptor field ... 106
Table 76. Video Codec Type field values in WFD R2 Video Codec Info Descriptor ... 107
Table 77. Video Codec Profile Bitmap values in WFD Video Codec Info Descriptor ... 107
Table 78. Video Codec Level Bitmap values in WFD R2 Video Codec Info Descriptor ... 108
Table 79. WFD R2 Aux Stream Formats Information ... 109
Table 80. Aux Stream Codecs Bitmap ... 109
Table 81. Reference Overlay Resolution Refresh Rate ... 110
Table 82. WFD R2 Audio Formats Information ... 110
Table 83. LPCM Modes bitmap for WFD R2 devices ... 112
Table 84. CTA Audio Modes Bitmap ... 113
Table 85. AAC-ELDv2 Modes Bitmap ... 113
Table 86. WFD R2 Extended Capabilities Bitmap ... 114
Table 87. WFD R2 Synchronous Streaming Information field bit mapping ... 114
Table 88. wfd-url0 and wfd-url1 values in wfd-presentation-url ... 121
Table 89. wfd-client-rtp-ports parameter values in M3 Response message ... 121
Table 90. wfd-client-rtp-ports parameter values in M3 Response message and corresponding values in the subsequent M4/M6 Request message ... 122
Table 91. Display connector type field ... 124
Table 92. CTA data block header ... 129
Table 93. CTA Extended Tag Code ... 129
Table 94. CTA audio formats allowed in WFD R2 ... 130
Table 95. RTSP methods that the WFD Source and/or WFD Sink can invoke ... 132
Table 96. Reason Code for RTSP M4 Response in WFD ... 133
Table 97. Summary of RTSP parameters ... 135
Table 98. List of defined RTSP messages and their identifiers ... 138
Table 99. Remote_I2C_Read_Request ... 151
Table 100. Remote_I2C_Write_Request ... 152
Table 101. Remote_I2C_Read_Reply_Ack ... 152
Table 102. Remote_I2C_Read_Reply_Nak ... 152
Table 103. Remote_I2C_Write_Reply_Ack ... 152
Table 104. Remote_I2C_Write_Reply_Nak ... 153
Table 105. MPEG-TS parameters for supported audio, video and image formats ... 156
Table 106. PES structure and field values for 48ksps/44.1ksps 16bits 2-ch LPCM ... 157
Table 107. PES structure and field values for Dolby AC-3 ... 158
Table 108. PES structure and field values for E-AC-3 ... 160
Table 109. PES structure and field values for AAC-LC ... 161
Table 110. RTP encapsulation of MPEG-TS ... 163
Table 111. PES structure and field values for Auxiliary Stream ... 164
Table 112. Alpha selection and associated blending modes ... 167
Table 113. JPEG stream descriptor ... 169

Table 114. PNG stream descriptor ... 170
Table 115. PES Structure for WFD R2 LPCM audio ... 172
Table 116. Quantization_word_length ... 174
Table 117. Audio_sampling_frequency ... 174
Table 118. Number_of_audio_channels ... 174
Table 119. Bitmap for the Speaker Location Names ... 176
Table 120. Syncwords associated with DTS-HD audio ... 178
Table 121. Frame packing arrangement SEI syntax element and values at Wi-Fi Display ... 181
Table 122. RTSP message example-1 ... 185
Table 123. RTSP message example-2 ... 187
Table 124. RTSP M3 message to query available buffer length ... 187
Table 125. RTSP M4 and M6 messages for switching to TCP ... 188
Table 126. RTSP M4 and M6 message for switching to UDP ... 188
Table 127. RTSP M3 message for requesting audio buffer information ... 189
Table 128. RTSP M3 message for requesting video buffer information ... 189
Table 129. HDCP 2.x Notation Mapping ... 191
Table 130. Example of RTSP messages to exchange CTA Audio parameters between WFD R2 Source and WFD R2 Sink ... 192
Table 131. Data Block Header Example ... 192
Table 132. AC-3 Short Audio Descriptor Example ... 193
Table 133. Enhanced AC-3 Short Audio Descriptor Example ... 193
Table 134. DTS-HD Short Audio Descriptor Example ... 193
Table 135. L-PCM 3D Short Audio Descriptor Example ... 193
Table 136. Room Configuration Data Block Example ... 194
Table 137. Speaker Location Data Block Example ... 194
Table 138. Audio InfoFrame Example ... 195
Table 139. Room Configuration Data Block Example 2 ... 195
Table 140. Speaker Location Data Block Example 2 ... 196
Table 141. Speaker Location Data Block Example 3 ... 196

List of figures

Figure 1. Logical data and control plane connections 20
Figure 2. Example of WFD topology 21
Figure 3. Audio-only WFD Session 24
Figure 4. Video-only WFD Session 24
Figure 5. Audio and video WFD Session 24
Figure 6. WFD Session under Coupled Sink Operation 24
Figure 7. WFD Connection using Wi-Fi P2P 25
Figure 8. TDLS topology 25
Figure 9. Discovery and WFD connection over Wi-Fi infrastructure 25
Figure 10. Reference model for session management in WFD devices 35
Figure 11. Reference model audio/video payload processing 36
Figure 12. Tunneled Probing when TDLS is used 38
Figure 13. WFD capability negotiation (or coupling) flow using RTSP 45
Figure 14. Time-line of a WFD session 47
Figure 15. DNS-SD for active subscription and solicited advertising 50
Figure 16. Timeline of a WFD Session with a Secondary Sink 52
Figure 17. Eliminating Network Jitter using RTP 56
Figure 18. WFD AV packet format 57
Figure 19. Typical message flow for switching to TCP mode in a WFD R2 Session and switching back to UDP mode 63
Figure 20. Encapsulations of User Inputs over TCP/IP 64
Figure 21. UIBC capability negotiation 65
Figure 22. UIBC update 66
Figure 23. Overview of Audio/Video and Auxiliary Streams over MPEG2-TS 74
Figure 24. Message Sequence for Capability Negotiation to Setup Direct Streaming and Auxiliary Streams in a WFD Session 76
Figure 25. Message sequence for AV Streaming Switch 77
Figure 26. Block Structure of EDID data 119
Figure 27. Triggering the WFD Sink to send an RTSP Request message 143
Figure 28. Remote I2C transaction example 151
Figure 29. Overview of the H.222.0 Systems Layer – Transmission, Multiplexing and Synchronization 155
Figure 30. Example of recommended encapsulation of MPEG2-TS packets 163
Figure 31. Example of recommended encapsulation of MPEG2_TS packets 164
Figure 32. Structure of JPEG Access Units in PES Packets 170
Figure 33. Structure of PNG Access Units in PES Packets 171
Figure 34. Layout of a Chunk 171
Figure 35. PES structure with X MCA frames in the payload 175
Figure 36. Typical composition of a PES payload 178

1 Introduction

This document is the technical specification for Wi-Fi CERTIFIED Miracast™, a mechanism to discover, pair, connect and render multimedia content sourced from a Wi-Fi Display Source at a Wi-Fi Display Sink. This specification defines the architecture and a set of protocols between a Wi-Fi Display Source and Wi-Fi Display Sink.

1.1 Scope

The content in this specification is designed to ensure interoperability between a Wi-Fi Display Source and Sink and provide guidelines to enhance performance and user experience. It specifies how Wi-Fi Display (WFD) Devices discover, pair and establish Wi-Fi Display Session(s) with other WFD Device(s). This specification includes a set of appendices that specify implementation requirements and provide implementation guidelines.

The following system requirement areas are covered:

- WFD Device discovery
- WFD Service discovery
- WFD capability discovery
- WFD Connection establishment
- WFD Session establishment
- Payload formats for video and audio streams from WFD Source to WFD Sink
- Transport and multiplex protocol for video and audio payload
- Link Content Protection
- WFD Session termination
- Persistent WFD Groups
- Direct Streaming Operation

1.2 References

Knowledge of the documents listed in this section is required for understanding this technical specification. If a reference includes a date or a version identifier, only that specific version of the document is required. If the listing includes neither a date nor a version identifier, then the latest version of the document is required. In the event of a conflict between this specification and the following referenced documents, the contents of this specification take precedence.

[1] ITU-T Rec. H.264 (03/2010)

[2] ITU-T Rec. H.222.0 (10/2014)

[3] RFC 3550, “RTP: A Transport Protocol for Real-Time Applications”, July 2003.

[4] RFC 2250, “RTP Payload Format for MPEG-1/MPEG-2 Video”, Jan. 1998.

[5] Section 4.5, 4.6 and 4.7 at “Guideline of Transmission and control for DVD-Video/Audio through IEEE1394 Bus”, version 1.0, September 2002: http://www.dvdforum.org/images/Guideline1394V10R0_20020911.pdf

[6] RFC 3551, “RTP Profile for Audio and Video Conferences with Minimal Control”

[7] Wi-Fi Alliance, Wi-Fi Peer-to-Peer (P2P) Technical Specification, version 1.7

[8] Wi-Fi Alliance, Wi-Fi Simple Configuration Specification, version 2.0.0, December 2010

[9] ITU-T Rec. H.262 | ISO/IEC 13818-2

[10] E-EDID 1.4: “VESA Enhanced Extended Display Identification Data Standard”, Release A, revision 2, September 25, 2006

[11] "A DTV Profile for Uncompressed High Speed Digital Interfaces", CTA-861-E, March, 2008.

[12] VESA E-DDC v1.2: "VESA Enhanced Display Data Channel (E-DDC) Standard", Version 1.2, December 26, 2007

[13] IEEE P802.11z-2010

[14] IEEE 802.11-2012

[15] IEEE 802.1AS-2011, "Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks", March 30, 2011)

[16] Wi-Fi Alliance, Wi-Fi Protected Access, Version 3.1, August, 2004

[17] ISO/IEC 13818-7: "Information technology – Generic coding of moving pictures and associated audio information – Part7: Advanced Audio Coding (AAC)", (October 15, 2004)

[18] "High Bandwidth Digital Content Protection System Interface Independent Adaptation", Revision 2.0 October, 2008, http://www.digital-cp.com/hdcp_technologies

[19] "Errata – Summary of Errata and Clarifications to the HDCP Interface Independent Adaptation Specification Rev 2.0" http://www.digital-cp.com/hdcp_technologies

[20] RFC2326, "Real Time Streaming Protocol (RTSP)"

[21] "Universal Serial Bus Specification - USB Device Class Definition for Human Interface Devices", Version 1.11, Jun. 2001

[22] "Bluetooth Special Interest Group (SIG) - Human Interface Device (HID) Profile", May. 2003

[23] RFC 20, "ASCII format for Network Interchange", ANSI X3.4-1968, October 16, 1969

[24] ISO/IEC 8859-1:1998, "8-bit single-byte coded graphic character sets, Part 1: Latin alphabet No. 1"

[25] RFC2234, "Augmented BNF for Syntax Specifications: ABNF", November 1997

[26] IETF "STD7: Transmission Control Protocol", this is based on RTF793 and updated by RFC1122, 3168 and 6093.

[27] "High Bandwidth Digital Content Protection System Interface Independent Adaptation", Revision 2.1 July, 2011 http://www.digital-cp.com/hdcp_technologies

[28] IEEE P802.11v-2011

[29] RFC 6335, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", August 2011

[30] Service Name and Transport Protocol Port Number Registry, IANA, April 9, 2012 http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml

[31] HDCP License Agreement, August 31, 2011 http://www.digital-cp.com/files/static_page_files/26D315BF-1A4B-B294-D04BB484EE81591E/HDCP%20License%20Agreement0831_2011_clean%20_2_.pdf

[32] HDCP 2.0 addendum to HDCP License Agreement <http://digital-cp.com/files/static_page_files/62BFCBA3-1A4B-B294-D09C885021187455/HDCP%202%200%20Addendum_Clean_FINAL2_04_30_11_ver2.pdf

[33] "VESA and Industry Standards and Guidelines for Computer Display Monitor Timing", Version 1.0, Revision 11, May 1, 2007

[34] "High Bandwidth Digital Content Protection System Interface Independent Adaptation", Revision 2.2, October 16, 2012, http://www.digital-cp.com/hdcp_technologies

[35] HDCP License Agreement, November 2, 2012, http://www.digital-cp.com/files/static_page_files/C2A8EA1A-1A4B-B294-D049C3234A800E09/HDCP%20License%20Agreement1102_2012_clean%20_2_.pdf

[36] ITU-T Rec. H.222.0 (2014) Amendment 1: Delivery of timeline for external data (04/2015)" ATSC A/52:2012, "Digital Audio Compression (AC-3, E-AC-3) Standard, March 2012

[37] ANSI/CEA Standard: A DTV Profile for Uncompressed High Speed Digital Interfaces, ANSI/CTA-861-F, August 2013

[38] "VESA and Industry Standards and Guidelines for Computer Display Monitor Timing (DMT)", Version 1.0, Revision 13, February 8, 2013

[39] ITU-T Rec H.265/ISO/IEC 23008-2/MPEG-H Part 2

[40] JPEG Standard, ISO/IEC 10918, ITU-T T.81/T.83/T.84/T.86

[41] ECMA TR/98, JPEG File Interchange Format (JFIF),1'st Edition, June 2009

[42] ISO/IEC 15948:2003 - Information technology -- Computer graphics and image processing -- Portable Network Graphics (PNG): Functional specification, W3C Recommendation 10 November 2003 (available at http://www.w3.org/TR/PNG/)

[43] RFC4571 - Framing Real-time Transport Protocol (RTP) and RTP Control Protocol (RTCP) Packets over Connection-Oriented Transport

[44] Compositing and Blending Level 1, W3C Candidate Recommendation, 13 January 2015 at http://www.w3.org/TR/compositing-1/

[45] HEVC Screen Content Coding Draft Text 3", JCTVC-T1005, R. Joshi et al, http://phenix.int-evry.fr/jct/doc_end_user/documents/20_Geneva/wg11/JCTVC-T1005-v2.zip

[46] "HID Usage Tables", Version 1.12, October 28, 2004

[47] Draft CTA-861-G Advanced Audio Extensions, June 2015

[48] ISO/IEC 62574, "Audio, video and multimedia systems – General channel assignment of multichannel audio, 488 April 7, 2014

[49] Multicast DNS, RFC 6762, Feb 2013

[50] DNS-Based Service Discovery, RFC 6763, Feb 2013

[51] Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry, RFC 6335, Aug 2011

[52] Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB Systems (DVB Document A038), August 2015

[53] ETSI TS 102 366, "Digital Video Broadcasting (DVB); Digital Audio Compression (AC-3, Enhanced AC-3) Standard"

[54] ETSI TS 101 154, “Digital Video Broadcasting (DVB); Specification for the use of Video and Audio Coding in Broadcasting Applications based on the MPEG-2 Transport Stream”

[55] ETSI TS 102 114 “DTS Coherent Acoustics; Core and Extensions with Additional Profiles”

[56] DECE “DMedia: Common File Format & Media Format Formats Specification version 2.2”, http://www.uvcentral.com/page/ultraviolet-online-guide-homepage

[57] ISO/IEC 14496-3 Amd.3 Cor.1, “Information Technology – Coding of audio-visual objects – Part 3: Audio; Transport of unified speech and audio coding (USAC)”

2 Definitions and acronyms

2.1.1 Shall/should/may/might word usage

The words shall, should, and may are used intentionally throughout this document to identify the requirements for the Wi-Fi Display specification.

The word *shall* indicates a mandatory requirement. All mandatory requirements must be implemented to assure interoperability with other Wi-Fi Display products.

The word *should* denotes a recommended approach or action.

The word *may* indicates a permitted approach or action with no implied preference.

The words *might* and *can* indicate a possibility or suggestion.

2.1.2 Definitions

The following definitions are applicable to this specification.

Table 1. Definitions

Term	Definition
AP	Wi-Fi Access Point
Coupled Sink Operation	An operation hereby a WFD Source transmits video content to a Primary Sink and transmits audio content to a Secondary Sink, after coupling is established between the Primary Sink and the Secondary Sink.
Coupling	A procedure to exchange capabilities between a Primary Sink and a Secondary Sink so that a WFD Source can know the presence of both the Primary Sink and the Secondary Sink if the one of them is discovered via WFD Device Discovery. Once this procedure is done, the Primary Sink and the Secondary Sink is said to be 'Coupled' and both are in 'Coupled state'.
Paired	A WFD Source and a WFD Sink are said to be paired if they have successfully completed WFD Capability Negotiation.
Primary Sink	Either an R1 Primary Sink or an R2 Primary Sink. A Primary Sink may support Coupled Sink Operation.
R1 Primary Sink	A device that supports rendering video content only or both audio and video content using the R1 protocol defined herein. If an R2 Primary Sink is connected to a WFD R1 Source, the R2 Primary Sink acts as an R1 Primary Sink.
R2 Primary Sink	A device that supports rendering video content only or both audio and video content using the R2 protocol defined herein.
R2 Secondary Sink	A device that supports rendering audio content only using the R2 protocol defined herein.
Rendering	The process of generating an image or sound from a set of data
Secondary Sink	A Secondary Sink may support Coupled Sink Operation. A Secondary Sink also supports rendering audio content directly from a WFD Source regardless of the coupling status with Primary Sink or the presence of the Primary Sink.
Specification	Within this document, the term specification is used to refer to this document, i.e., the Wi-Fi Display Technical Specification
WFD Connection	Layer 2 connection to be used for WFD.
WFD R1 Device	A synonym for Wi-Fi Display R1 Device.
WFD R2 Device	A synonym for Wi-Fi Display R2 Device.
WFD Session	A Wi-Fi Display connection between a WFD Source and a WFD Sink.

Term	Definition
	WFD Topology: The arrangement in which the Wi-Fi Display Source and Sink are connected to each other for a WFD Session and (in some cases) to other Wi-Fi devices.
Wi-Fi Display Device (or WFD Device)	Either a WFD Source or a WFD Sink
Wi-Fi Display R1 Device	Either a WFD R1 Source or a WFD R1 Sink.
Wi-Fi Display R1 Sink (or WFD R1 Sink)	A device that receives multimedia content from a WFD R1 Source over a Wi-Fi link and renders it. If a WFD R2 Sink is connected to a WFD R1 Source, the WFD R2 Sink acts as a WFD R1 Sink. A WFD R1 Sink is either an R1 Primary Sink or an R2 Secondary Sink.
Wi-Fi Display R1 Source (or WFD R1 Source)	A device that supports streaming multimedia content to a WFD R1 Sink(s) over a Wi-Fi link using the R1 protocol defined in [42]. If a WFD R2 Source is connected to a WFD R1 Sink, the WFD R2 Source acts as a WFD R1 Source.
Wi-Fi Display R2 Device	Either a WFD R2 Source or a WFD R2 Sink.
Wi-Fi Display R2 Sink (or WFD R2 Sink)	A device that receives multimedia content from a WFD R2 Source over a Wi-Fi link and renders it using the R2 protocol defined herein. A WFD R2 Sink is either an R2 Primary Sink or an R2 Secondary Sink.
Wi-Fi Display R2 Source (or WFD R2 Source)	A device that supports streaming multimedia content to a WFD R2 Sink(s) over a Wi-Fi link using the R2 protocol defined herein.
Wi-Fi Display Sink (or WFD Sink)	Either a WFD R1 Sink or a WFD R2 Sink. A WFD Sink is either a Primary Sink or a Secondary Sink.
Wi-Fi Display Sink dongle (or WFD Sink dongle)	A WFD Sink that supports outputting a video and/or audio signal to an external rendering device
Wi-Fi Display Source (or WFD Source)	Either a WFD R1 Source or a WFD R2 Source.
Wi-Fi P2P	A protocol that provides Wi-Fi device-to-device connectivity including discovery and pairing, without requiring an AP.

2.1.3 Abbreviations and acronyms

Table 2 defines the acronyms used throughout this document. Some acronyms are commonly used in publications and standards defining the operation of wireless local area networks, while others have been generated by Wi-Fi Alliance.

Table 2. Abbreviations and acronyms

Acronyms	Definition
(Wi-Fi) P2P	(Wi-Fi) Peer-to-peer (used as a name of a topology or a technology, which has a certification program name as Wi-Fi Direct™)
AAC	Advanced Audio Coding
ADTS	Audio Data Transport Stream
AKE	Authentication and Key Interchange
AP	Wi-Fi Access Point
ASCII	American Standard Code for Information Exchange
ASO	Arbitrary Slice Ordering
AU	Access Unit
AV	Audio Video
BSS	Basic Service Set
BSSID	Basic Service Set Identifier
CABAC	Context Adaptive Binary Arithmetic Coding

Acronyms	Definition
CAVLC	Context Adaptive Variable Length Coding
Cb	Blue difference Chroma component
CBP	(H.264) Constrained Baseline Profile
CD	Compact Disk
CEA	Consumer Electronics Association
CHP	Constrained High Profile
CODEC	COmpressorDECompressor
CP	Content Protection
Cr	Red difference Chroma component
DA	Destination Address
DNS-SD	DNS-Based Service Discovery
DTS	Decode Timestamp
DTV	Digital Television
E-DDC	Enhanced Display Data Channel
EDID	Extended Display Identification Data
FMO	Flexible Macroblock Ordering
GO	P2P Group Owner
GOP	Group of Pictures
gPTP	generalized Precision Time Protocol
HDCP	High-bandwidth Digital Content Protection
HDMI	High Definition Media Interface
HH	Handheld
HID	Human Interface Device
HIDC	Human Interface Device Class
IE	Information Element
IEC	International Electrotechnical Commission
IEEE	Institution of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
ISO	International Standards Organization
ITU	International Telecommunication Union
L2	OSI Layer 2, a Data Link Layer protocol
L3	OSI Layer 3, a Network Layer protocol
LAN	Local Area Network
LLC	Logical Link Control
LPCM	Linear Pulse Coded Modulation

Acronyms	Definition
MAC	Medium Access Control
MBAFF	Macroblock-Adaptive Frame-Field Coding
mDNS	Multicast DNS
MPEG	Moving Picture Experts Group
NAL	Network Abstraction Layer
OOB	Out-Of-Box
OUI	Organizationally Unique Identifier
PAT	Program Association Table
PC	Preferred Connectivity
PCR	Program Clock Reference
PES	Packetized Elementary Stream
PHY	Physical Layer
PicAFF	Picture Adaptive Frame/Field Coding
PID	Packet Identifier
PMT	Program Map Table
PSH	PUSH flag
PTR	Pointer
PTS	Presentation Timestamp
QP	Quantization Parameter
R1	Wi-Fi Display Technical Specification v1.0
R2	Wi-Fi Display Technical Specification v1.0
RFC	Request for Comment
RHP	Restricted High Profile
RS	Redundant Slices
RTP	Real-time Protocol
RTSP	Real Time Streaming Protocol
SCR	System Clock Reference
SEI	Supplemental Enhancement Information
SKE	Session Key Exchange
SNAP	SubNetwork Access Protoco;
SRV	Service
SSID	Service Set Identifier
STA	Non-AP Station
TCP	Transmission Control Protocol
TDLS	Tunneled Direct Link Setup (See [13]
TLV	Type-Length-Value

Acronyms	Definition
TOS	Type of Service
TS	Transport Stream
TXT	Text
UDP	Universal Datagram Protocol
UIBC	User Input Back Channel
URG	Urgent Flag
URI	Universal Resource Identifier
URL	Universal Resource Locator
USB	Universal Serial Bus
VCL	Video Coding Layer
VESA	Video Electronics Standards Association
WFD	Wi-Fi Display
WLAN	Wireless Local Area Network
WPA2™	Wi-Fi Protected Access® 2
WSD	Wi-Fi Display Service Discovery

3 WFD Architecture and Requirements

Figure 1 illustrates the functional blocks in the Wi-Fi Display data and control planes. The data plane consists of video codec (section 3.4.2 and 3.4.3), audio codec (section 3.4.1), PES packetization (Appendix B), the HDCP system 2.x (section 4.7), and MPEG2-TS over RTP/UDP/IP (section 4.10.2 and Appendix B). The control plane consists of RTSP over TCP/IP (section 6), remote I2C Read/Write (section 7), UIBC with HIDC and generic user input (section 4.11), and the HDCP session key establishment (section 4.7). The Wi-Fi P2P/TDLS block forms the layer-2 connectivity using either Wi-Fi P2P or TDLS as described in section 4.5.

Note: When using “HDCP system 2.x” or “HDCP 2.x” in this specification, “x” means 0, 1, 2, or higher numerical value. At the time of publication of this specification, HDCP 2.2 is the latest version.

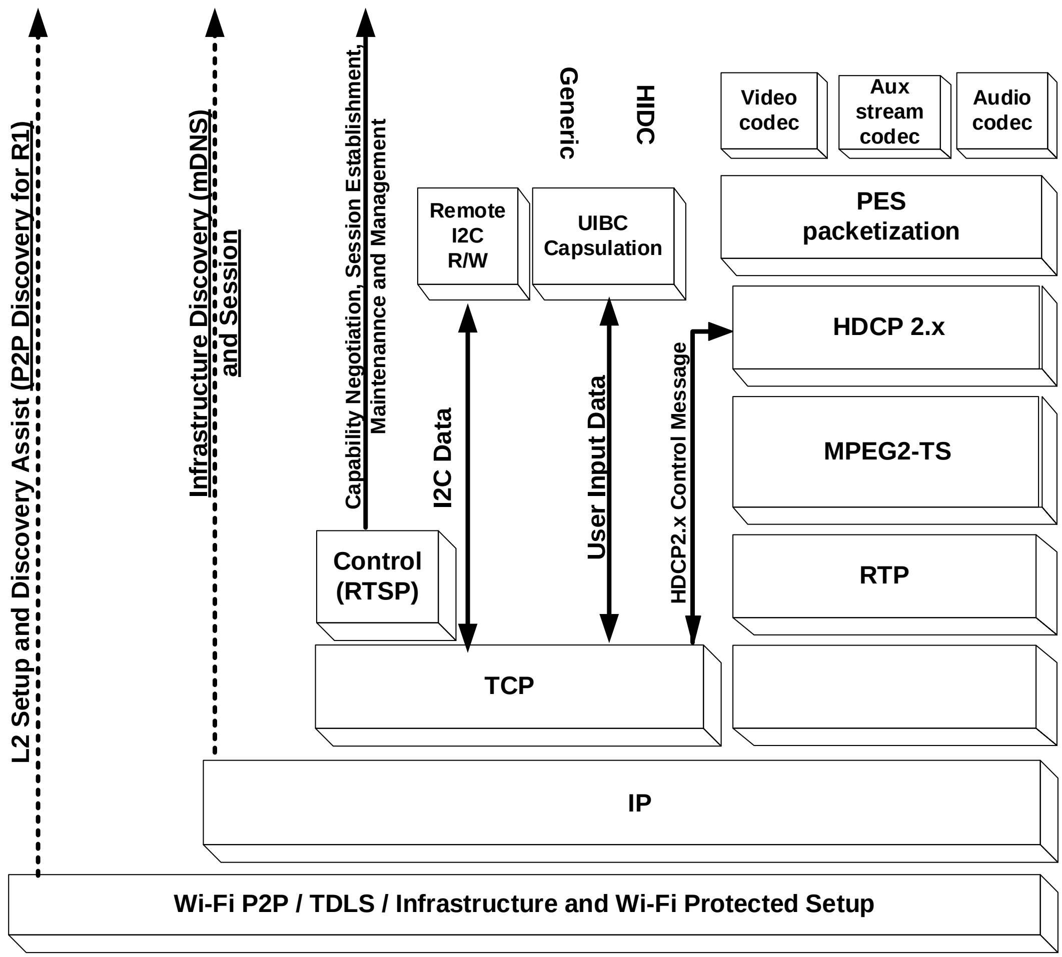


Figure 1. Logical data and control plane connections

3.1 WFD Source, Primary Sink, Secondary Sink and WFD Session

Figure 2 shows a simplified example of WFD Topology of the WFD devices in a WFD Session. Here one WFD Source and one Primary Sink are connected for AV streaming and stream control signaling. Other variations of the WFD Topologies and WFD Sessions are shown in the following sub-sections.

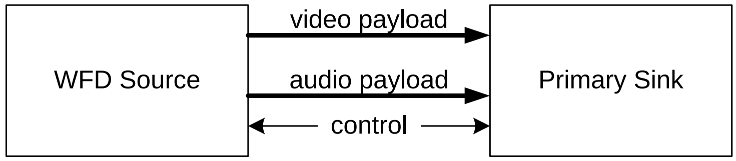


Figure 2. Example of WFD topology

3.1.1 WFD Source

A WFD Source shall support a single WFD Session. Support of more than one WFD Session is outside the scope of this specification.

During a WFD Session, a WFD Source shall transmit an MPEG2-TS ([2]) to one WFD Sink.

A WFD Source shall support transmitting an MPEG2-TS that contains multiplexed a single audio and a single video elementary streams.

A WFD Source may choose to transmit an MPEG2-TS that contains only a video elementary stream.

A WFD Source may choose to transmit an MPEG2-TS that contains only an audio elementary stream.

If a WFD Source is connected to a Primary Sink and depending on the capability of the Primary Sink or content itself to be streamed or user's choice, the WFD Source may transmit an MPEG2-TS that contains either 1) an audio elementary stream and a video elementary stream or 2) only a video elementary stream.

If a WFD Source is connected to a Primary Sink and depending on the capability of the Primary Sink or content itself to be streamed or user's choice, the WFD Source may transmit an MPEG2-TS that contains only an audio elementary stream.

If a WFD Source is connected to a Secondary Sink, the MPEG2-TS shall contain only an audio elementary stream.

A WFD R2 Source may transmit a content in its native encoding format without transcoding when the WFD R2 Sink supports the native format of that content.

A WFD R2 Source shall support changing the codecs, formats and profiles of the content streams during an active session without causing any termination to the ongoing Session.

If a WFD R2 Source is capable to transmit related content to two or more WFD R2 Sinks, the WFD R2 Source shall provide accurate timestamp information so that each WFD R2 Sink can render contents synchronously. Refer to Appendix B for details.

A WFD R2 Source may support concurrent Sessions over a P2P connection with two or more WFD R2 Sinks.

A WFD R2 Source may support concurrent Sessions over an IP network over Wi-Fi infrastructure with two or more WFD R2 Sinks when all WFD R2 devices are connected to the same IP subnet over Wi-Fi infrastructure.

A WFD R2 Source may transmit either landscape or portrait content to a rotation capable R2 Sink.

A WFD R2 Source may choose to setup more than one concurrent Session with a multi-mon capable R2 Sink

Table 3 summarizes the required capabilities of a WFD Source.

3.1.2 WFD Sink(s)

A WFD sink shall support a single WFD Session. Support of more than one WFD Session is outside the scope of this specification.

Two types of WFD sinks are defined, i.e., a Primary Sink and a Secondary Sink.

3.1.2.1 Primary Sink

During a WFD Session, a Primary Sink shall support receiving an MPEG2-TS from one WFD Source.

A Primary Sink shall support receiving an MPEG2-TS that contains multiplexed single audio elementary stream and single video elementary stream.

A Primary Sink shall support receiving an MPEG2-TS that contains only a video elementary stream.

A Primary Sink may support receiving an MPEG2-TS that contains only an audio elementary stream.

If a Primary Sink has an integrated video rendering function, the Primary Sink shall support one of following:

a. render the video content included in the received MPEG2-TS that contains multiplexed single audio elementary stream and single video elementary stream.
b. output the video content included in the received MPEG2-TS that contains multiplexed single audio elementary stream and single video elementary stream to an externally connected video rendering device.

If a Primary Sink supports both of (a) and (b) above, the Primary sink may choose one operation depending on local policy (e.g., detecting attachment or detachment of an external rendering device) or a user may choose one operation.

If a Primary Sink has an integrated video rendering function, the Primary Sink shall support one of following:

a. render the video content included in the received MPEG2-TS that contains only a video elementary stream.
b. output the video content included in the received MPEG2-TS that contains only a video elementary stream to an externally connected video rendering device.

If a Primary Sink supports both of (a) and (b) above, the Primary Sink may choose one operation depending on local policy (e.g., detecting attachment or detachment of an external rendering device) or a user may configure the Primary Sink to choose one operation.

If a Primary Sink does not have an integrated video rendering function, the Primary Sink shall support outputting the video content included in the received MPEG2-TS that contains multiplexed single audio elementary stream and single video elementary stream to an externally connected video rendering device.

If a Primary Sink does not have an integrated video rendering function, the Primary Sink shall support outputting the video content included in the received MPEG2-TS that contains only a video elementary stream to an externally connected video rendering device.

If a Primary Sink has an integrated audio rendering function, the Primary Sink shall support one of following depending on implementation:

a. render the audio content included in the received MPEG2-TS that contains multiplexed an audio and a video elementary streams.
b. output the audio content included in the received MPEG2-TS that contains multiplexed audio elementary stream and video elementary stream to an externally connected audio rendering device.

If a Primary Sink supports both (a) and (b) above, the Primary Sink may choose one operation depending on local criteria (e.g., detecting attachment or detachment of an external rendering device) or a user may configure the Primary Sink to choose one operation.

If a Primary Sink has an integrated audio rendering function, the Primary Sink may support one of following depending on implementation:

a. render the audio content included in the received MPEG2-TS that contains only an audio elementary stream.
b. output the audio content included in the received MPEG2-TS that contains only an audio elementary stream to an externally connected audio rendering device.

If a Primary Sink supports both (a) and (b) above, the Primary Sink may choose one operation depending on local criteria (e.g., detecting attachment or detachment of an external rendering device) or a user may configure the Primary Sink to choose one operation.

If a Primary Sink does not have an integrated audio rendering function, the Primary Sink shall support outputting the audio content included in the received MPEG2-TS that contains multiplexed single audio elementary stream and single video elementary stream to an externally connected audio rendering device.

If a Primary Sink does not have an integrated audio rendering function, the Primary Sink may support outputting the audio content included in the received MPEG2-TS that contains only an audio elementary stream to an externally connected audio rendering device.

Table 3 summarizes the required capabilities of a Primary Sink.

3.1.2.2 Secondary Sink:

During a WFD Session, a Secondary Sink shall support receiving an MPEG2-TS from one WFD Source.

A Secondary sink shall support receiving an MPEG2-TS that contains only an audio elementary stream.

Table 3 summarizes the required capabilities of a Secondary Sink.

3.1.3 Requirements for WFD Devices under Coupled Sink Operation

A WFD Source may support Coupled Sink Operation.

A Primary Sink may support Coupled Sink Operation.

A Secondary Sink may support Coupled Sink Operation.

Figure 6 illustrates the role of a WFD Source, a Primary Sink and a Secondary Sink under Coupled Sink Operation. In a Coupled Sink Operation, a Primary Sink and a Secondary Sink that support Coupling are Coupled together and in Coupled status (see sections 3.1.2 and 4.9.1).

If a WFD Source supports Coupled Sink Operation and operates with a Primary Sink and a Secondary Sink in Coupled status, and if the Primary Sink does not support rendering audio content, the WFD Source shall transmit an MPEFG2-TS that contains a video elementary stream to the Primary Sink and shall transmit an MPEG2-TS that contains an audio elementary stream to the Secondary Sink.

If a WFD Source supports Coupled Sink Operation and operates with a Primary Sink and a Secondary Sink in Coupled status, and if the Primary Sink supports rendering audio content, the WFD Source shall transmit an MPEG2-TS that contains a video elementary stream to the Primary Sink and shall support dynamically switching the destination of an audio elementary stream between the Primary Sink and the Secondary Sink. This switching of the destination of an audio elementary stream may be triggered by a request from either the Primary Sink or the Secondary Sink, and shall not require tearing down of the already established WFD Session. This procedure is specified in sections 4.9.6, 6.1.11 and 6.4.10.

A Primary Sink that supports Coupled Sink Operation shall follow the Coupled Sink Operation as described in section 4.9.1.

If a Primary Sink is Coupled with a Secondary Sink and if the Primary Sink operates with a WFD Source that supports Coupled Sink Operation, the Primary Sink shall support receiving an MPEG2-TS that contains a video elementary stream from the WFD Source.

A Secondary Sink that supports Coupled Sink Operation shall follow the Coupled Sink Operation as described in section 4.9.1.

If a Secondary Sink is Coupled with a Primary Sink and if the Secondary Sink operates with a WFD Source that supports Coupled Sink Operation, the Secondary Sink shall support receiving an MPEG2-TS that contains an audio elementary stream from the WFD Source.

3.1.4 WFD Session

There are four kinds of WFD Sessions as described below

- Audio only WFD Session where there is only one Primary Sink or one Secondary Sink, as illustrated in Figure 3.
- Video only WFD Session where there is only one Primary Sink, as illustrated in Figure 4.
- Audio/video WFD Session where there is only one Primary Sink, which renders both audio and video, as illustrated in Figure 5.

- Audio/video WFD Session where there is Coupled WFD Sinks, and the Primary Sink renders video while the Secondary Sink renders the corresponding audio. Figure 6 shows audio rendered at the Secondary Sink.

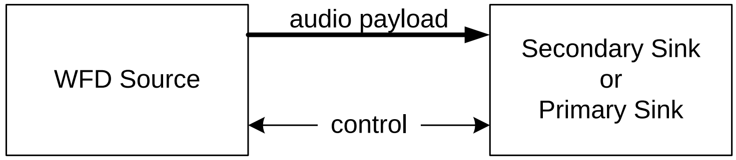


Figure 3. Audio-only WFD Session

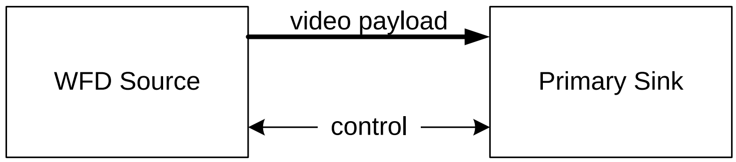


Figure 4. Video-only WFD Session

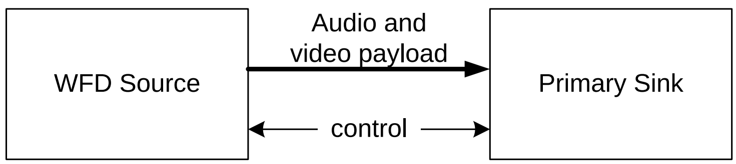


Figure 5. Audio and video WFD Session

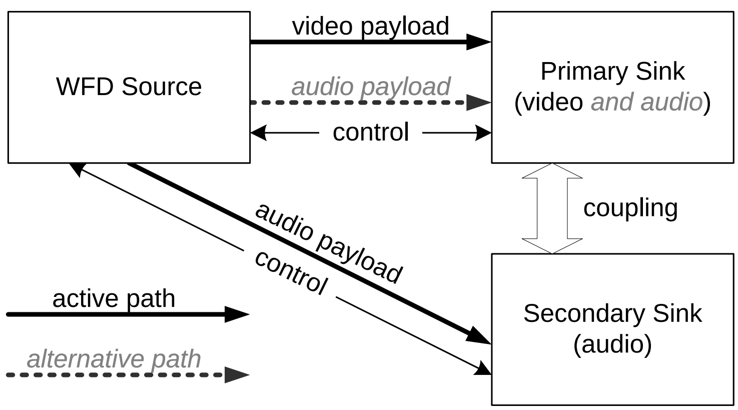


Figure 6. WFD Session under Coupled Sink Operation

3.2 WFD Connection Topology

A WFD Source shall use a WFD Connection with a WFD Sink for all WFD data and control messages. A WFD Sink shall use a WFD Connection with a WFD Source for all WFD data and control messages. The WFD Connection shall be either Wi-Fi P2P or TDLS for an R1 device, or Wi-Fi P2P or Infrastructure for an R2 device.

3.2.1 Wi-Fi P2P

Figure 7 shows a WFD Connection using Wi-Fi P2P (Wi-Fi Direct) [7]. A WFD Device shall support a WFD Connection using Wi-Fi P2P.

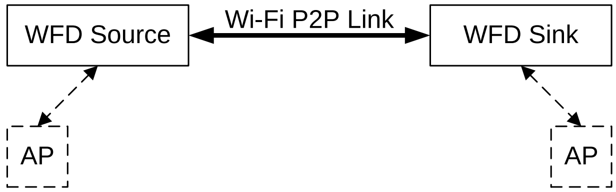


Figure 7. WFD Connection using Wi-Fi P2P

Note: The APs shown in Figure 7 may be the same AP, or different APs, or may not exist.

A WFD Device may support concurrent operation with infrastructure BSS.

If a WFD Session including a single WFD Sink, either a WFD Source or a WFD Sink may be a P2P GO. If a WFD Session includes both a Primary Sink and a Secondary Sink, then the WFD Source shall be a P2P GO.

3.2.2 TDLS

Figure 8 shows a WFD Connection using TDLS. WFD devices may support the WFD Connection using TDLS.

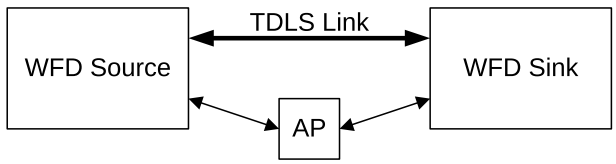


Figure 8. TDLS topology

If TDLS is used for a WFD Connection, then a WFD Source shall support maintaining connection with an AP (or a P2P GO) that the WFD Source is associated with. If TDLS is used for a WFD Connection, a WFD Sink shall support maintaining connection with an AP (or a P2P GO) that the WFD Sink is associated with.

Although Figure 8 has been simplified to show only one WFD Sink, a Secondary Sink in the same BSS may also be part of the same WFD Session; see Figure 6. In this case, the Secondary Sink shall have a different TDLS link with the WFD Source from the TDLS link between the WFD Source and the Primary Sink.

3.2.3 Wi-Fi Infrastructure

Figure 9 shows a WFD connection using Wi-Fi infrastructure between a WFD R2 Source and a WFD R2 Sink. If a WFD R2 Device is capable of connecting to an IP network over Wi-Fi infrastructure, then the WFD R2 Device shall support service discovery and connection over Wi-Fi infrastructure using mDNS/DNS-SD as described in [48] and [49].

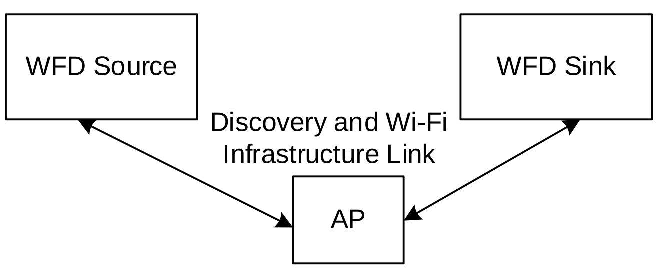


Figure 9. Discovery and WFD connection over Wi-Fi infrastructure

3.3 Functions and services

3.3.1 Basic Wi-Fi functions and services

This specification requires that a WFD Device shall pass the following Wi-Fi Alliance Certifications:

- 802.11n Certification (implicitly requires WPA2 [16] and WMM)

- Wi-Fi Protected Setup™ Certification [8] (Implicit in Wi-Fi Direct certification)
- Wi-Fi Direct Certification [7]

A WFD Device that supports Wi-Fi Display over TDLS shall pass the following Wi-Fi Alliance Certification:

- Wi-Fi TDLS Certification [13]

A WFD R2 Device that supports Wi-Fi P2P over 11ac shall pass the following Wi-Fi Alliance certifications.

- 802.11ac certification

3.3.2 Wi-Fi Display specific functions and services

Table 3 summarizes the functions and services for WFD devices.

The WFD Source column indicates whether the function/service is Mandatory (M) or Optional (O), for a WFD Source.

The Primary Sink and Secondary Sink columns indicate whether the function/service is Mandatory (M) or Optional (O), for a WFD Sink.

A WFD Device that advertises itself as being capable of both WFD Source and Primary Sink functionality during the WFD Device Discovery shall support both sets of functionalities. For such WFD devices, one role (WFD Source vs. Primary Sink) shall be selected prior to a single WFD Session and shall not change during that WFD Session. Such WFD R1 Device may support concurrent operation as a WFD R1 Source in a WFD Session and as a Primary Sink in another WFD Session but such operation is outside the scope of this specification. A WFD R2 dual-role device may support concurrent operation as a Source and Primary Sink in the same WFD Session.

Table 3. Functions and services

<table>
<tr><th rowspan="2">Functions and Services</th><th colspan="2">WFD Source</th><th colspan="2">Primary Sink</th><th colspan="2">Secondary Sink</th></tr>
<tr><th>R1</th><th>R2</th><th>R1</th><th>R2</th><th>R1</th><th>R2</th></tr>
<tr><td>WFD Device Discovery (section 4.3)</td><td colspan="2">M</td><td colspan="2">M</td><td colspan="2">M[3]</td></tr>
<tr><td>Post Association WFD Discovery and WFD Session over Wi-Fi Infrastructure (section 4.8.1)</td><td colspan="2">M</td><td colspan="2">M</td><td colspan="2">M</td></tr>
<tr><td>WFD Capability negotiation (section 4.6)</td><td colspan="2">M</td><td colspan="2">M</td><td colspan="2">M</td></tr>
<tr><td>WFD Coupled Sink Operation (section 4.9)</td><td colspan="2">O</td><td colspan="2">O</td><td colspan="2">O</td></tr>
<tr><td>WFD Connection Setup, with a WFD Sink (section 4.5.2)</td><td colspan="2">M</td><td colspan="2">M</td><td colspan="2">M</td></tr>
<tr><td>WFD Session establishment, with a WFD Sink (section 4.8)</td><td colspan="2">M</td><td colspan="2">M</td><td colspan="2">M</td></tr>
<tr><td>Encode and packetization of the captured Display</td><td colspan="2">M</td><td colspan="2">N/A</td><td colspan="2">N/A</td></tr>
<tr><td>Transport of multiplexed audio and video payload</td><td colspan="2">M</td><td colspan="2">M</td><td colspan="2">N/A</td></tr>
<tr><td>De-multiplex, de-packetization and decode of received audio and video payload</td><td colspan="2">N/A</td><td colspan="2">M[1]</td><td colspan="2">N/A</td></tr>
<tr><td>Rendering of decoded video on local display panel or a display panel that is attached to a WFD Sink (dongle)</td><td colspan="2">N/A</td><td colspan="2">M</td><td colspan="2">N/A</td></tr>
<tr><td>Power Save mechanisms[4]</td><td colspan="2">M</td><td colspan="2">M</td><td colspan="2">M</td></tr>
<tr><td>Session termination (section 4.12)</td><td colspan="2">M</td><td colspan="2">M</td><td colspan="2">M</td></tr>
<tr><td>Encode and packetization of captured audio</td><td colspan="2">M</td><td colspan="2">N/A</td><td colspan="2">N/A</td></tr>
<tr><td>Transport of video payload without audio</td><td colspan="2">O</td><td colspan="2">M</td><td colspan="2">N/A</td></tr>
<tr><td>Transport of audio payload without video</td><td>O</td><td>M</td><td colspan="2">O</td><td colspan="2">M</td></tr>
<tr><td>Multiplex video and audio payload</td><td colspan="2">M[2]</td><td colspan="2">N/A</td><td colspan="2">N/A</td></tr>
<tr><td>De-packetization and decode of received audio payload that is not multiplexed with video payload</td><td colspan="2">N/A</td><td colspan="2">O</td><td colspan="2">M</td></tr>
</table>

Functions and Services	WFD Source		Primary Sink		Secondary Sink	
	R1	R2	R1	R2	R1	R2
Rendering of decoded audio on local speakers or speakers attached to a WFD Sink (dongle)	N/A		M[1]		M	
Link Content Protection (for protected content) (section 4.7) Note: If Link Content Protection is not supported either by the WFD Source or the WFD Sink, protected content shall not be streamed.	O		O		O	
Time Synchronization (section 4.10.1)	O		O		O	
Concurrent WLAN operation (section 4.14)	O		O		N/A	
Persistent WFD group (section 4.13)	O		O		O	
AV Stream Control using RTSP (sections 6.2.5, 6.2.6, 6.2.7)	M		M		M	
AV Audio Stream Routing Control during Coupled Sink Operation (section 4.10.4)	O		O		O	
User Input Back Channel (section 4.11)	O		O		N/A	
Remote I2C Read/Write (section 7)	O		O		N/A	
WFD Standby / resume (section 4.15)	O		O		O	
Direct Streaming (Non-Transcoding Mode) Operation (section 4.16	N/A	O	N/A	O	N/A	N/A
Transport of Auxiliary Content Stream (section 4.16.4)	N/A	O	N/A	O	N/A	N/A
Video Frame Skipping (section 4.10.3.1)	O	O	O	O	N/A	N/A
Data transport over TCP and TCP/UDP switch (section 4.10.6)	N/A	M	N/A	M	N/A	M

Notes:

1. If a Primary Sink does not have an integrated audio rendering function or an audio output port to be connected to an external audio rendering device, decoding or rendering the audio payload is not required to be supported.

2. If a Primary Sink does not have an integrated audio rendering function or an audio output port to be connected to an external audio rendering device, the WFD Source may not transmit multiplexed audio and video payload to the Primary Sink.

3. A Secondary Sink may support initiating the WFD Device Discovery, but shall support following the WFD Device Discovery procedure initiated by a WFD Source or a Primary Sink.

4. Capability to support P2P client in WMM-PS is mandatory when acting as Wi-Fi P2P GO, as specified in [7]. Capabilities to follow Opportunistic Power Save and Notice of Absence from Wi-Fi P2P GO are mandatory when acting as Wi-Fi P2P client, as specified in [7].

3.4 Encoder/decoder characteristics

The audio/video industry uses a wide variety of different methods to encode and decode AV content and there is considerable potential for different AV devices to be incompatible with one another. This specification defines a core subset of these methods to ensure interoperability between all WFD devices at a baseline level (Mandatory), and allows for the inclusion of other methods (Optional) at the discretion of the device manufacturer.

A WFD Sink shall only indicate the audio and video configurations that it supports in an RTSP M3 Response message (described in section 6.4.3).

3.4.1 Audio

An audio capable WFD Device shall support the audio format of 2 channel LPCM audio with16 bits per sample and 48000 samples/second as specified in [5]. Other allowable optional audio formats [5] which are applicable to all WFD devices including WFD R2 Device, are listed in Table 4.

Table 4. Optional audio CODEC formats

Format description
LPCM 44.1 ksps, 16 bits, 2 channels; CD Quality Audio
48ksps, 16bits per sample, 2 channels, MPEG-2 AAC-LC using ADTS
48ksps, 16bits per sample, 4 channels, MPEG-2 AAC-LC using ADTS
48ksps, 16bits per sample, 6 channels, MPEG-2 AAC-LC using ADTS
48ksps, 16bits per sample, 8 channels, MPEG-2 AAC-LC using ADTS
48ksps, 16 bits per sample, 2 channels, AC-3
48ksps, 16 bits per sample, 4 channels, AC-3
48ksps, 16 bits per sample, 6 channels, AC-3

An audio capable WFD R2 Device shall support the mandatory audio format specified in this specification for a WFD R1 Device, and may support the optional audio formats specified for a WFD R1 Device. In addition, a WFD R2 Device may support additional audio formats specified in Table 5. A WFD R2 Source or WFD R2 Sink shall support the formats marked “M” and may need to support the formats marked as “CM” in the mandatory/optional column; support for all other formats marked with “O” are optional.

Table 5. List of supported audio CODEC formats for audio stream from Source to Sink in forward channel

Format type	Sampling rate (ksps) and bits per sample	Supported number of channels	Mandatory/Optional		Remarks
			Source (encoding)	Sink (decoding)	
LPCM	44.1 ksps, 16 bits	2	O	O	As in WFD R1
	48 ksps, 16 bits	2	M	M	As in WFD R1
	48 ksps 16 bits	1	O	O	For WFD R2 only
	48 ksps, 16 bits	6,8	O	O	For WFD R2 only
	48 ksps, 24 bits	2	O	O	For WFD R2 only
	96 ksps, 16 bits	2	O	O	For WFD R2 only
	96 ksps, 24 bits	2,6,8	O	O	For WFD R2 only
	48 and 96 ksps, 16 to 24 bits	up to 32, see CTA-861-G [1]	O	O	For WFD R2 only
AAC-LC	48 ksps, 16 bits	2, 4, 6, 8	O	O	As in WFD R1
AC-3	48 ksps, 16 bits	2, 4, 6	O	O	As in WFD R1
E-AC-3	48 ksps	Multiple	O	O	For WFD R2 only
Dolby TrueHD	48 ksps	Multiple	O	O	For WFD R2 only
AC-4	up to 192 ksps 24 bits	Multiple	O	O	For WFD R2 only
Dolby MAT	48 ksps	up to 7.1	O	O	For WFD R2 only
DTS-HD	up to 192 ksps 24 bits	Multiple	O	O	For WFD R2 only
MPEG-4 AAC	48 ksps	Multiple	O	O	For WFD R2 only
MPEG-H 3D Audio	48 ksps	Multiple	O	O	For WFD R2 only
AAC-ELDv2	48 ksps	up to 7.1	O	O	For WFD R2 only

Format type	Sampling rate (ksps) and bits per sample	Supported number of channels	Mandatory/Optional		Remarks
			Source (encoding)	Sink (decoding)	
Notes: 1. When WFD R2 Source and WFD R2 Sink support CTA Audio, additional LPCM configurations maybe supported (see section 5.4.3).					

3.4.2 Video and Image Codecs

A WFD Device shall use H.264 [1] as the video CODEC.

A video capable WFD Device shall support 640x480 p60 with the Constrained Baseline Profile (CBP) codec of H.264 at level 3.1 as defined in [1].

A WFD Device may use the H.264 Level from 3.1 to 4.2 for the CBP and RHP in this specification.

A video capable WFD Device shall support 640x480 p60 with codec of H.264 CBP at level 3.1. If a WFD Device supports higher resolution(s) of 60Hz family (i.e., at least one of 29.97Hz, 30.00Hz, 59.94Hz and 60.00Hz) than 640x480, it shall also support 720x480 p60 with codec of H.264 CBP at level 3.1. If a WFD Device supports higher resolution(s) of 50Hz family (i.e., at least one of 25.00Hz and 50.00Hz) than 640x480, it shall also support 720x576 p50 with codec of H.264 CBP at level 3.1.

All other combinations of 2D video formats listed in Table 10, H.264 Profile listed in Table 6 and H.264 level (from 3.1 to 4.2) are optional.

Table 6 lists H.264 tools for each H.264 Profile, which are applicable to all WFD devices including WFD R2 devices.

A Primary Sink shall support tools marked “Y” in the CBP column. If a Primary Sink supports RHP, the Primary Sink shall support tools marked “Y” in the RHP column.

A Primary Sink is not required to support tools marked “N” for each Profile.

A WFD Source may use tools marked “Y” for each Profile. A WFD Source shall not use tools marked “N” for each Profile.

The Constrained High Profile (CHP) is based on the standardized High Profile of H.264, but a WFD Source shall not use the B slice tool or the CABAC entropy coding tool when using CHP.

Table 6. Wi-Fi Display H.264 profiles

Tools	CBP	RHP
I and P Slices	Y	Y
B Slices	N	N
SI and SP Slices	N	N
Multiple Reference Frames	N	Y
In-Loop Deblocking Filter	Y	Y
CAVLC Entropy Coding	Y	Y
CABAC Entropy Coding	N	N
Flexible Macroblock Ordering (FMO)	N	N
Arbitrary Slice Ordering (ASO)	N	N
Redundant Slices (RS)	N	N
Data Partitioning	N	N
Interlaced Coding (PicAFF, MBAFF)	N	Y
4:2:0 Chroma Format	Y	Y

Tools	CBP	RHP
Monochrome Video Format (4:0:0)	N	Y
4:2:2 Chroma Format	N	N
4:4:4 Chroma Format	N	N
8 Bit Sample Depth	Y	Y
9 and 10 Bit Sample Depth	N	N
11 to 14 Bit Sample Depth	N	N
8x8 vs. 4x4 Transform Adaptivity	N	Y
Quantization Scaling Matrices	N	Y
Separate Cb and Cr QP control	N	Y
Separate Color Plane Coding	N	N
Predictive Lossless Coding	N	N
Notes: 1. The RHP (formerly known as CHP) is specified by Wi-Fi Alliance in this specification with some restrictions to standard H.264 high profile as listed in Table 7.		

A video capable WFD R2 Device shall support the mandatory video formats specified in this specification for a WFD R1 Device, and may support the optional video formats specified for a WFD R1 Device. In addition, a WFD R2 Device may support additional video and image codecs and formats included in this specification, which are listed in Table 7, Table 8 and Table 9. A WFD R2 Source or R2 Primary Sink shall support the formats marked “M” and may be required to support the formats marked as “CM” in the mandatory/optional column; support for all other formats marked with “O” are optional.

A WFD R2 Device using the H.264 as the video CODEC may use Levels 3.1 to 5.2 in this specification. A WFD R2 Device using the H.265 as the video CODEC may use Levels 3.1 to 5.1 in this specification.

Interlace content may be carried using H.265, but no interlace coding tools are expected to be supported.

Table 7. Video codecs and profiles applicable for transcoding mode

Codec Type	Supported Profiles	Mandatory/Optional				Remarks
		Source (encoding)		Primary Sink (decoding)		
		R1	R2	R1	R2	
H.264	Constrained Baseline (CBP)	M	M	M	M	Applicable tools as listed in Table 6.
	Restricted High (RHP)	O	O	O	O	Applicable tools as listed in Table 6.
	Restricted High 2 (RHP2)	N/A	O	N/A	O	Formerly known as CHP, with CABAC enabled.
	Progressive High Profile	N/A	O	N/A	O	Constraint of the existing High Profile without interlace coding tools (i.e. PicAFF and MBAFF)
H.265	Main Profile	N/A	O	N/A	O	8 bits per sample with 4:2:0 Chroma sampling.
	Main 444 Profile	N/A	O	N/A	O	8 bits per sample with 4:4:4 Chroma sampling.
	Screen Content Coding (SCC)- 8 bit 444 Profile	N/A	O	N/A	O	8 bits per sample with 4:4:4 Chroma sampling, as specified in [45][45]
	Main 444 10 Profile	N/A	O	N/A	O	10 bits per sample with 4:4:4 Chroma sampling.

Table 8. Video codecs and profiles applicable for Non-Transcoding Mode

Codec Type	Supported Profiles	Mandatory/Optional Primary Sink (decoding)		Remarks
		R1	R2	
H.264	Constrained Baseline Profile (BP)	N/A	CM	8 bits per sample with 4:2:0 Chroma sampling
	Main Profile (MP)	N/A	O	8 bits per sample with 4:2:0 Chroma sampling, CABAC enabled
	High Profile (HiP)	N/A	O	8 bits per sample with 4:2:0 Chroma sampling, CABAC enabled
	Progressive High Profile	N/A	O	Constraint of the existing High Profile without interlace coding tools (i.e. PicAFF and MBAFF)
	Progressive High Still Picture Profile	N/A	O	8 bits per sample with 4:2:0 Chroma sampling, single still picture. No interlace coding tools.
H.265	Main Profile	N/A	O	8 bits per sample with 4:2:0 Chroma sampling
	Main 10 Profile	N/A	O	8 to 10 bits per sample with 4:2:0 Chroma sampling
	Main 444	N/A	O	8 bits per sample with 4:4:4 Chroma sampling
	Main Still Picture	N/A	O	8 bits per sample with 4:2:0 Chroma sampling, single still picture encoding
	Screen Content Coding (SCC)- 8 bit 444 Profile	N/A	O	8 bits per sample with 4:4:4 Chroma sampling, as specified in [45]

Table 9. Codecs and profiles applicable for auxiliary streams

Codec Type	Supported Profiles	Mandatory/Optional				Remarks
		Source (encoding)		Primary Sink (decoding)		
		R1	R2	R1	R2	
PNG	N/A	N/A	CM[1]	N/A	CM[1]	
JPEG	N/A	N/A	O	N/A	O	
H.264	Constrained Baseline (CBP)	N/A	CM2	N/A	CM[2]	

Notes:

1. Conditional mandatory to be supported if a WFD R2 Device supports the transport of auxiliary stream as described in section 4.16.

2. Conditional mandatory to be supported if a WFD R2 Device supports the transport of auxiliary stream as described in section 4.16 and HDCP content protection as described in section 4.7. A WFD R2 Source uses H.264 CBP for encoding of auxiliary stream during direct streaming mode if the auxiliary content (e.g., subtitles or captions) associated with the video stream is also subject to Link Content Protection requirements (i.e., protected content). If the auxiliary stream associated with the protected video content is not subject to Link Content Protection requirements, the WFD R2 Source may use any codec type listed in Table 10 for encoding the auxiliary stream that is supported by both the WFD R2 Source and WFD R2 Sink.

Table 10 lists the set of 2D video resolutions and frame rates. For Standard Definition and High Definition modes refer to [11], and for VESA formats refer to [33]. In addition to the display resolutions and frame rates specified for a WFD R1 Device, a video capable WFD R2 Device [33] may support additional display resolutions and frame rates specified in this specification. The Scope column in Table 10 lists whether each resolution and frame rate is applicable to all video capable WFD devices or to a WFD R2 Device only.

Table 10. Supported video resolutions and frame rates

Format		Description	Required minimum H.264 Level	Required minimum H.265 Level	Scope (R1/R2)
Resolution	Frames/Fields per second, (i) interlaced or (p) progressive				
640x480	p60[1]	Standard Definition (Mandatory for all WFD devices that support video Mandatory when using HEVC for all WFD R2 devices that support HEVC video), CEA format.	3.1	3.1	R1, R2
720x480	p60[1]	Standard Definition, CEA format.	3.1	3.1	R1, R2
720x480	i60[1]	Standard Definition, CEA format.	3.1	3.1	R1, R2
720x576	p50	Standard Definition, CEA format.	3.1	3.1	R1, R2
720x576	i50	Standard Definition, CEA format.	3.1	3.1	R1, R2
1280x720	p30[2]	High Definition, CEA format. Mandatory for all video codecs that a WFD R2 Device supports	3.1	3.1	R1, R2
1280x720	p60[1]	High Definition, CEA format.	3.2	4	R1, R2
1920x1080	p30[2]	High Definition, CEA format.	4	4	R1, R2
1920x1080	p60[1]	High Definition, CEA format.	4.2	4.1	R1, R2
1920x1080	i60[1]	High Definition, CEA format.	4	4	R1, R2
1280x720	p25	High Definition, CEA format.	3.1	3.1	R1, R2
1280x720	p50	High Definition, CEA format.	3.2	4	R1, R2
1920x1080	p25	High Definition, CEA format.	3.2	4	R1, R2
1920x1080	p50	High Definition, CEA format.	4.2	4.1	R1, R2
1920x1080	i50	High Definition, CEA format.	3.2	4	R1, R2
1280x720	p24[3]	High Definition, CEA format.	3.1	3.1	R1, R2
1920x1080	p24[3]	High Definition, CEA format.	3.2	4	R1, R2
3840x2160	P24	Ultra High Definition, CEA format.	5.1	5	R2
3840x2160	p25	Ultra High Definition, CEA format.	5.1	5	R2
3840x2160	p30	Ultra High Definition, CEA format.	5.1	5	R2
3840x2160	P50	Ultra High Definition, CEA format.	5.2	5.1	R2
3840x2160	P60	Ultra High Definition, CEA format.	5.2	5.1	R2
4096x2160	P24	Ultra High Definition, CEA format.	5.2	5	R2
4096x2160	p25	Ultra High Definition, CEA format.	5.2	5	R2
4096x2160	P30	Ultra High Definition, CEA format.	5.2	5	R2
4096x2160	P50	Ultra High Definition, CEA format.	5.2	5.1	R2
4096x2160	P60	Ultra High Definition, CEA format.	5.2	5.1	R2
800x600	p30	SVGA; VESA format.	3.1	3.1	R1, R2
800x600	p60	SVGA; VESA format.	3.2	4.0	R1, R2
1024x768	p30	XGA; VESA format.	3.1	3.1	R1, R2

Format		Description	Required minimum H.264 Level	Required minimum H.265 Level	Scope (R1/R2)
Resolution	Frames/Fields per second, (i) interlaced or (p) progressive				
1024x768	p60	XGA; VESA format.	3.2	4	R1, R2
1152x864	p30	XGA+; VESA format.	3.2	4	R1, R2
1152x864	p60	XGA+; VESA format.	4	4.1	R1, R2
1280x768	p30	WXGA; VESA format.	3.2	4	R1, R2
1280x768	p60	WXGA; VESA format.	4	4.1	R1, R2
1280x800	p30	WXGA; VESA format.	3.2	4	R1, R2
1280x800	p60	WXGA; VESA format.	4	4.1	R1, R2
1360x768	p30	WXGA; VESA format.	3.2	4	R1, R2
1360x768	p60	WXGA; VESA format.	4	4.1	R1, R2
1366x768	p30	WXGA; VESA format.	3.2	4	R1, R2
1366x768	p60	WXGA; VESA format.	4.2	4.1	R1, R2
1280x1024	p30	SXGA (17” and 19” LCD); VESA format.	3.2	4	R1, R2
1280x1024	p60	SXGA (17” and 19” LCD); VESA format.	4.2	4.1	R1, R2
1400x1050	p30	SXGA+ (14” and 15” PC); VESA format.	3.2	4	R1, R2
1400x1050	p60	SXGA+ (14” and 15” PC); VESA format.	4.2	4.1	R1, R2
1440x900	p30	WXGA+ (19” LCD); VESA format.	3.2	4	R1, R2
1440x900	p60	WXGA+ (19” LCD); VESA format.	4.2	4.1	R1, R2
1600x900	p30	VESA format.	3.2	4	R1, R2
1600x900	p60	VESA format.	4.2	4.1	R1, R2
1600x1200	p30	UXGA (20” LCD); VESA format.	4	5	R1, R2
1600x1200	p60	UXGA (20” LCD); VESA format.	4.2	5.1	R1, R2
1680x1024	p30	WSXGA (19” LCD); VESA format.	3.2	4	R1, R2
1680x1024	p60	WSXGA (19” LCD); VESA format.	4.2	4.1	R1, R2
1680x1050	p30	SXGA (22” LCD); VESA format.	3.2	4	R1, R2
1680x1050	p60	SXGA (22” LCD); VESA format.	4.2	4.1	R1, R2
1920x1200	p30	WUXGA; VESA format.	4.2	5	R1, R2
1920x1200	p60	WUXGA; VESA format.	5.2	5.0	R2
2560x1440	p30	WQHD; VESA format.	5	5	R2
2560x1440	p60	WQHD; VESA format.	5.2	5.1	R2
2560x1600	p30	WQXGA; VESA format.	5	5	R2
2560x1600	p60	WQXGA; VESA format.	5.2	5.1	R2
800x480	p30	Handheld devices	3.1	3.1	R1, R2
800x480	p60	Handheld devices	3.1	3.1	R1, R2
854x480	p30	Handheld devices	3.1	3.1	R1, R2

Format		Description	Required minimum H.264 Level	Required minimum H.265 Level	Scope (R1/R2)
Resolution	Frames/Fields per second, (i) interlaced or (p) progressive				
854x480	p60	Handheld devices	3.1	3.1	R1, R2
864x480	p30	Handheld devices	3.1	3.1	R1, R2
864x480	p60	Handheld devices	3.1	3.1	R1, R2
640x360	p30	Handheld devices	3.1	3.1	R1, R2
640x360	p60	Handheld devices	3.1	3.1	R1, R2
960x540	p30	Handheld devices	3.1	3.1	R1, R2
960x540	p60	Handheld devices	3.1	3.1	R1, R2
848x480	p30	Handheld devices	3.1	3.1	R1, R2
848x480	p60	Handheld devices	3.1	3.1	R1, R2
Notes: 1. 60fps (frame per sec for progressive and field per sec for interlace) for CEA resolutions here includes 60.000fps and 59.94fps. 2. 30fps for CEA resolutions here includes 30.000Hz and 29.97fps. 3. 24fps for CEA resolutions here includes 24fps and 23.98fps.					

DTS/PTS values in PES Header shall not use rounded numbers but shall use the exact number of each frames (or fields) per second.

3.4.3 Stereoscopic 3D video

Because Stereoscopic 3D video as defined in the Wi-Fi Display Technical Specification v1.0 has been deprecated, this section is removed.

4 WFD Session functional description and procedures

The following sections discuss the discovery, connection setup, capability negotiation, content protection and session establishment. At a high level, a user interface on a WFD Source and/or a WFD Sink presents the discovered WFD devices to the user via a user interface so that the user may select the peer device to be used in a WFD Session. Once device selection is performed by the user, a WFD Connection is established and the transport layer is used to stream AV media from a WFD Source to a peer WFD Sink.

The ability to select a peer is mandatory on a WFD Sink and optional on WFD Source.

Presentation and the method of selection of the discovered WFD devices is outside the scope of this specification.

4.1 Reference Model

Figure 10 shows a reference model for session management of a WFD Source and WFD Sink. This conceptual model includes a set of predefined functions such as WFD Device Discovery, presentation, session control, and transport.

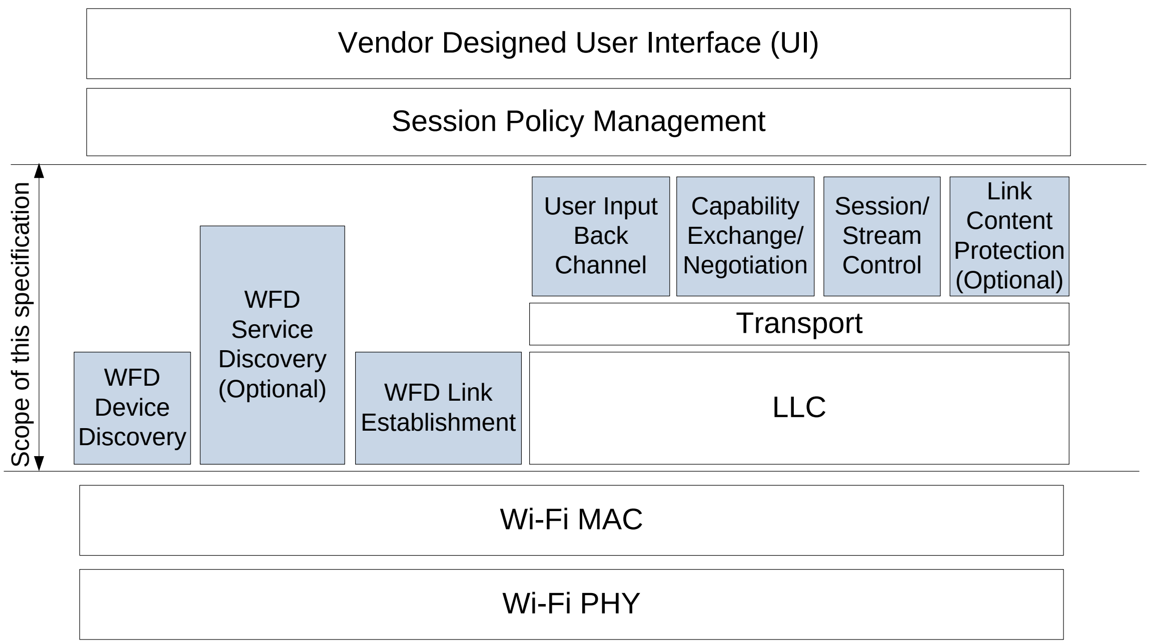


Figure 10. Reference model for session management in WFD devices

Figure 11 shows a reference model for AV payload processing for WFD Source and WFD Sink.

The protocols and procedures for each of the functions illustrated in Figure 10 and Figure 11 are described in subsequent sections.

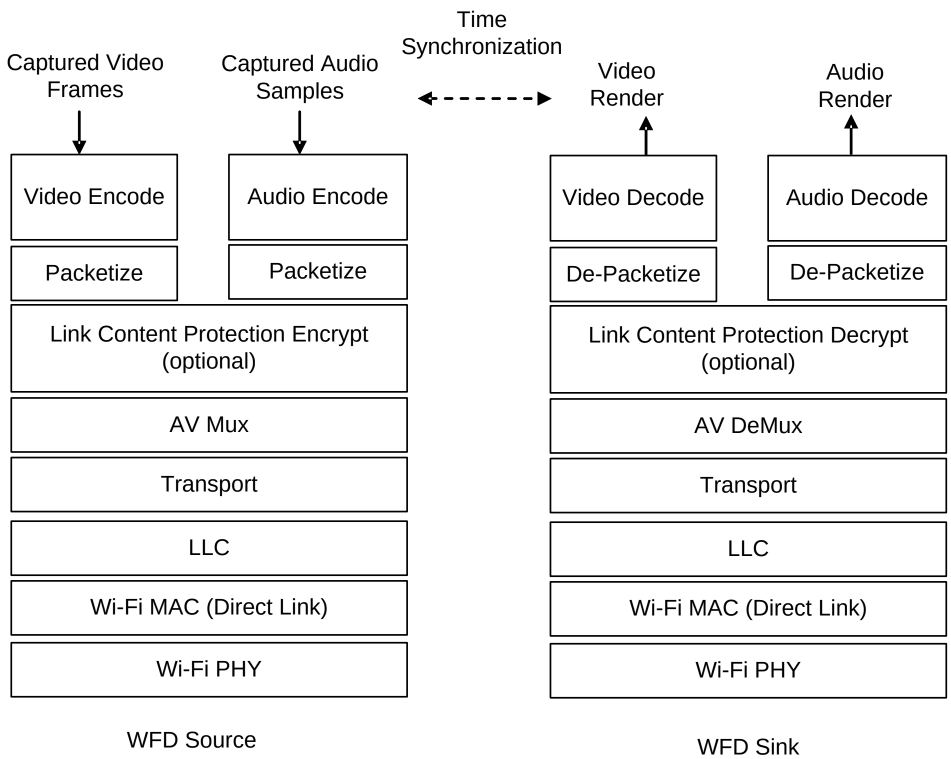


Figure 11. Reference model audio/video payload processing

4.2 WFD connection setup, WFD session establishment and management functions

This section describes the management protocols, procedures and order of operations used to establish and manage a WFD Session.

Figure 14 and Figure 15 provide pictorial representations of the lifetime of a WFD Session. A WFD Session is defined between a WFD Source and a WFD Sink (or WFD Sinks in case of Coupled Sink Operation). In a WFD Session, the WFD Source transmits audio and/or video content to a WFD Sink (or WFD Sinks) and a WFD Sink receives (and renders) the content or outputs the content to an external device.

The general sequence for WFD Connection Setup, WFD Session establishment, and management is as follows:

1. WFD Device Discovery: Initially, a WFD Source and a WFD Sink discover each other’s presence, prior to WFD Connection Setup. See section 4.3 for details on this process. When both the WFD R2 Source and WFD R2 Sink are connected to the Wi-Fi infrastructure network or existing P2P connection, they can discover each other using mDNS/DNS-SD as described in section 4.4.2.
2. WFD Service Discovery: This optional step allows a WFD Source and a WFD Sink to discover each other’s service capabilities prior to the WFD Connection Setup. See section 4.4 for details on this process.
3. Device Selection: This step allows a WFD Source or a WFD Sink to select the peer WFD Device for WFD Connection Setup. During this step, user input and/or local policies may be used for device selection.
4. WFD Connection Setup: This step selects the method (Wi-Fi P2P or TDLS) for the WFD Connection Setup with the selected peer WFD Device and allows establishment of a WPA2-secured single hop link with the selected WFD Device. See sections 4.5.2, 4.5.3 and 4.5.4 for details on this process. A WFD R2 Device may select Wi-Fi infrastructure method for the WFD Connection Setup with the selected peer WFD R2 Device.
5. WFD Capability Negotiation: This step includes a sequence of RTSP message exchanges between the WFD Source and WFD Sink(s) to determine the set of parameters that define the audio/video payload during a WFD Session. See section 4.6 for details on this process.

6. WFD Session Establishment: This step establishes the WFD Session. During this step, the WFD Source selects the format of audio/video payload for a WFD Session within a capability of the WFD Sink and informs the selection to the WFD Sink. See section 4.8 for details on this process.
7. User Input Back Channel Setup: This optional step establishes a communication channel between the WFD Source and the WFD Sink for transmitting control and data information emanating from user input at the WFD Sink. See section 4.11 for details on this process.
8. Link Content Protection Setup: This optional step derives the session keys for Link Content Protection used for transmission of protected content. See section 4.7 for details on this process.
9. Payload Control: Payload transfers are started after the above sequences are completed, and may be controlled during a WFD Session. See sections 4.10.3 and 4.10.4 for details of this process.
10. WFD Source and WFD Sink standby: This optional step enables the WFD Source and WFD Sink to manage and control power modes such as standby and resume (e.g., wake-up) while the WFD Session is maintained. See section 4.15 for details on this process.
11. WFD Session Teardown: This step terminates the WFD Session. See section 4.12 for details on this process.

4.3 WFD R1 Discovery

4.3.1 WFD R1 Service Discovery

Because the Service Discovery using WFD IE as defined in the Wi-Fi Display Technical Specification v1.0 has been deprecated, this section is removed. This feature is not applicable to a WFD R2 Device.

4.3.2 WFD R1 Device Discovery

Wi-Fi Display Device Discovery builds upon the P2P Device Discovery mechanisms defined in [7] enabling a WFD Device to quickly find a peer WFD Device and to determine whether a connection may be established for a subsequent WFD Session.

A WFD Source shall support initiating and following the procedures of WFD Device Discovery.

A Primary Sink shall support initiating and following the procedures of WFD Device Discovery.

A Secondary Sink may support initiating WFD Device Discovery.

A Secondary Sink shall support following procedures of WFD Device Discovery initiated by a WFD Source or a Primary Sink.

A WFD Device shall comply with all procedures as specified for P2P Device Discovery in [7] with the following additions.

- A WFD Device shall include the WFD Information Element (WFD IE) in all Beacon, Probe Request and Probe Response frames. The WFD IE carries basic information such as device-type and device-status as specified in section 5.1.1 so as to facilitate an optimal connection with a peer WFD Device. If a WFD Device is acting as a GO and receives a Probe Request frame containing a WFD IE, then that WFD Device shall respond with a Probe Response frame containing the information of its WFD capable client(s) as specified in section 5.1.10.
- A WFD Device that is associated with an infrastructure AP, and that is operating as a Wi-Fi P2P device, should respond to Probe Requests containing a P2P IE, a WFD IE, and a P2P wildcard SSID. The Probe Response frame shall have the P2P IE and the WFD IE. This Probe Response frame should be transmitted on the channel on which the Probe Request was received.

If a device supports the capability to become either a WFD Source or a Primary Sink, the device may advertise its device-type as a dual role capable device during the WFD Device Discovery. Additionally, such a device may advertise its device-type as a dual-role capable device during a WFD Service Discovery as described in section 4.4. However, the device shall advertise only one of these capabilities (i.e., either a WFD Source or a Primary Sink) during WFD Connection Setup, and WFD Capability Negotiation.

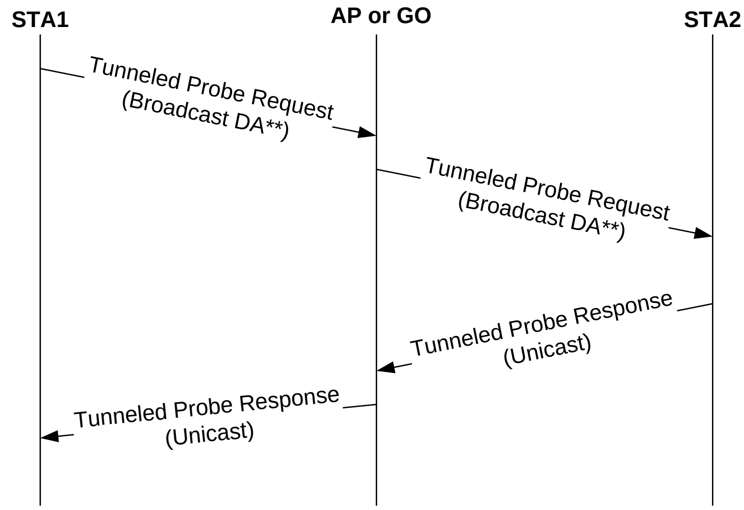
If a WFD Device supports TDLS for a WFD Session, the WFD Device may transmit a tunneled Probe Request frame (Figure 12) containing a WFD IE via an AP or a GO to a broadcast destination address (DA).

A WFD Device may transmit a tunneled Probe Request frame via an AP or a GO to a unicast DA (for example, if the MAC address of the target WFD Device is already known) as in the case of previously paired WFD devices.

If a WFD Device supports TDLS as the connection mechanism for a WFD Session and if the WFD Device receives a tunneled Probe Request frame with the WFD IE, the WFD Device shall respond by transmitting a tunneled Probe Response frame with the WFD IE (via the AP or the GO) to the STA that transmitted the tunneled Probe Request.

A WFD Device which does not support TDLS may respond to a tunneled Probe Request with a tunneled Probe Response frame.

If a Wi-Fi P2P device does not support intra-BSS distribution, then that device shall not attempt to setup a TDLS link and it shall not transmit tunneled Probe Requests or Responses.



** Tunneled Probe Requests may also be sent to Unicast DA, e.g., for subsequent re-connections

Figure 12. Tunneled Probing when TDLS is used

4.4 WFD R2 Discovery

The WFD R2 Device discovery procedure over P2P is based on Probe Request/Response exchange and is similar to the WFD R1 Device discovery procedure.

The WFD R2 discovery procedure over a BSS infrastructure connection is executed using mDNS/DNS-SD as described in [48] and [49] with extensions/definitions specified in this section and following subsections.

In order to execute WFD R2 discovery over a BSS infrastructure connection, the WFD R2 Device shall be capable of acting as both an mDNS/DNS-SD browser and an mDNS/DNS-SD responder simultaneously

The following rules specify the use of mDNS/DNS-SD during WFD R2 discovery over a BSS infrastructure connection for a WFD R2 Device:

1. A WFD R2 Device shall activate as an mDNS/DNS-SD responder by default (Out-Of Box setting).
2. If a WFD R2 Device acting as an mDNS/DNS-SD responder for a BSS interface receives an mDNS query over a BSS infrastructure connection and the received mDNS query includes a Service Type that the WFD R2 Device supports, the WFD R2 Device should activate an mDNS/DNS-SD browser for a BSS interface to send an mDNS query (refer to section 4.8.1).

4.4.1 WFD R2 Service Discovery

The service discovery over infrastructure with IP connectivity is performed using multicast DNS (mDNS) [48] and DNS Service Discovery (DNS-SD) [48]. The details are described in section 4.8.1.

Service Names used in this specification are complaint with the service naming conventions described in [49]. Two Service Names are defined in this specification:

WFD R2 Source: displaysrc

WFD R2 Primary Sink: display

These Service Names are IANA registered service name as described in RFC6335 [59]. The Service Type is structured according to the following convention as defined in [50]:

Service Type = <_><*Service Name*><_><*Transport Protocol*>

The <Transport Protocol> for WFD R2 devices is labeled as “tcp”. The Service Types for the corresponding WFD R2 Source and WFD R2 Primary Sink are:

WFD R2 Source: displaysrc._tcp

WFD R2 Primary Sink: display._tcp

The Instance Name of a WFD R2 service is used to provide additional descriptions of the instance of the service. For more information about the rules in naming an Instance Name, see [50]. An example of an Instance Name is “John Living Room TV”.

4.4.2 WFD R2 Device Discovery

This section describes the discovery mechanisms used by a WFD R2 Device. A WFD R2 Device shall be capable of discovering WFD R2 devices and WFD R1 devices.

4.4.2.1 Discovering a WFD R1 Device

A WFD R2 Device is able to discover a WFD R1 Device using the existing mechanism defined in section 4.3. This mechanism is based on P2P Probe Request and P2P Probe Response frames with WFD IE and no Service Hash(s). Refer to section 4.3 for details.

4.4.2.2 Discovering a WFD R2 Device

The Wi-Fi Display R2 Device Discovery over P2P builds upon the P2P Device Discovery mechanisms defined in [7], enabling a WFD R2 Device to find a peer WFD R2 Device and to determine whether a connection may be established for a subsequent WFD Session.

A WFD R2 Source shall support initiating and following the procedures of WFD R2 Device Discovery.

An R2 Primary Sink shall support initiating and following the procedures of WFD R2 Device Discovery.

A WFD R2 Device shall comply with all procedures as specified for P2P Device Discovery in [7] with the following additions:

- A WFD R2 Device shall include the WFD Information Element (WFD IE) that contains both of the WFD Device Information subelement and the WFD R2 Device Information subelement in all Beacon and P2P Probe Request frames. The WFD IE carries basic information such as device-type and device-status as specified in section 5.1.1 so as to facilitate an optimal connection with a peer WFD Device.
- If a WFD R2 Device receives a P2P Probe Request including the WFD IE that contains both of the WFD Device Information subelement and the WFD R2 Device Information subelement, the WFD R2 Device shall include the WFD IE that contains both of the WFD Device Information subelement and the WFD R2 Device Information subelement in the P2P Probe Response frame sent as a response. The WFD IE carries basic information such as device-type and device-status as specified in section 5.1.1 so as to facilitate an optimal connection with a peer WFD R2 Device.
- If a WFD R2 Device receives a P2P Probe Request including the WFD IE that contains the WFD Device Information subelement but does not contain the WFD R2 Device Information subelement, the WFD R2 Device shall include the WFD Device Information subelement and may include the WFD R2 Device Information subelement in the WFD IE in the P2P Probe Response frame sent as a response. The WFD IE

carries basic information such as device-type and device-status as specified in section 5.1.1 so as to facilitate an optimal connection with a peer WFD R1 Device.

- If a WFD R2 Device is acting as a P2P GO and receives a Probe Request frame containing a WFD IE, then that WFD R2 Device shall respond with a Probe Response frame containing the information of its WFD capable P2P Client(s) as specified in section 5.1.10.
- A WFD R2 Device associated with an infrastructure AP on a channel on which a Probe Request including a WFD IE is received, then the WFD R2 Device shall respond provided it is not already a member of a P2P Group and it does not conflict with the regulatory restriction on that operating channel.
- A WFD R2 Device that is acting as a P2P Client shall be capable of sending a Probe Request on an operating channel.
- If a WFD R2 Device that is acting as a P2P Client receives a Probe Request containing the WFD IE on the operating channel of the P2P group that the P2P Client is connected, the WFD R2 Device shall respond with a Probe Response on the operating channel as long as it does not conflict with the regulatory restriction on that operating channel.
- A WFD R2 Device acting as a P2P GO shall be capable of sending a Probe Request containing the WFD IE on its operating channel

If a WFD R2 Device supports the capability to become either a WFD R2 Source or an R2 Primary Sink, the device may advertise its device-type as a dual role capable device during the WFD R2 Device Discovery. However, the device shall advertise only one of these capabilities (i.e., either a WFD R2 Source or an R2 Primary Sink) during WFD Connection Setup and WFD Capability Negotiation.

4.5 WFD Connection Setup

The primary purpose of the WFD Device Discovery and (optional) Service Discovery procedures are to facilitate the decision of which WFD devices are to be paired and setup a WFD Session. However, the decision process (i.e., whether to establish a paring between a WFD Source and a WFD Sink) is implementation specific and outside the scope of this specification.

WFD devices shall support WFD Connection Setup over P2P and may support WFD Connection Setup over TDLS as specified herein. WFD R2 devices may support WFD connection setup over infrastructure as specified in section 4.8.1.

There are two underlying connectivity schemes for establishing a WFD Connection for WFD R1 devices, i.e., Wi-Fi P2P and TDLS as specified in section 3.2. The connectivity scheme to be employed is determined based on the Connectivity Scheme Resolution as detailed in section 4.5.1.1.

There are two underlying connectivity schemes for establishing a WFD Connection for WFD R2 devices: Wi-Fi P2P and Wi-Fi infrastructure, as specified in section 3.2. The connectivity scheme to be employed is determined based on the Connectivity Scheme Resolution as detailed in section 4.5.1.2.

After a successful WFD Connection Setup between WFD devices (using either Wi-Fi P2P or TDLS as specified in sections 4.5.2 and 4.5.3 respectively), a WFD Device shall perform WFD Capability Negotiation as specified in section 4.6. Upon successful completion of the WFD Capability Negotiation phase, the two WFD devices are said to be paired.

If a discovered WFD Device sets WFD Session Availability bits (B5B4) in the WFD Device Information field of the WFD Device Information subelement to 0b00 (i.e., not available), other WFD devices shall not attempt WFD Connection establishment with that WFD Device until that WFD Device indicates its availability by setting same bits to 0b01 (i.e., available).

4.5.1 Connectivity Scheme Resolution

4.5.1.1 Connectivity Scheme Resolution for WFD R1 Device

This section is only applicable to WFD R1 devices.

WFD devices determine the connectivity scheme to be used for a WFD Session based on the mutual resolution of the Preferred Connectivity (PC) bit and the information in an Associated BSSID subelement carried by the WFD IE.

A WFD Device which supports TDLS and prefers its use for Wi-Fi Display shall set the PC bit in the WFD Device Information field of WFD Device Information subelement within the WFD IE to one, only if either one of the following two compound conditions is fulfilled:

1. If a WFD Device is associated with an AP and
   if the AP has set its TDLS Prohibited bit to zero in the Capabilities field of the Extended Capabilities element, as specified in [13], contained in the Beacon.
2. If a WFD Device is associated with a GO and
   if the GO has set its TDLS Prohibited bit to zero in the Capabilities field of the Extended Capabilities element, as specified in [13], contained in the Beacon, and
   if the GO has set its intra-BSS distribution bit to one in the Device Capability Bitmap of the P2P Capability attribute, as specified in [7], contained in the Beacon.

When both a WFD Source and a WFD Sink attempt to form a WFD Session and are associated with the same AP (as determined from the Associated BSSID subelement within the WFD IE), the connectivity scheme to be used for the WFD Session shall be resolved as per the rules defined in Table 11 below.

Table 11. Connectivity scheme resolution

PC (WFD Source)	PC (WFD Sink)	Associated with same BSSID	Resolved connectivity scheme
0	0	Do not care	Wi-Fi P2P
0	1	Do not care	Wi-Fi P2P
1	0	Do not care	Wi-Fi P2P
1	1	No	Wi-Fi P2P
1	1	Yes	TDLS

A WFD Device that supports TDLS shall include the MAC address of the interface which it intends to use for the subsequent connection as an Alternate MAC Address subelement within the WFD IE in the (tunneled) Probe Response frame if the interface is different from the one on which the (tunneled) Probe Request was received.

The WFD Device shall use the rules defined in Table 11 to deduce the connectivity scheme to be used. If the resolved connectivity scheme implies use of an interface different from the one used during device discovery, then the MAC address corresponding to the inferred interface shall be included in the Alternate MAC Address subelement within the WFD IE in the (tunneled) Probe Response frame.

The WFD Device performing discovery may then use the information indicated in the Alternate MAC Address subelement within the WFD IE received in the (tunneled) Probe Response either to setup a TDLS link with the WFD Device or as a Device ID attribute in the P2P IE in the Probe Request to retrieve the P2P attributes of the WFD Device required for P2P link establishment.

4.5.1.2 Connectivity Scheme Resolution for WFD R2 Device

If a WFD R2 Device wants to establish a WFD Session over an existing infrastructure connection, the WFD R2 Device should send an mDNS query frame over the existing infrastructure connection first. If a WFD R2 Device wants to establish the WFD Session over a new P2P connection, the WFD R2 Device should first send a Probe Request frame over a P2P interface.

4.5.2 Establishing a WFD Connection using Wi-Fi P2P

A WFD Device which intends to establish a WFD Session using Wi-Fi P2P as the underlying connectivity scheme shall form a P2P group as per the methods specified in [7].

After the establishment of a P2P group, WFD devices perform WFD Capability Negotiation as specified in section 4.6. Subsequently, the IP address of the P2P client is assigned by the GO, as specified in section 3.2.6 in [7].

A WFD Device which intends to be discovered shall set the P2P Discoverability bit in the Device Capability Bitmap of the P2P Capability attribute to 0b1.

4.5.2.1 Usage of a WFD IE to establish P2P connection

To establish a P2P connection for a WFD Session, a WFD IE with the WFD Device Information subelement (as specified in section 5.1.2) shall be included in the following frames transmitted by a WFD Device:

- Beacon frames (see section 5.2.1),
- Probe Request frames (see section 5.2.2),
- Probe Response frame (see section 5.2.3),
- Association/Reassociation Request frames (see section 5.2.4),
- Association/Reassociation Response frames (see section 5.2.5),
- GO Negotiation Request frames (see section 5.2.6.1),
- GO Negotiation Response frames (see section 5.2.6.2),
- GO Negotiation Confirmation frames (see section 5.2.6.3),
- P2P Invitation Request frames (see section 5.2.6.4),
- P2P Invitation Response frames (see section 5.2.6.5),
- Provision Discovery Request frames (see section 5.2.6.6), and
- Provision Discovery Response frames (see section 5.2.6.7).

In addition, to establish a P2P connection for a WFD Session with a peer WFD R2 Device, the WFD R2 Device shall include the WFD R2 Device Information subelement (as specified in section 5.1.12) in the WFD IE in the frames mentioned above.

Note: When a WFD R2 Device responds with a Probe Response to a received Probe Request transmitted by the WFD R1 Device, i.e., the received Probe Request does not contain the WFD R2 Device Information subelement in the WFD IE, the WFD R2 Device is not required to include WFD R2 Device Information subelement in the WFD IE in the transmitting Probe Response. Also refer to section 4.4.2.

The WFD IE may optionally include an Associated BSSID subelement (as specified in section 5.1.3).

The content of the WFD Device Information subelement should be immutable during the period of P2P connection establishment, with the following exceptions:

- WFD Device Type bits and WFD R2 Device Type bits
  - If the WFD Device supports dual role capability (i.e., both a WFD Source and a Primary Sink), it can indicate this capability in the WFD Device Type bits (B1B0) of the WFD Device Information field of WFD Device Information subelement and/or the WFD R2 Device Type bits (B1B0) of the WFD R2 Device Information field of the WFD R2 Device Information subelement within the WFD IE in Probe Request frames and/or Probe Response frames during the P2P Scan phase and/or Find phase and in Beacon frames by setting to 0b11
  - In any other cases, these bits shall not be set to 0b11. The role of a WFD (R2) Device shall be fixed as either a WFD (R2) Source or a (R2) Primary Sink while establishing a WFD Connection.
- WFD Session Availability bits
  - Depending on the availability of the WFD Device for a WFD Session, the value for WFD Session Availability bits (B5B4) of the WFD Device Information field of WFD Device Information subelement within the WFD IE may change.

If the type of WFD Device indicated in the WFD Device Type bits (B1B0) in the WFD Device Information field in the WFD Device Information subelement and/or the WFD R2 Device Type bits (B1B0) of the WFD R2 Device Information field of the WFD R2 Device Information subelement within the WFD IE in the received GO Negotiation Request frame, Association/Reassociation Request frame, or P2P Invitation Request frame addressed to the recipient is different from the expected value (e.g., indicating that the sender is a WFD Source and the recipient of the request is also a WFD Source), the recipient should indicate a status code of 2 (fail, incompatible parameter) in the Status attribute within the P2P IE when transmitting the corresponding GO Negotiation Response frame, Association/Reassociation Response frame, or P2P Invitation Response frame respectively.

After the establishment of a WFD Session, the WFD Device that is acting as a P2P GO shall set the WFD Device Type bits (B1B0) of the WFD Device Information field of the WFD Session Information subelement within the WFD IE in all Probe Response frames and P2P Invitation Request/ Response frames, to indicate the type of WFD Device associated.

If the WFD Device is a P2P client of this GO, and indicates its WFD Device Type bits (B1B0) as 0b11 in the WFD Device Information field of the WFD Device Information subelement within the WFD IE in preceding Probe Request/Response frames, the WFD Device that is acting as a P2P GO shall set the WFD Device Type bits (B1B0) of the WFD Device Information field of the WFD Session Information subelement to 0b11 in the WFD IE in all Probe Response frames and P2P Invitation Request/Response frames.

4.5.3 Establish a WFD Connection using TDLS

This section is only applicable to WFD R1 devices.

When two WFD devices that are associated with a common infrastructure AP or a common GO intend to establish a WFD Session and have determined that TDLS is the preferred connectivity scheme (as per section 4.5.1), each WFD Device shall support acting as either a TDLS initiator STA or a TDLS responder STA to setup the TDLS connection through the AP or the GO as specified in [13].

The TDLS link shall be protected using WPA2.

If an associated infrastructure AP uses WEP or WPA for the link between the AP and the WFD Device, the WFD Device shall not transmit the TDLS Setup Request frame for WFD. If an associated infrastructure AP uses WEP or WPA for the link between the AP and the WFD Device, the WFD Device shall not accept the TDLS Setup Request frame for a WFD Connection. If a WFD Device receives such a request, the WFD Device shall reject it by transmitting a TDLS Setup Response frame with status code as 5 (“Security disabled”).

Subsequent to the establishment of a TDLS connection, WFD devices shall perform WFD Capability Negotiation as specified in section 4.6.

The IP address of a WFD Device to be used in TDLS topology is assigned by a DHCP server or similar entity in the network. The negotiated IP address is used for the Layer-3 (L3) connection over the TDLS link and is conveyed to the peer STA using the Local IP address subelement within the WFD IE as described in the process below.

A WFD Device initiating TDLS Setup with a peer WFD Device shall include a WFD IE with its Local IP Address subelement in its TDLS Setup Request frame. A WFD Device responding to this request shall include a WFD IE with its Local IP Address subelement in its TDLS Setup Response frame.

When transmitting a TDLS Setup Request frame or a TDLS Setup Response frame, a WFD Device shall not set the WFD Device Type bits (B1B0) of the WFD Device Information field of WFD Device Information subelement in the WFD IE to 0b11. The role of a WFD Device shall be fixed as either a WFD Source or a Primary Sink while establishing a WFD Connection when using TDLS.

If the type of WFD Device indicated in the WFD Device Type bits (B1B0) in the WFD Device Information field within the WFD IE in the received TDLS Setup Request frame is different from the expected value (e.g., indicating that the sender is a WFD Source and the recipient of the request is also a WFD Source) the recipient should respond using a TDLS Setup Response frame with status code of 38 (The request has not been successful as one or more parameters have invalid values).

4.5.4 Establishing a TCP connection

Upon successful WFD Connection Setup between WFD devices attempting to establish a WFD Session (using Wi-Fi P2P described in section 4.5.2 or TDLS described in section 4.5.3), the connected WFD devices attempt to establish a TCP connection.

TCP connection establishment is specified in IETF STD7 [26]. The TCP connection shall be initiated by the WFD Sink. The WFD Source plays the TCP server role and the WFD Sink plays the TCP client role. A Control Port (default is 7236 [30]) is used to establish and manage sessions between the WFD Source and WFD Sink. Note that the WFD Source can choose any value other than default 7236, and it should be within 49152 to 65535 (as the Private or Ephemeral Ports as described in [29]). The protocol running on the Control Port is RTSP (RFC 2326) [20] and is described in section 6.

Once the TCP connection is established, the RTSP protocol stack shall be active on the WFD Device until the RTSP session is torn down. During the lifetime of the RTSP session, an RTP media session is also active.

If an R2 Source intends to establish a WFD Session over infrastructure with one of the discovered R2 Sink(s), the R2 Source shall first establish a TCP connection with the R2 Sink by sending a TCP SYN packet to the R2 Sink's port number as indicated in the SRV record in an mDNS Response from the R2 Sink. After successful establishing the first TCP connection, the R2 Source shall send a TCP Connection Request Command using a WFD R2 Command structure defined in section 5.3 to the Sink by setting the Type field to 0 and indicating its Control port for RTSP in the Value field. When the R2 Sink receives the WFD R2 Command with Type field equal to 0, the R2 Sink shall initiate a second TCP connection by sending a TCP SYN packet to the port number as identified in the WFD R2 Command.

Note: From this point, establishing a TCP connection for RTSP is identical to the procedure described in the previous paragraph.

The R2 Sink may tear down the first TCP connection, i.e., for a WFD R2 Command, after successfully establishing the second TCP connection. The R2 Sink shall continue to listen to this port unless the R2 Sink stops advertising this port in the SRV record.

4.6 WFD Capability Negotiation

After a successful WFD Connection Setup (and establishment of a TCP connection), the WFD Capability Negotiation phase shall commence as specified herein. It takes place prior to the WFD Session establishment.

A WFD Device shall support the WFD Capability Negotiation process as the following sequence of messages exchanged between the WFD Source and WFD Sink(s) using the RTSP protocol described in chapter 6. Figure 13 provides an illustrated example of the message sequence for WFD Capability Negotiation.

- RTSP M1 Messages (see section 6.4.1):

  The WFD Source sends an RTSP OPTIONS Request message in order to determine the set of RTSP methods (see section 6.1.29.4) supported by the WFD Sink. On receipt of an RTSP M1 (RTSP OPTIONS) Request message from the WFD Source, the WFD Sink responds with an RTSP M1 (RTSP OPTIONS) Response message that lists the RTSP methods supported by the WFD Sink.

- RTSP M2 Messages (see section 6.4.2):

  After a successful RTSP M1 message exchange, the WFD Sink sends an RTSP OPTIONS Request message in order to determine the set of RTSP methods (see section 6.1.29) supported by the WFD Source. On receipt of an RTSP M2 (RTSP OPTIONS) Request message from the WFD Sink, the WFD Source responds with an RTSP M2 (RTSP OPTIONS) Response message that lists the RTSP methods supported by the WFD Source.

- RTSP M3 Messages (see section 6.4.3):
  - RTSP M3 Request:

    After a successful RTSP M2 exchange, the WFD Source sends an RTSP GET_PARAMETER Request message (RTSP M3 Request), explicitly specifying the list of WFD capabilities (see section 6.1) that are of interest to the WFD Source.

  - RTSP M3 Response:

    The WFD Sink responds with an RTSP GET_PARAMETER Response message (RTSP M3 Response).

- RTSP M4 Messages (see section 6.4.4):
  - RTSP M4 Request:

    Based on the RTSP M3 Response, the WFD Source determines the optimal set of parameters to be used for the WFD Session and sends an RTSP SET_PARAMETER Request message containing the parameter set to be used in the WFD Session between the WFD Source and WFD Sink. Note that the WFD Source chooses one appropriate audio and/or video format and other feature(s), from common capabilities supported by the WFD Source and the WFD Sink, possibly also depending on channel conditions, available throughput, original format of the content, available processing power of the WFD Source, and so on.

- RTSP M4 Response:

  On receipt of the RTSP M4 Request from the WFD Source, the WFD Sink responds with an RTSP M4 Response.

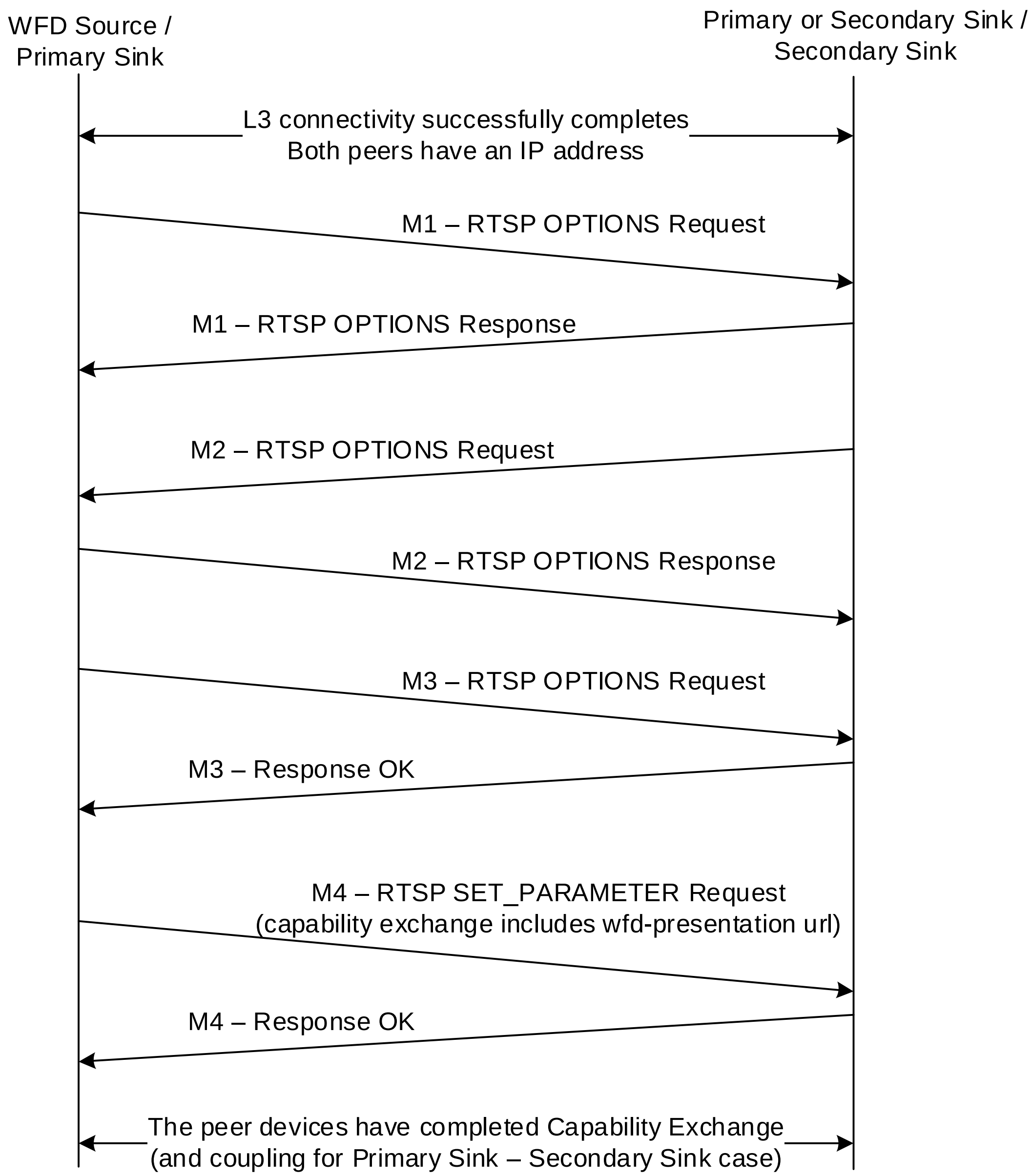


Figure 13. WFD capability negotiation (or coupling) flow using RTSP

Note: WFD Capability Negotiation is performed between a WFD Source and a WFD Sink to determine a common set of capabilities. If the WFD Session involves a Primary and Secondary Sinks as a Coupled Sink Operation, the WFD Source successfully completes the WFD Capability Negotiation procedure with both the Primary and the Secondary Sinks.

4.7 Link content protection setup

In this section, "HDCP (system) 2.x", refers to an exact notation of that version. The corresponding external reference document is identified in Appendix F.

For example, if a WFD Device implements and uses the HDCP system 2.14, the terms “HDCP system 2.x” and “HDCP 2.x” below in this section are interpreted as “HDCP system 2.1” and “HDCP 2.1” respectively”, and the correct reference is [27].

A WFD Source intending to include content subject to Link Content Protection requirements (i.e., protected content) in its RTP payload stream shall establish the HDCP 2.x session key with the WFD Sink, as described in the references identified in Appendix F.

If the WFD Source and WFD Sink support the HDCP system 2.x, both the WFD Source and the WFD Sink shall complete the HDCP 2.x session key establishment before starting any RTP session used for WFD streaming. Refer to the references identified in Appendix F for details.

If the HDCP 2.x session key establishment fails, the WFD Sink supporting the HDCP system 2.x may send an RTSP M7 Request message to the WFD Source supporting the HDCP system 2.x to start an RTP streaming. In this case, the WFD Source can only transmit audio and/or video content that is not required to be protected by the HDCP system 2.x to the WFD Sink.

The following applies only when protected content is transmitted from a WFD Source to WFD Sink(s):

> AKE, locality check, and SKE messages described in the references identified in Appendix F shall be transported over a TCP connection that is different from a TCP connection for RTSP messaging. The WFD Sink shall advertise its local TCP port ID for exchanging the HDCP 2.x messages, using the wfd-content-protection parameter (see section 6.1.5). The WFD Sink shall act as a TCP server for this connection.

If the WFD Session includes a Secondary Sink during Coupled Sink Operation and the audio content also requires content protection, a separate HDCP 2.x session key establishment shall be completed between the WFD Source and the Secondary Sink before starting any RTP session used for WFD streaming.

For the locality check, the recommendation on how to use the TCP/IP layer is described in Appendix B.4.

When a WFD Source transmits protected content to WFD Sink(s) the corresponding PES payload for video shall be encrypted as described in the references identified in Appendix F. The WFD Source may also encrypt the PES payload for audio using the HDCP system 2.x, depending on its local policy and the requirements of the content.

When a WFD Source transmits protected content to WFD Sink(s) during direct streaming as described in section 4.16 and the auxiliary content (e.g., subtitles or captions) associated to the video is subject to Link Content Protection requirements, the WFD R2 Source shall also encrypt the corresponding PES payload for the auxiliary stream.

A WFD Sink which supports receiving protected content shall conform to the HDCP system 2.x specifications as defined in the references identified in Appendix F.

A Primary Sink which supports receiving protected content using the HDCP system 2.x shall support decrypting video content (accompanying audio content does not exist or is not encrypted). In addition, a Primary Sink which supports receiving protected content using the HDCP system 2.x shall support decrypting both audio and video content. A Primary Sink that supports receiving protected content using the HDCP system 2.x and supports direct streaming (section 4.16) shall support decrypting audio, video and auxiliary content.

A WFD Sink, independent of support for reception of protected content, shall support handling audio content as “never copy”, unless the incoming PES payloads for video are not encrypted by the HDCP system 2.x. This means that the audio content in an audio only streaming session shall always be handled as “never copy”.

When both the WFD Source and the WFD Sink support the HDCP system 2.x and have successfully completed the HDCP 2.x session key establishment at least once for establishing (or an already established) WFD Session, following rules are applied:

1. For key renewal, the WFD Source and the WFD Sink shall keep the TCP connection for the HDCP 2.x session key establishment open, during the WFD Session.
2. If the WFD Source detects that the TCP connection for the HDCP 2.x session key establishment has been closed, the WFD Source shall immediately attempt to establish a TCP connection to the same port previously used for the HDCP 2.x session key establishment with the WFD Sink and then restart the HDCP 2.x session key establishment for session key renewal (as specified in the references identified in Appendix F). Under these conditions, the WFD Source shall not transmit any PES packets which require the HDCP encryption before the successful renewal of the HDCP 2.x session key. Additionally, if the TCP connection for the HDCP 2.x session key establishment cannot be re-opened, the WFD Source shall send an RTSP M5 Request message containing the trigger parameter TEARDOWN to terminate the RTSP procedures.

3. If the WFD Sink detects that the TCP connection for the HDCP 2.x session key establishment has been closed, the WFD Sink shall send an RTSP M8 Request message to terminate the RTSP procedures.

Note: Whether PES payload for video is encrypted or not can be inspected by the existence or absence of the HDCP registration descriptor in the PMT and/or the value for PES_extension_flag in PES header for video.

4.8 WFD session establishment

Upon successful completion of the WFD Capability Negotiation phase, the next step is WFD Session establishment and streaming audio and/or video content from the WFD Source to the WFD Sink. When an exchange of RTSP M7 Request and Response messages has successfully completed between the WFD Source and the WFD Sink, the WFD Session is established.

A WFD Device shall support WFD Session establishment, as specified herein. Figure 14 depicts the WFD Session establishment procedure, followed by WFD Session management (including termination).

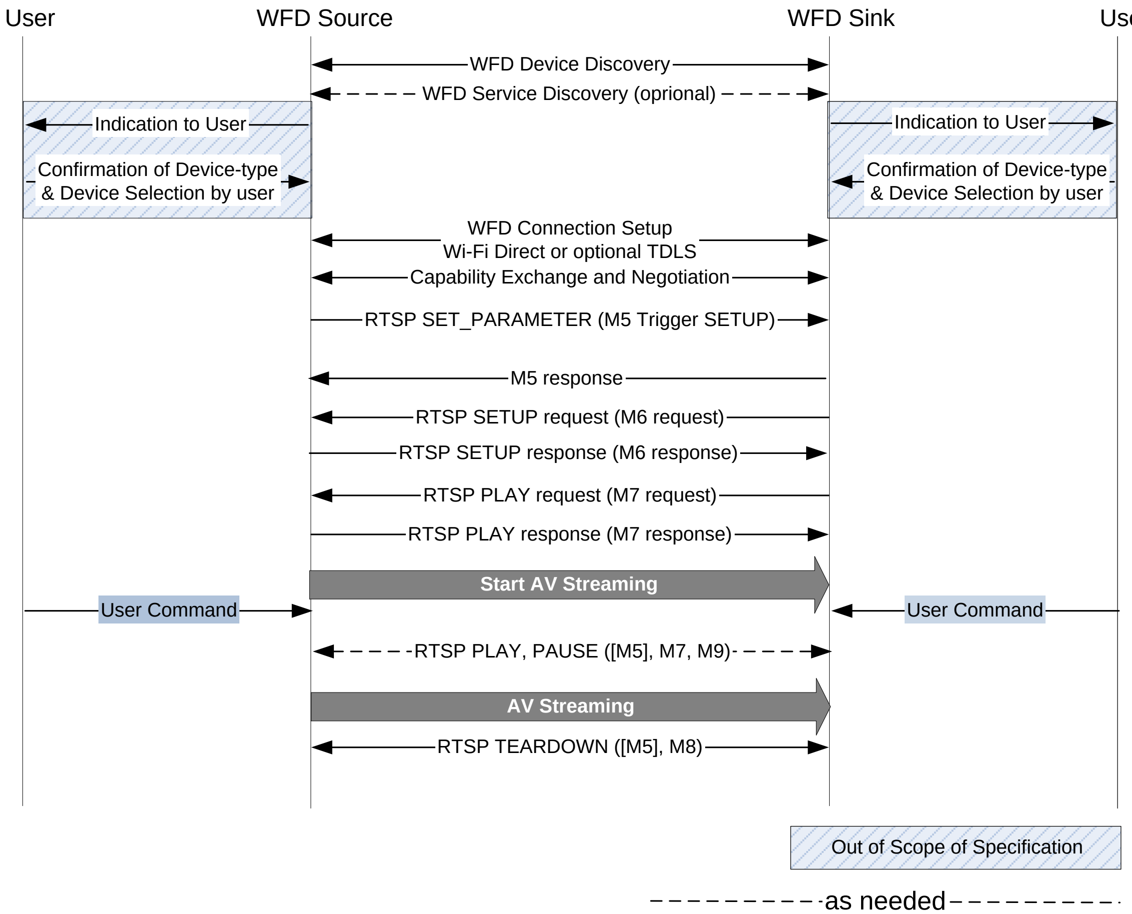


Figure 14. Time-line of a WFD session

The following describes the messages depicted in the diagram above:

- RTSP M5 Messages (see section 6.4.5): The WFD Source sends an RTSP SET_PARAMETER Request (RTSP M5 Trigger SETUP Request) message containing the trigger parameter SETUP. The WFD Sink responds with an RTSP SET_PARAMETER Response (RTSP M5 Response).
- RTSP M6 Messages (see section 6.4.6):
  - RTSP M6 Request: After a successful exchange of an RTSP M5 message containing a wfd-triggered-method parameter with the trigger method set to SETUP, the WFD Sink sends an RTSP SETUP Request (RTSP M6 Request) to the WFD Source.

- RTSP M6 Response: The WFD Source responds to the RTSP SETUP Request (RTSP M6 Request) with an RTSP Setup Response message. If the status code is an RTSP OK, the RTSP session establishment is successful, as described in [20].
- Upon successful RTSP session establishment:

  The WFD Source may send the following messages:

  RTSP M5 Request (trigger) messages to trigger the WFD Sink to send an RTSP PLAY (RTSP M7) or RTSP TEARDOWN (RTSP M8) Request messages to the WFD Source (see section 6.4.5).

  The WFD Sink may send the following message:

  RTSP M8 Request (TEARDOWN) messages to tear down the RTSP session between the WFD Source and the WFD Sink (see section 6.4.8).

- RTSP M7 Messages (see section 6.4.6): After a successful M6 message exchange, the WFD Sink sends an RTSP PLAY Request (RTSP M7 Request) to the WFD Source. This indicates to the WFD Source that the WFD Sink is ready to receive the RTP stream. The WFD Source responds with an RTSP PLAY (RTSP M7) Response. If the status code is an RTSP OK, the WFD Session establishment is successful.
- Upon successful WFD Session establishment:
  - The WFD Source may send the following messages in any order:

    RTSP M3 Request (GET_PARAMETER) to obtain capabilities on one or more RTSP parameters supported by the WFD Sink (see section 6.4.3),

    RTSP M4 Request (SET_PARAMETER) to set values for one or more RTSP parameters corresponding to the WFD Session between the WFD Source and the WFD Sink for WFD Capability Re-negotiation to update AV format (see section 4.10.3.2) with wfd-av-format-change-timing (see section 6.4.4), or to perform transport switch between UDP and TCP (see section 4.10).

    RTSP M5 Request (TRIGGER) to trigger the WFD Sink to send an RTSP PAUSE (RTSP M9) Request message to the WFD Source (see section 6.4.5),

    RTSP M12 Request (SET_PARAMETER with wfd-standby) to indicate that the WFD Source is entering WFD Standby mode (see section 6.4.12),

    RTSP M14 Request (SET_PARAMETER with wfd-uibc-capability to select the input type(s), input device(s) and other parameter(s) to be used for UIBC (see section 6.4.14), or

    RTSP M15 Request (SET_PARAMETER with wfd-uibc-setting) to enable/disable UIBC (see section 6.4.15).

    Upon receiving the above Request messages from the WFD Source, the WFD Sink responds with the corresponding Response messages.

  - The WFD Sink may send the following messages in any order:

    RTSP M7 Request (PLAY) to start or resume paused streaming of audio and/or video content from the WFD Source to the WFD Sink (see section 6.4.7),

    RTSP M9 Request (PAUSE) to pause the streaming of audio and/or video content from the WFD Source to the WFD Sink (see section 6.4.9),

    RTSP M10 Request (SET_PARAMETER with wfd-route) to request the WFD Source to change the audio rendering device under Coupled Sink Operation (see section 6.4.10),

    RTSP M11 Request (SET_PARAMETER with wfd-connector-type) to indicate the change of active connector type to the WFD Source (see section 6.4.11),

    RTSP M12 Request (SET_PARAMETER with wfd-standby) to indicate that the WFD Sink is entering WFD Standby mode (see section 6.4.12),

    RTSP M13 Request (SET_PARAMETER with wfd-idr-request) to request the WFD Source to send an IDR refresh (see section 6.4.13), RTSP M14 Request (SET_PARAMETER with wfd-uibc-capability to select the input type(s), input device(s) and other parameter(s) to be used for UIBC (see section 6.4.14), or

RTSP M15 Request (SET_PARAMETER with wfd-uibc-setting) to enable/disable UIBC (see section 6.4.15).

Upon receiving any of the above Request messages from the WFD Sink, the WFD Source responds with the corresponding Response messages.

- The WFD Source shall send RTSP M16 Request messages to ensure keep-alive of the WFD Sink (see section 6.4.16)

  Upon receiving of the above Request message from the WFD Source, the WFD Sink responds with the corresponding Response messages.

4.8.1 WFD Session between WFD R2 devices over a BSS infrastructure connection

4.8.1.1 Overview

Service discovery over infrastructure with IP connectivity is performed using multicast DNS (mDNS) [56] and DNS Service Discovery (DNS-SD) [50]. mDNS and DNS-SD cover the following three functions to support zero configuration networking:

1. Self-assigned link local addressing
2. Unique link local host naming
3. Service discovery

In the context of service discovery over infrastructure, it is assumed that devices have been assigned with IP addresses and unique host names have been resolved within the local network. mDNS defines several mechanisms to reduce the traffic over the network, including Duplicate Question Suppression (caching at the querier), Known-Answer Suppression (suppression of duplicate responses), Duplicate Answer Suppression, Multi-packet Known-Answer Suppression, exponential back-off and service announcement (see [48] for more information).

DNS-SD supports three operations, namely, publication (service advertisement), discovery (browsing for services) and resolution (resolve instance name to IP address, and port number). The overview of each operation is described in sections 4.8.1.2, 4.8.1.3 and 4.8.1.4, respectively.

4.8.1.2 Publication (Service Advertisement)

To publish a service, an application shall register the service. When a service is registered, PTR, SRV and TXT records are created.

4.8.1.3 Browsing for Services (Discovery)

The service discovery utilizes the DNS record registered during publication to discover a list of instance names, which allows user to make selection based on the list. Browsing for services is achieved using PTR record lookup for a matching service type. The PTR records enable service discovery by mapping the service name to a list of instance names of that service name.

4.8.1.4 Resolution

Resolution is an operation to resolve Instance Name of a service to a host name and IP address. To resolve an Instance Name of a service, an application performs a SRV record lookup with the Instance Name of the service. The mDNS responder responds with the SRV record containing the service instance’s host name and port number. The application may then resolve the host name to an IP address, and begin using the service on the specified port.

4.8.1.5 DNS-SD Procedures over Infrastructure

The DNS-SD over infrastructure supports Active subscription and solicited advertisement protocol. The protocol relies on the publication, browsing and resolution of the DNS-SD.

Figure 15 shows the overall procedures in which a Seeker is acting as an Active Subscriber while the Advertiser is acting as a Solicited Advertiser. A service is registered at the Advertiser in which the PTR, SRV and TXT records are created and cached. A user initiates a WFD service on Device acting as a Seeker.

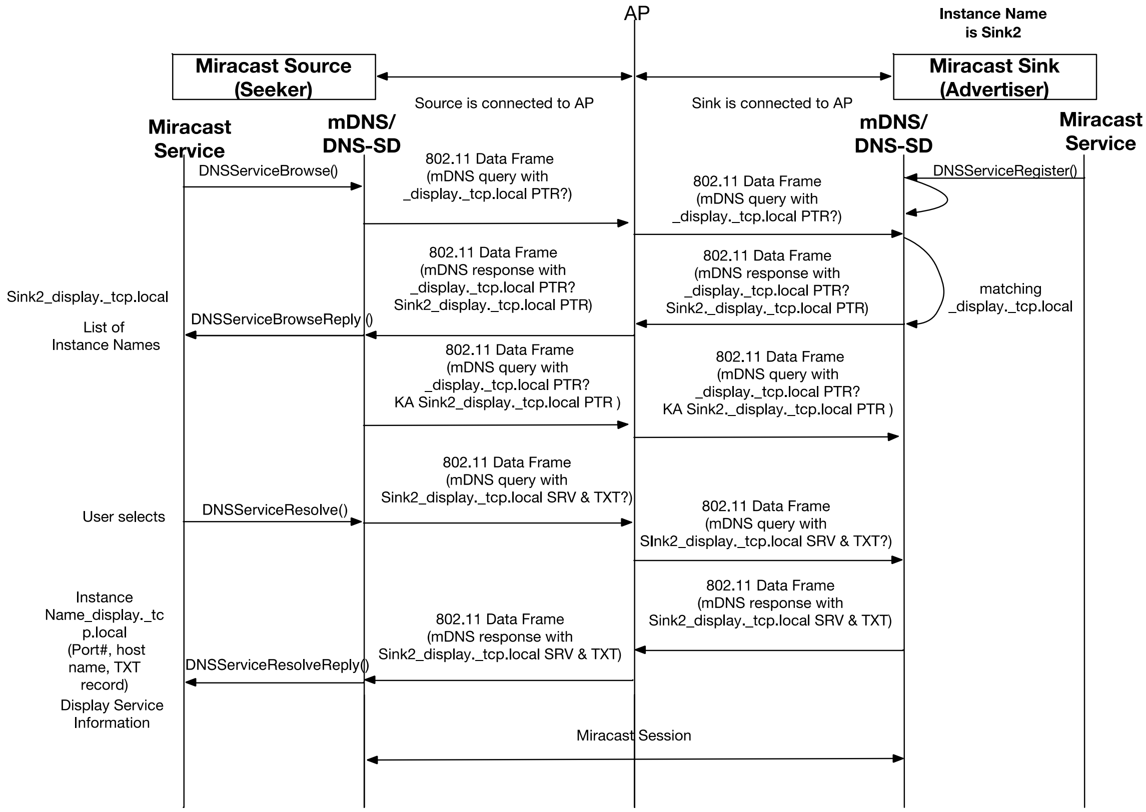


Figure 15. DNS-SD for active subscription and solicited advertising

The Seeker shall send an mDNS query for PTR record lookup from UDP port 5353 using standard multicast address 224.0.0.251 for IPv4 or FF02:FB for IPv6, to the Wi-Fi infrastructure network through the AP. The mDNS query for PTR record lookup contains the PTR record name "_display._tcp.local", indicating that the Seeker is searching for a list of Instance Names of Service Type "_display._tcp".

The mDNS responder at the Advertiser on the Wi-Fi infrastructure network that receives the mDNS query shall perform Service Type "_display._tcp" matching. Only devices with matching Service Type shall respond with an mDNS Response, which contains a PTR record giving its Instance Name.

The application on the Seeker device extracts the Instance Name from the PTR record and presents to the user a list of the Instance Names corresponding to the discovered Advertiser devices. The user selects a particular Advertiser device from the on-screen list and the Seeker device shall send mDNS query to the selected Advertiser device to obtain TXT and SRV record associated with that Instance Name. The mDNS responder at the Advertiser may include SRV records, TXT records and all address records (type "A" and "AAAA") when responding to the PRT record lookup by the Seeker. In this case, the SRV lookup is skipped.

The Seeker shall include a Known-Answer list in its next PTR record lookup so that all Advertisers that have previously responded shall not respond to the same query.

The selected Advertiser device (i.e., "Sink2_display._tcp.local") shall respond with the mDNS Response containing the SRV and TXT record. With the SRV and TXT record, the application at the Seeker shall initiate the WFD Session by performing WFD Capability Negotiation followed by WFD session establishment.

4.8.2 WFD Session between a WFD R2 Device and a WFD R1 Device

WFD Session establishment between a WFD R1 Device and a WFD R2 Device shall be based on WFD R1 discovery, connection and setup procedures. Refer to section 4.3 and section 4.5. The WFD R2 Device Information subelement and mDNS/DNS-SD session setup mechanisms shall not be used for establishing a WFD Session with WFD R1 devices.

4.9 Coupled Sink Operation

A WFD Source may optionally support Coupled Sink Operation. Coupled Sink Operation is possible only if the WFD Source, the Primary Sink and the Secondary Sink support Coupled Sink Operation. In such a configuration, the WFD Source streams the video content to the Primary Sink and has the ability to route the corresponding audio content either to the Primary Sink or to the Secondary Sink. The methods to determine when to switch from rendering the audio payload at the Primary Sink to rendering of the audio payload at the Secondary Sink, and vice versa, are implementation dependent and outside the scope of this specification.

The underlying connection between the WFD Source and the WFD Sinks can be either Wi-Fi P2P or TDLS. Note that if WFD devices intend to establish a WFD Session among themselves using TDLS while they are Wi-Fi P2P clients of the same Wi-Fi P2P GO, they can do so only if the Wi-Fi P2P GO supports intra-BSS distribution.

Prior to the Coupled Sink Operation, a ‘Coupled’ state needs to be established between the Primary and Secondary Sinks. The coupling procedure is almost identical to the WFD Capability Negotiation described in section 4.6 (see section 4.9.1 for more detail). RTSP messages exchanged during the process of coupling are described in section 4.9.1. This enables the Primary Sink and the Secondary Sink to present a Coupled status to a WFD Source during WFD Device Discovery as described in section 4.3. Figure 16 depicts the timeline of a WFD Session using a Coupled Sink Operation.

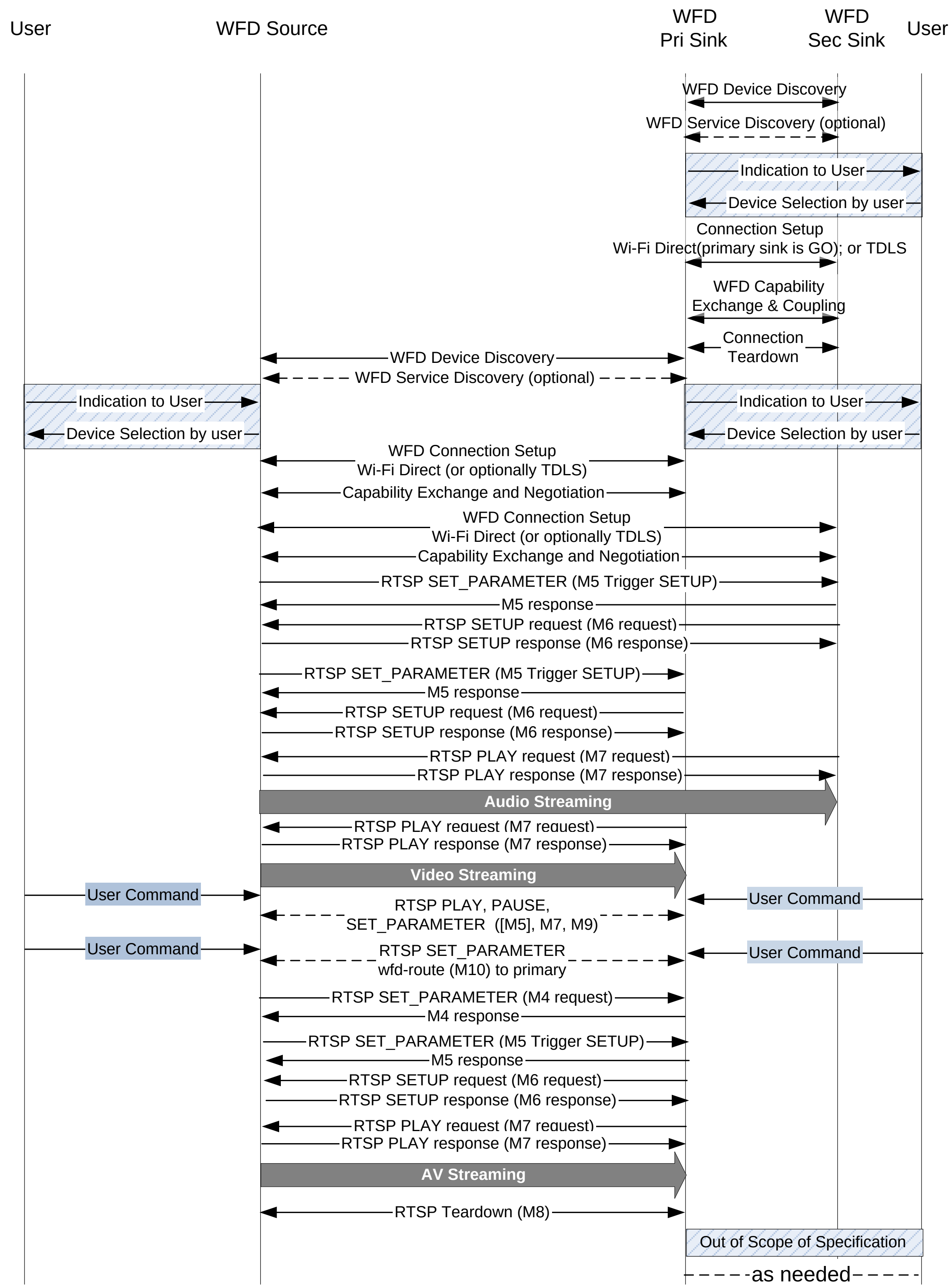


Figure 16. Timeline of a WFD Session with a Secondary Sink

4.9.1 Primary and Secondary Sink Coupling

The following steps establish coupling between two WFD Sinks, if both the Primary Sink and the Secondary Sink support Coupled Sink Operation:

1. Device Discovery using WFD IE:

   A Primary Sink and a Secondary Sink become aware of each other's existence through the exchange of the WFD IE (which among other information, specifies device-type and device-status) during the Device Discovery Phase as described in section 4.3. A Primary Sink may attempt to couple with a Secondary Sink based on its availability as indicated by the Coupled Sink Status field. The criteria by which a Primary Sink determines to couple with a Secondary Sink are implementation dependent and outside the scope of this specification.

2. Service Discovery using the WFD Service Discovery Procedure:

   Prior to coupling, both Primary and Secondary Sinks may exchange detailed WFD capability information using the WFD Service Discovery Procedure as described in section 4.4.

3. Connection Establishment:

   The WFD Sinks establish a connection between themselves as described in sections 4.5.1, 4.5.2 and 4.5.3.

   During the process of TCP connection establishment, the Primary Sink takes the role of a WFD Source and the Secondary Sink takes the role of the WFD Sink in the procedure defined for the case of the WFD Source and the WFD Sink specified in section 4.5.4.

4. Coupling:

- The coupling procedure is similar to the WFD Capability Negotiation procedure described in section 4.6, but the role of each WFD Device is different. In terms of which device is the message requester, and which is the message responder, when referring to section 4.6 and chapter 6 for RTSP message exchange the Primary Sink takes the role of a WFD Source and the Secondary Sink takes the role of the WFD Sink.
- RTSP M1 Messages (see section 6.4.1): The Primary Sink sends an RTSP OPTIONS Request message in order to determine the set of RTSP methods (see section 6.1.29) supported by the Secondary Sink. On receipt of an RTSP M1 (RTSP OPTIONS) Request message from the Primary Sink, the Secondary Sink responds with an RTSP M1 (RTSP OPTIONS) Response message that lists the RTSP methods supported by the Secondary Sink.
- RTSP M2 Messages (see section 6.4.2): After a successful RTSP M1 message exchange, the Secondary Sink sends an RTSP OPTIONS Request message in order to determine the set of RTSP methods (see section 6.1.29) supported by the Primary Sink. On receipt of an RTSP M2 (RTSP OPTIONS) Request message from the Secondary Sink, the Primary Sink responds with an RTSP M2 (RTSP OPTIONS) Response message that lists the RTSP methods supported by the Primary Sink.
- RTSP M3 Messages (see section 6.4.3):
  - RTSP M3 Request: After a successful RTSP M2 message exchange, the Primary Sink sends an RTSP GET_PARAMETER Request message (RTSP M3 Request) that includes the wfd-audio-codecs (see section 6.1) parameter to the Secondary Sink.
  - RTSP M3 Response: The Secondary Sink responds with an RTSP GET_PARAMETER Response message (RTSP M3 Response).

    Note: The Primary Sink could use the latency values from the wfd-audio-codecs parameter to accomplish functions like lip sync.

- RTSP M4 Messages (see section 6.4.4):
  - RTSP M4 Request: After a successful M3 message exchange, the Primary Sink shall send an RTSP SET_PARAMETER Request message to the Secondary Sink, containing wfd-video-formats and wfd-3d-formats parameters that describe support for video and stereoscopic 3D at the Primary Sink.
  - RTSP M4 Response: The Secondary Sink responds with an RTSP M4 Response.

    Note: The Secondary Sink could use the latency values from the wfd-video-formats or the wfd-3d-formats parameter to perform functions like lip sync.

  - At the end of a successful RTSP M4 Request/Response message exchange, the Primary Sink and the Secondary Sink are Coupled. Subsequently, both the Primary and the Secondary Sinks indicates their coupling status as "Coupled" in their respective wfd-coupled-sink parameter when responding to RTSP GET_PARAMETER Requests and in the Coupled Sink Information subelement within the WFD IE.

5. Connection Teardown

   Once a pair of WFD Sinks is Coupled, they tear-down their TCP connection and their L2 connection as follows:

- The TCP connection for RTSP is torn-down using the RTSP TEARDOWN message.
- If connected using Wi-Fi P2P, one of the WFD Sinks transmits a Deauthentication frame to its peer entity as described in the Deauthentication procedure defined in [7].
- If connected using TDLS, one of the WFD Sinks transmits a TDLS teardown frame to its peer entity as specified in [13].

Once Coupled, both the Primary and the Secondary Sinks shall respond to RTSP GET_PARAMETER Requests for the wfd-coupled-sink parameter with the status field set to ‘Coupled’ thus enabling a WFD Source to discover the pair of Primary and Secondary Sinks as per the methods described in section 4.9.2.

Once Coupled and when transmitting frames including the WFD IE, both the Primary and the Secondary Sinks shall set Coupled Sink Status bit (B1B0) to 0b01 (‘Coupled’) in the Coupled Sink Information subelement within the WFD IE.

A Primary Sink may tear-down a previously established coupling with a Secondary Sink at any time by re-establishing a connection (this step is required only if not currently connected) and transmitting an RTSP SET_PARAMETER Request with the wfd-coupled-sink parameter set to the ‘Teardown Coupling’ status. Subsequently, both the Primary and the Secondary Sinks shall indicate their status as ‘Not coupled’ in their respective wfd-coupled-sink parameters when responding to RTSP GET_PARAMETER requests for the wfd-coupled-sink parameter in an RTSP procedure and in the Coupled Sink Information subelement within the WFD IE.

4.9.2 WFD Device Discovery

A WFD Source and Coupled WFD Sinks discover one another as per the mechanisms defined in section 4.3.

A WFD Source may discover a Primary Sink (or a Secondary Sink) and indirectly discover the Coupled Secondary Sink (or a Primary Sink) using the Coupled Sink Information subelement. It may then determine the presence of the Coupled WFD Sink by a directed Probe Request.

4.9.3 WFD Service Discovery

A WFD Source and Coupled WFD Sinks may optionally exchange additional WFD capabilities using the WFD Service Discovery protocol as specified in section 4.4.

4.9.4 WFD Device Pairing

A WFD Source and a Coupled pair of WFD (Primary & Secondary) Sinks may establish WFD Connection using either Wi-Fi P2P or (optionally) TDLS, as the underlying connectivity mechanism. The connectivity scheme to be used for the WFD Session shall be resolved as per the rules defined in Table 12.

Table 12. Connectivity resolution scheme when using Secondary Sinks

PC (WFD Source)	PC (Primary Sink)	PC (Secondary Sink)	Associated with same BSSID	Resolved Connectivity Scheme
0	0	0	Do not care	Wi-Fi P2P
0	0	1	Do not care	Wi-Fi P2P
0	1	0	Do not care	Wi-Fi P2P
0	1	1	Do not care	Wi-Fi P2P
1	0	0	Do not care	Wi-Fi P2P
1	0	1	Do not care	Wi-Fi P2P
1	1	0	Do not care	Wi-Fi P2P
1	1	1	No	Wi-Fi P2P
1	1	1	Yes	TDLS

If the underlying connectivity mechanism is resolved as Wi-Fi P2P, one of the following sequences is used to setup the corresponding connection between the WFD Source and WFD Sink(s). Note that this rule is only applied to the WFD Session for Coupled Sink Operation in which there are both a Primary Sink and a Secondary Sink. If there is one WFD Sink, either one of the WFD Source or the WFD Sink may become a GO, as specified in section 3.2.1.

a. If a P2P Group exists with the WFD Source as the P2P GO, the Primary and the Secondary Sinks join the existing P2P Group, or

b. The Primary or the Secondary Sink initiates a P2P Group formation procedure with the WFD Source setting the GO-intent to zero as defined in [7], indicating preference not to be the GO for the P2P Group (prior to the ensuing GO-negotiation phase of the Wi-Fi P2P connection setup), or

c. The WFD Source initiates a P2P Group formation procedure with the Primary Sink setting its GO-intent to indicate 'GO-only' prior to the ensuing GO-negotiation phase of the Wi-Fi P2P connection setup.

If the underlying connectivity mechanism is resolved as TDLS, the WFD Source sets up TDLS links with both the Primary and Secondary Sinks respectively as per the methods specified in [13].

4.9.5 WFD capability negotiation

Once the WFD Connection is setup, the WFD Source performs WFD Capability Negotiation with both the Primary and Secondary Sinks as described in Figure 13 and section 4.6, thereby completing the pairing process.

4.9.6 WFD Session establishment

If the WFD Source, the Primary Sink and the Secondary Sink support Coupled Sink Operation, using the information derived from the WFD Capability Negotiation, the WFD Source initiates the WFD Session establishment procedure with the Primary and the Secondary Sinks. The WFD Session establishment message exchange is shown in Figure 15. The procedure of the WFD Session establishment for the optional Coupled Sink Operation is identical to the one described in section 4.8 except for the following:

- The WFD Source shall teardown any existing audio only WFD Session with the Secondary Sink, before establishing a WFD Session for Coupled Sink Operation including that Secondary Sink.
- The WFD Source shall initiate the WFD Session establishment procedure with both the Primary and Secondary Sinks. To establish the WFD Session for the Coupled Sink Operation, both procedures shall be successful. Refer to section 6.5 for timeout rules to establish a WFD Session.
- The initiation of AV streaming depends on the implementation of the WFD Source. The behavior of the WFD Source with respect to the receipt of RTSP M7 messages from the WFD Sinks is an implementation detail and is outside the scope of this specification.
- Once the WFD Session is successfully established the Primary or the Secondary Sink may at any time send an RTSP M10 message to change the sink at which the audio stream is rendered.

4.10 AV streaming and control

Higher-layer data services including the transport of audio/video payload and exchange of control information, such as deriving the HDCP2.x session keys, are protocols layered over IP.

4.10.1 Time synchronization

Although the audio and video MPEG2-TS streams carry embedded PCR information allowing for the AV decoder to perform clock recovery and render the AV content appropriately; the use of Wi-Fi as the underlying transport is likely to induce a higher degree of jitter than can be tolerated by typical AV decoder implementations, leading to errors in clock recovery, subsequent frame-loss, and glitch events. Additionally, in the case of a WFD Session involving Coupled WFD Sinks, time synchronization may be needed to maintain synchronization between the rendering of audio and video at different WFD Sinks, i.e., lip-sync.

To preserve audio/video fidelity and compensate for jitter, the local clocks at the WFD Source and WFD Sink(s) may require synchronization. The time synchronization mechanism supported by a WFD Device is indicated using the Time Synchronization Support bit in the WFD Device Information field contained in the WFD Device Information subelement.

By setting the Time Synchronization Support bit to one, a WFD Device indicates that it supports the use of the generalized Precision Time Protocol (gPTP) specified in IEEE 802.1AS [15], clause 12, “Media-dependent layer specification for IEEE 802.11 links” for time synchronization. If supporting this mechanism, the WFD Source assumes the role of the grandmaster clock in the gPTP domain. On a WFD Sink supporting this mechanism, synchronization to the grandmaster clock is achieved by exchanging 802.11v [28] Timing Measurement messages with the WFD Source. When this mechanism is supported, a WFD Source and a WFD Sink shall use a synchronized clock source with a clock source used for PCR for this time measurement.

With its clock synchronized to the WFD Source, a WFD Sink may then employ the use of an implementation-specific de-jittering mechanism to minimize the jitter introduced over the Wi-Fi network, as shown in Figure 17.

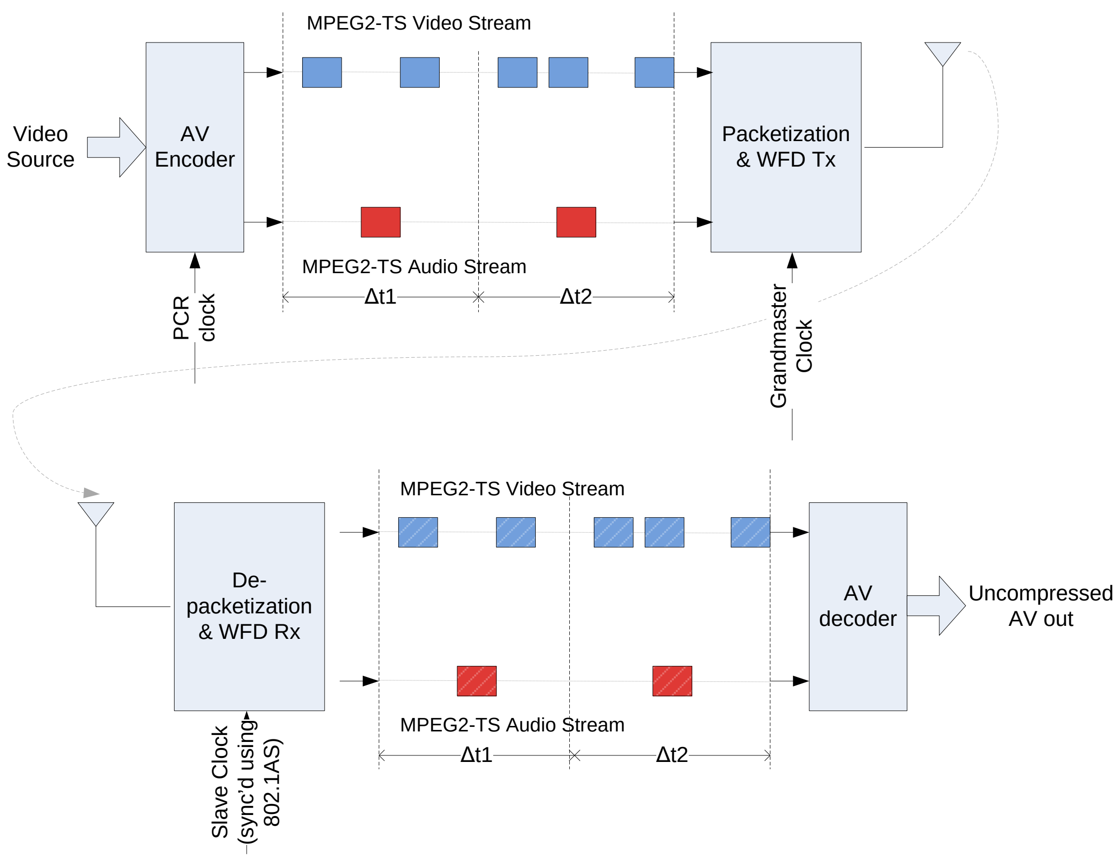


Figure 17. Eliminating Network Jitter using RTP

The AV decoder uses the de-jittered WFD Packets to recover the PCR clock and correctly decode and render content.

4.10.2 AV streaming

Audio/Video elementary streams generated by a WFD Source shall be packetized using a MPEG2-TS container format and encapsulated by RTP/UDP/IP headers prior to 802.11 packetization and transmission to the WFD Sink. The MPEG System Layer is specified in [2] and Appendix A provides an overview.

Figure 18 depicts the structure of a WFD AV frame as generated by a WFD Source.

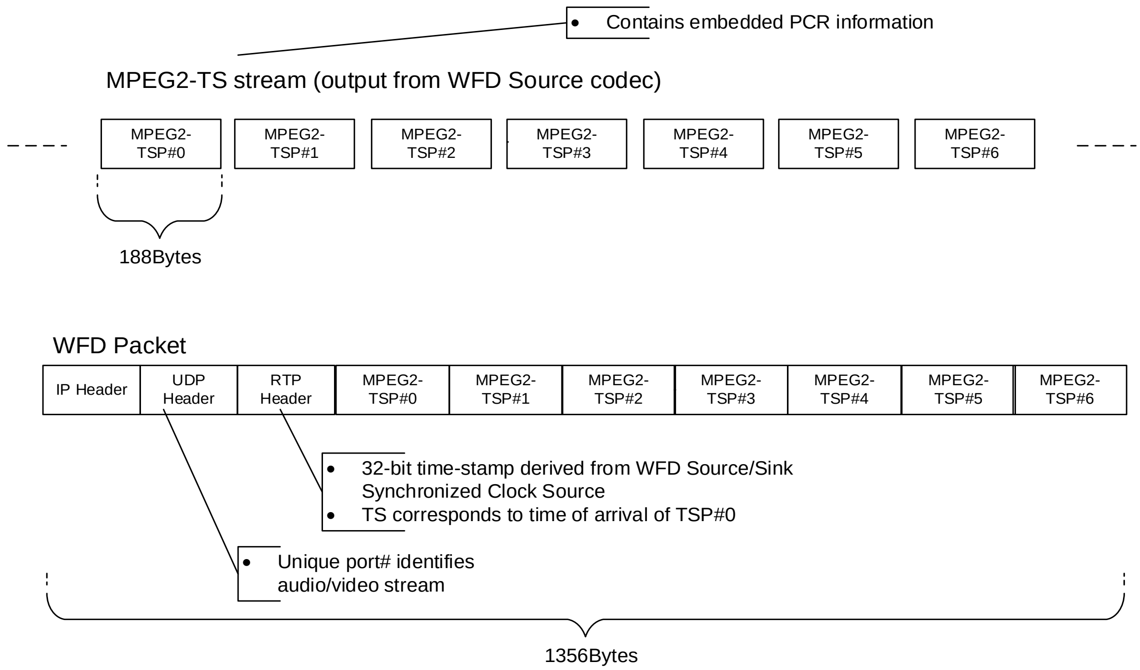


Figure 18. WFD AV packet format

A WFD Source shall support multiplexing the audio and/or video streams into a single MPEG2 transport stream containing audio and/or video, as well as program clock reference (PCR) information. Only in the case of Coupled Sink Operation, the WFD Source may optionally support generating separate MPEG2 transport streams for audio and video, each containing its own program clock reference (PCR) time-stamp information.

Settings of the MPEG2-TS parameters for audio/video streams are specified in Appendix B

The MPEG2-TS packets are packetized using RTP, UDP and IP headers respectively, as shown in Figure 18. The method for encapsulating an MPEG2-TS into RTP packets is specified in Appendix B.1.

The 32-bit RTP time-stamp (RTP TS) information [3] is derived from the master-clock maintained by the WFD Source, and is represented in 90 kHz units where one tick corresponds to 11.11us. The RTP TS for a WFD packet is set to map to the time of arrival of the first MPEG2-TS packet at the RTP encapsulation layer.

The relationship between MPEG2-TS usage and the setting at wfd-client-rtp-ports parameter in both RTSP M3 Response and M4 Request messages are specified in section 6.1.10.

Additional rules for usage of H.264 and H.222 specifications are specified in Appendix D.

When the WFD Source transmits video only content to the Primary Sink, it may transmit either a MPEG2-TS containing multiplexed an audio elementary stream and a video elementary stream or a MPEG2-TS containing only a video only elementary stream. In the former case, the audio payload shall use the mandatory format, i.e., LPCM, 16bits, 48ksps and 2ch, and the audio payload shall carry either actual audio data or null data.

A WFD Source shall support transmitting video elementary stream where one picture is constructed by one slice. A Primary Sink shall support receiving video elementary stream where one picture is constructed by one slice.

A WFD Source may support transmitting video elementary stream where one picture is constructed by multiple slices. A Primary Sink may support receiving video elementary stream where one picture is constructed by multiple slices.

Indication of this optional feature is described in section 5.1.5 (2D video) and 5.1.6 (3D video) for WFD Service Discovery, and in section 6.1.3 (2D video), 6.1.4 (3D video) and 6.1.22 (2D video for R2) for RTSP messages. Operation rules for using multiple slices per a picture are described in Appendix D.

4.10.3 AV Encoding Rate Control

The WFD Source may change the encoding rate depending on various factors such as channel condition, power consumption optimization, and so on. The policy for how the encoding rate control is selected is outside the scope of this specification.

There are two methods for adjusting the encoding rate control at the WFD Source:

1. An implicit method without WFD Capability Re-negotiation via the exchange of RTSP M4 Request /Response messages.

   The WFD Source may use, for example, compression ratio change, macroblock skipping, or frame skipping. The former two can be handled within the H.264 standard, and no additional rules need to be included in this specification. Video frame skipping requires some rules to ensure interoperability, and they are specified in section 4.10.3.1.

2. Initiated explicitly using an RTSP M4 Request/Response message exchange.

   For example, the WFD Source may request changes in the actual display frame rate, display resolution, and so on. This method requires some rules to ensure interoperability, and they are specified in section 4.10.3.2.

4.10.3.1 Video Frame skipping

If the WFD Sink supports video frame skipping, the WFD Source may skip transmission of some of the encoded video frames to the WFD Sink. As a result, the effective video frame rate may be lower than the one negotiated between the WFD Source and WFD Sink during the WFD Capability Negotiation phase. Although the WFD Source decides which encoded video frames to skip, the policy that defines which of the encoded video frames are skipped is outside the scope of this specification.

A WFD Sink, which indicates this capability, shall tolerate video frame skipping, as long as the interval between two transmitted video frames is smaller than or equal to the interval indicated in the Max Skip Interval bits in the frame-rate-control-bitmap in the wfd-video-format or wfd-3d-formats or wfd2-video-formats parameters in the RTSP M3 messages from the WFD Sink.

In all cases, after any lapse in encoded transmissions, the WFD Source shall ensure that SCR updates are current, and DTS/PTS values are consistent as described in [2].

When video frame skipping is applied, the following rules shall be satisfied.

- A video frame shall not be skipped if it is referenced by another video frame that is not skipped (i.e., when a video frame refers to the other video frame(s), it shall refer to video frame(s) that are not skipped).
- The WFD Source shall maintain the time interval between two successive video frames to be less than or equal to the maximum/allowable time-interval indicated using the Max Skip Interval bits (B3:B1) in frame-rate-control-bitmap included in the wfd-video-formats parameter in the RTSP M3 Response message from the WFD Sink.

  Note: In the case where Max Skip Interval bits (B3:B1) are set to all zeros (indicating the maximum allowable time interval is unspecified), any interval between two transmitted video frames is acceptable and the behavior of the WFD Sink for scene change recovery is implementation specific.

- When an interlaced format is used, a correct pair of even and odd fields which construct a video frame shall be transmitted at a nominal time interval.
- Frame number (frame_num) in the slice header(s) shall be sequential numbers for transmitted video frames (i.e., skipped video frames are not counted).
- TS packets carrying PCR shall be transmitted at intervals equal to or less than 100msec, as specified in MPEG2-TS specification [2].
- The WFD Source should transmit two (or more) video frames with a nominal interval during the period that does not have skipped frame(s) so that the decoder can update output data.

4.10.3.2 Explicit AV format change

During the lifetime of a WFD Session, the WFD Source may want to update the parameter set in use which was previously negotiated during the WFD Capability Negotiation phase just prior to the WFD Session establishment.

A WFD Source may update the parameter set used in the WFD Session at any time by transmitting an RTSP SET_PARAMETER Request (RTSP M4 Request) message containing the appropriate parameter updates (described in section 6.1) to the WFD Sink(s). The updated parameter set takes effect after the WFD Source receives a corresponding RTSP SET_PARAMETER Response containing a status code of RTSP OK, from the WFD Sink(s). This RTSP message exchange is called WFD Capability Re-negotiation.

When changing the AV format during one RTP session, the RTSP M4 Request message shall include the wfd-av-format-change-timing (specified in section 6.1.3) parameter.

When changing video format, i.e., resolution or refresh rate, the WFD Source shall transmit an IDR picture just after changing video format(s) to reset parameters, as specified in [1]. It should be noted that transmission of an IDR picture may require higher instantaneous peak throughput even if the resolution and/or refresh rate after change is lower than the previous values.

If the Primary Sink sets the Video Frame Rate Change Support bit (B4) of the frame-rate-control-support field in the wfd-video-formats parameter or the wfd-3d-formats parameter or the wfd2-video-formats parameter (R2 Primary Sink only) in the RTSP M3 Response message to zero, then the WFD Source shall not change the video frame rate when performing explicit AV format change(s) using WFD Capability Re-negotiation without user intervention.

If the Primary Sink sets the Video Frame Rate Change Support bit (B4) of the frame-rate-control-support field in the wfd-video-formats parameter or the wfd-3d-formats parameter in the RTSP M3 Response message to one, then the WFD Source may change the video frame rate when performing explicit AV format change using WFD Capability Re-negotiation without any user intervention. With user intervention, the WFD Source may change video frame rate when performing explicit AV format change using WFD Capability Re-negotiation regardless of the value of this bit setting. If this bit is set to zero by the Primary Sink, visible video artifacts may occur at the Primary Sink during the transition from one refresh rate to the next.

The rules specified below apply only to automatic AV format change(s) using the RTSP M4 Request message during one RTP session, without any user intervention.

When the AV format changes as a result of user intervention (e.g., a program change at the WFD Source by selecting a different channel, or audio routing changes from a Primary Sink to a Secondary Sink), or as a result of suggestion received from an R2 Primary Sink as described in section 4.10.4, it is not necessary to follow the rules specified below.

During a WFD Session, the video codec and the video codec profile in use shall not be changed in an RTSP M4 Request message or in an MPEG2-TS.

During a WFD Session, a WFD Source may change H.264 Level in use by indicating this change in an RTSP M4 Request message or in an MPEG2-TS. However, changing H.264 level from a lower level to a higher level during a WFD Session may cause a decoder reset at the WFD Sink and may result a discontinuity of video rendering. To avoid such discontinuities, it is recommended that a WFD source sets the highest H.264 level that is supported by both the WFD Source and the WFD Sink.

During a WFD Session, a WFD R2 Source may change the H.265 Level in use by indicating this change in an RTSP M4 Request message or in an MPEG2-TS. However, changing the H.265 level from a lower level to a higher level during a WFD Session may cause a decoder reset at the WFD R2 Sink and may result in a discontinuity of video rendering. To avoid such discontinuities, the WFD R2 Source should set the H.265 level to the highest that is supported by both the WFD R2 Source and the WFD R2 Sink.

During a WFD Session, the scan method of video, i.e., progressive or interlace shall not be changed in an RTSP M4 Request message or in an MPEG2-TS.

If the WFD Sink supports a refresh rate change, as indicated in the Refresh rate change support field in the wfd-video-formats parameter (specified in section 6.1.3) or in the wfd-3d-formats parameter (specified in section 6.1.4) or the wfd2-video-formats parameter (specified in section 6.1.22 only for WFD R2 devices), the WFD Source may change video refresh rate among supported video refresh rates indicated by the WFD Sink.

During a WFD Session, video resolution and refresh rate shall not be changed simultaneously in an RTSP M4 Request message or in an MPEG2-TS.

During a WFD Session, audio format (i.e., sample rate, bits per sample, the number of channels, audio codec) shall not be changed in an RTSP M4 Request message or in an MPEG2-TS.

4.10.4 AV Session Control

WFD Sources that support Coupled Sink Operation shall support the wfd-route parameter. On receipt of an RTSP M10 Request (RTSP SET_PARAMETER Request) message, the WFD Source shall route the audio stream to the destination specified in the wfd-route (specified in section 6.1.11) parameter contained in the RTSP M10 Request. After transmission of an RTSP M10 Response message with a status code of RTSP OK, the WFD Source sends RTSP M4 Request message(s) to the Primary Sink and/or the Secondary Sink, as follows;

- If the WFD R1 Source transmits an RTSP M10 Response message with a status code of RTSP OK as a response to the RTSP M10 Request message that includes the wfd-route parameter indicating “primary”, the WFD R1 Source shall send an RTSP M4 Request message including the wfd-audio-codecs parameter and either one of the wfd-video-formats parameter or the wfd-3d-formats parameter to the R1 Primary Sink.

  If the WFD R2 Source transmits an RTSP M10 Response message with a status code of RTSP OK as a response to the RTSP M10 Request message that includes the wfd-route parameter indicating “primary”, the WFD R2 Source shall send an RTSP M4 Request message including the wfd-audio-codecs parameter or the wfd2-audio-codecs parameter and one of the wfd-video-formats parameter, the wfd-3d-formats parameter, or the wfd2-video-formats parameter to the R2 Primary Sink.

- If the WFD R1 Source transmits an RTSP M10 Response message with a status code of RTSP OK as a response to the RTSP M10 Request message that includes the wfd-route parameter indicating “secondary” the WFD R1 Source shall send an RTSP M4 Request message including either one of the wfd-video-formats parameter or the wfd-3d-formats parameter (and without the wfd-audio-codecs parameter) to the R1 Primary Sink, and the WFD R1 Source shall send an RTSP M4 Request message including the wfd-audio-codecs parameter (and with neither of the wfd-video-formats nor the wfd-3d-formats) to the R1 Secondary Sink.

  If the WFD R2 Source transmits an RTSP M10 Response message with a status code of RTSP OK as a response to the RTSP M10 Request message that includes the wfd-route parameter indicating “secondary” the WFD R2 Source shall send an RTSP M4 Request message including one of the wfd-video-formats parameter, the wfd-3d-formats parameter or the wfd2-video-formats parameter (and without the wfd-audio-codecs parameter and the wfd2-audio-codecs parameter) to the R2 Primary Sink, and the WFD R2 Source shall send an RTSP M4 Request message including only one of the wfd-audio-codecs parameter and the wfd2-audio-codecs parameter (and with neither of the wfd-video-formats, the wfd-3d-formats nor the wfd2-video-formats parameters) to the R2 Secondary Sink.

4.10.5 WFD video recovery

The video decoder in the WFD Sink receives VCL and NAL packets which could be lost due to intermittently degraded channel conditions. Over time the errors could accumulate and eventually render the decoder inoperable or cause the display to show an image with reduced quality. To prevent the decoder from failing, a WFD Sink may detect the error in the bitstream and request the WFD Source to transmit an IDR picture, using the following procedure.

The WFD Sink may request the IDR picture from the WFD Source by sending a SET_PARAMETER message containing a wfd-idr-request parameter (section 6.1.20). Upon successful reception of the SET_PARAMETER message containing a wfd-idr-request parameter, the WFD Source shall send an IDR picture with SPS and PPS in the bit stream and send a Response message with a status code of RTSP OK at the earliest opportunity.

4.10.6 Switching of Transport Protocol in a WFDS Session

A WFD R2 Session shall support RTP media data delivery uisng both UDP and TCP protocols, and switching between the protocols shall be allowed as needed. The WFD Session establishment shall be completed prior to starting the AV streaming over UDP based on the procedures specified in sections 4.6, 4.7 and 4.8. The transport switch from UDP to TCP or vice versa may be performed any time after the WFD Session establishment using the procedures described in this section.

When the RTP data is sent over TCP transport, each RTP packet shall be framed according to RFC 4571 [43].

The RTSP messages include additional information for switching to TCP from UDP or vice versa for RTP media data streaming and playback control during a WFD R2 Session. To support these capabilities, the feature is summarized by the following new RTSP structures (details of these parameters are described in section 6.1):

- wfd2-transport-switch – to switch the transport between UDP and TCP
- wfd2-buffer-length – query support for amount of buffering supported by WFD R2 Source/Sink
- wfd2-audio-playback-status – query the status of the audio buffer in the WFD R2 Sink that include amount of audio data being currently buffered and current PTS value of the audio being rendered by the WFD R2 Sink
- wfd2-video-playback-status – query the status of the video buffer in the WFD R2 Sink that include amount of video data being currently buffered and current PTS value of the video being rendered by the WFD R2 Sink.

If a WFD R2 Source requires to switch to TCP transport, the WFD R2 Source shall include the wfd2-transport-switch parameter with the profile field set to RTP/AVP/TCP;unicast (as described in section 6.1.25) in the RTSP M4 Request (SET_PARAMETER Request) message to the WFD R2 Sink. If the WFD R2 Sink receives a RTSP M4 Request message that contains the wfd2-transport-switch parameter, the WFD R2 Sink shall respond with the RTSP M4 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. The WFD R2 Source may optionally include the wfd2-buffer-length parameter in the M4 Request message to indicate the value of the buffer length the WFD R2 Source prefers the WFD R2 Sink to use. If the WFD R2 Sink receives the RTSP M4 Request message that contains the wfd2-buffer-length parameter in addition to the wfd2-transport-switch parameter, the WFD R2 Sink may respond with RTSP OK in the M4 Response message that may also include the wfd2-buffer-length parameter to indicate the lower value for the buffer length it can support as described in section 6.1.26.

After sending the M4 Response with RTSP OK as a response to the RTSP M4 Request message containing a wfd2-transport-switch parameter with the profile field set to RTP/AVP/TCP;unicast, the WFD R2 Sink shall send an RTSP M6 Request message to the WFD R2 Source. The RTSP M6 Request message (SETUP) shall include the details for the client's RTP port and may include the optional RTCP port number (when supported) to be used over TCP in the Transport header as specified in section 6.2.4. The WFD R2 Source shall respond with an RTSP M6 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. When the WFD Source sends the RTSP M6 Response message indicating a status code of RTSP OK, the Transport header in the SETUP Response message shall include the details for server's RTP port and may include the optional RTCP port number (when supported) as specified in section 6.2.4.

After the WFD R2 Sink receives the RTSP M6 Response message indicating a status code of RTSP OK from a WFD R2 Source during a transport switch (i.e., when switching from UDP to TCP or vice-versa), the WFD R2 Sink shall flush any pending audio/video data received over the previous transport port. The WFD R2 Source shall start transmitting video stream with an IDR frame after the transport switch. The WFD R2 Source shall use the same SSRC ID field in RTP packet header for RTP data transmitted after the transport switch.

Note: It is possible that the WFD R2 Source may not send PAT/PMT immediately when starting to transmit the video stream over the new transport port after a transport switch. In such case, it is expected that the WFD R2 Sink continues using the PAT/PMT received previously for the video stream until the next PAT/PMT information is received.

When using TCP transport for RTP data, the WFD R2 Sink runs a TCP server and the WFD R2 Source connects to the server. Prior to sending an RTSP SETUP Request (M6 Request) message with client's port information applicable for TCP, the WFD R2 Sink shall start its TCP server and be ready to accept the connection.

If the WFD R2 Source and/or the WFD R2 Sink detects a failure in TCP connection any time after switching to TCP transport protocol, the WFD R2 Source and/or the WFD R2 Sink shall perform TCP connection recovery procedure as described below:

- If the WFD R2 Source detects a TCP connection failure, the WFD R2 Source shall send a TCP SYN packet to the WFD R2 Sink over the same TCP port that was used before the failure. Once the transport switch to TCP has been performed during an active WFD Session, the WFD R2 Sink shall continue listening to the TCP port and shall accept the connection request received from the WFD R2 Source. If the TCP connection over the same TCP port is successfully established, the WFD R2 Source and the WFD R2 Sink shall resume using the existing TCP connection for RTP media data. If the TCP connection recovery over the same TCP port fails, the WFD R2 Source shall perform a transport switch by sending a RTSP M4 Request (SET_PARAMETER Request) message containing the wfd2-transport-switch parameter in which the profile field is set either to RTP/AVP/TCP;unicast when the WFD R2 Source decides to continue in TCP or as RTP/AVP/UDP;unicast when the WFD R2 Source decides to switch back to UDP Transport Protocol as described latter in this section.

- If the WFD R2 Sink detects a TCP connection failure, the WFD R2 Sink shall re-configure the TCP Server for listening to the same TCP port negotiated with the WFD R2 Source by re-opening of the TCP socket or re-binding to the same port as necessary and ready to accept a new connection request coming from the WFD R2 Source. If the WFD R2 Sink fails to open or listen to the same TCP port due to any reason within 60 seconds, the WFD R2 Sink shall send an RTSP SETUP Request (M6 Request) message to the WFD R2 Source containing the TCP port information to be used for the TCP connection. The WFD R2 Source shall respond with an RTSP M6 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. The timeout value applicable for RTSP M6 Request/Response message exchange and the procedure in the event of timeout is specified in section 6.5. If the WFD R2 Source sends the RTSP M6 Response with a status code other than RTSP OK, the WFD R2 Source shall perform a transport switch for switching back to UDP Transport Protocol by using the procedures specified below in this section.

In order to perform the transport switch back to UDP from TCP, the WFD R2 Source shall send the RTSP M4 Request (SET_PARAMETER Request) message by including the wfd2-transport-switch parameter with the profile field set to RTP/AVP/UDP;unicast (as described in section 6.1.25) to the WFD R2 Sink. Once the WFD R2 Sink responds with the RTSP M4 Response message with RTSP OK, the WFD R2 Sink shall send an RTSP M6 Request message to the WFD R2 Source. The RTSP SETUP Request (M6 Request) message shall include the details for the client's RTP port and may include the optional RTCP port number (when supported) to be used over UDP in the Transport header as specified in section 6.2.4. The WFD R2 Source shall respond with an RTSP M6 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. When the WFD Source sends the RTSP M6 Response message indicating a status code of RTSP OK, the Transport header in the SETUP Response message shall include the details for server's RTP port and may include the optional RTCP port number (when supported) as specified in section 6.2.4.

Figure 19 shows a typical message flow for switching the transport between UDP and TCP in a WFD R2 Session.

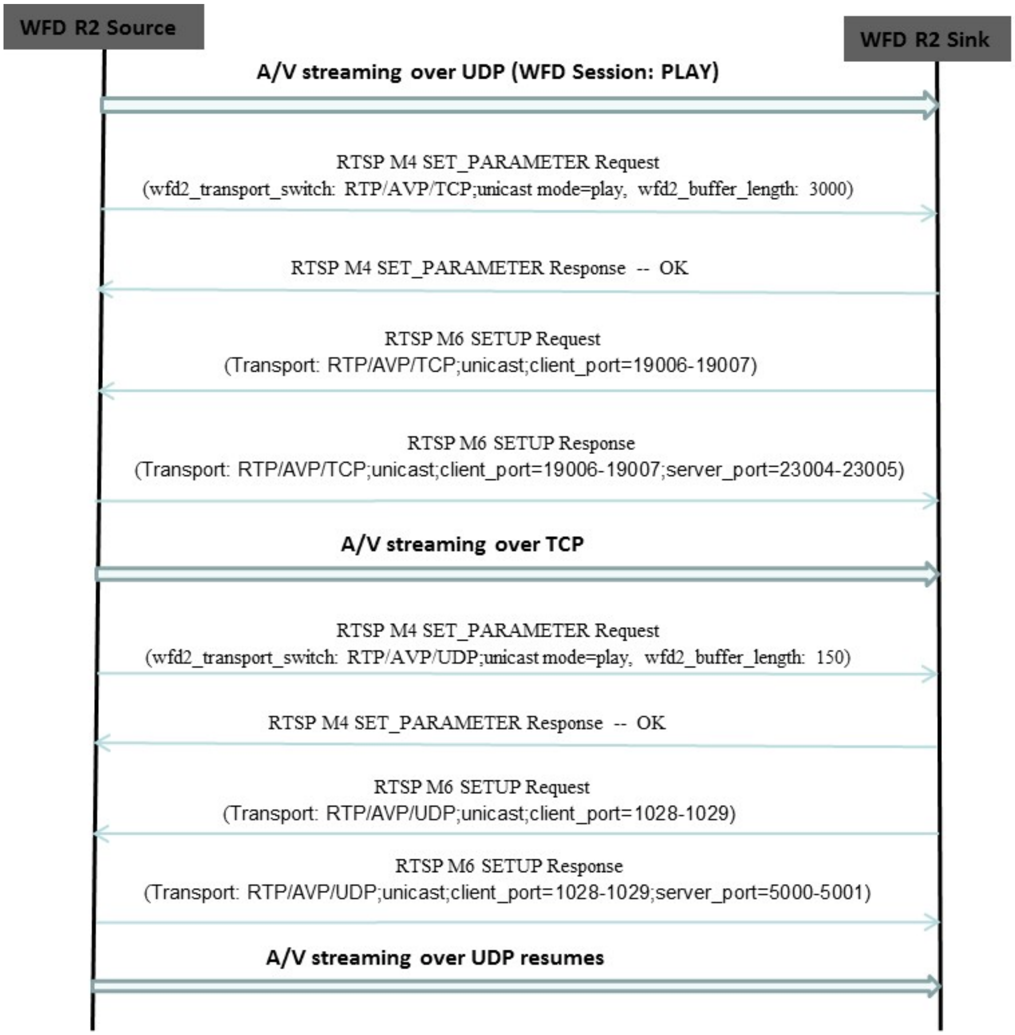


Figure 19. Typical message flow for switching to TCP mode in a WFD R2 Session and switching back to UDP mode

Before performing a transport switch, the WFD R2 Source may send a RTSP GET_PARAMETER (RTSP M3 Request) containing the wfd2-buffer-length, wfd2-audio-playback-status and/or wfd2-video-playback-status parameters, and the WFD R2 Sink shall respond with the corresponding values for the requested parameters in the RTSP M3 Response message as described in sections 6.1.26, 6.1.27 and 6.1.28. The WFD R2 Source may include the wfd2-audio-playback-status and/or wfd2-video-playback-status parameters in the M3 Request to query the current presentation time of the audio and/or video stream being played at the WFD R2 Sink. The wfd2-buffer-length may be included in the M3 Request to query the amount of buffer length that can be supported by the WFD R2 Sink at that time if a transport switch is performed subsequently after the exchange of M3 Request/Response messages. The exchange of these optional parameters (wfd2-buffer-length, wfd2-audio-playback-status or wfd2-video-playback-status) in the M3 Request/Response messages may be performed anytime during an ongoing WFD Session.

4.11 User Input Back Channel

The User Input Back Channel (UIBC) is an optional WFD feature that when implemented facilitates communication of user inputs to a User Interface, present at the WFD Sink, to the WFD Source.

All UIBC user inputs are packetized using a common packet header and transported over TCP/IP. The user input categories include Generic, and HIDC.

The Generic category is used for device agnostic user inputs that are processed at the application level. Generic user inputs are formatted using the Generic Input Body. HIDC is used for user inputs generated by HIDs like remote control, keyboard, etc.[46]. HIDC user inputs are formatted using the HIDC Input Body.

4.11.1 UIBC Data Encapsulation

The TCP payload structure for UIBC including the common packet header for user inputs is shown in Figure 20.

<table>
<tr><th>Bit Offset</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th></tr>
<tr><td>0</td><td colspan="3">Version</td><td>T</td><td colspan="8">Reserved</td><td colspan="4">Input Category</td></tr>
<tr><td>16</td><td colspan="16">Length</td></tr>
<tr><td>32</td><td colspan="16">Timestamp (optional)</td></tr>
<tr><td></td><td colspan="16">UIBC Input Body</td></tr>
</table>

Figure 20. Encapsulations of User Inputs over TCP/IP

The fields in the common packet header are described below:

- Version (3 bits):

  The version of the protocol. This field shall be set to 0b000.

- T (1 bit):

  Presence of the optional timestamp before the input body, where 0 means the timestamp field does not exist, and 1 means the timestamp field exists.

- Reserved (8 bits):

  These bits are reserved for future use, and shall be set to all zeros on transmission and ignored on reception.

- Length (16 bits):

  The length of the entire TCP payload in units of 8 bits, from bit offset 0 to the end of the UIBC Input Body (including padding if any).

- Input Category (4 bits):

  The category of the inputs delivered by this TCP payload. The Input Category codes are shown in Table 13.

Table 13. Input Category Code

Input Category	Category	Notes
0	Generic	User input data is (are) formatted using the Generic Input Body.
1	HIDC	User input data is (are) formatted using the HIDC Input Body.
2-15	Reserved	

- (Optional) Timestamp (16 bits):

  The last 16 bits of the WFD Source marked RTP timestamp of the frames that are being displayed when user inputs are applied.

  After a common packet header, a UIBC Input Body field follows.

- UIBC Input Body:

  This field is either a Generic Input Body or an HIDC Input Body as indicated in Input Category field and contains information describing one or more user inputs. One user input corresponds to one Generic Input message or one

HIDC messagedepending on the selected Input Category. In addition, this field should be padded up to an integer multiple of 16 bits to have an even integer number in the Length field.

4.11.2 UIBC Establishment and Maintenance

The UIBC is established and maintained using RTSP GET_PARAMETER and SET_PARAMETER messages. The message sequences are shown in Figure 21 and Figure 22.

The TCP port number to be used by the WFD Source for UIBC message transactions is included in the wfd-uibc-capability parameter in the RTSP M4 and/or M14 Request message, and this port at the WFD Source shall be ready to accept incoming connections from the WFD Sink before sending a subsequent RTSP M4 and/or M14 Request message containing the wfd-uibc-capability parameter. Once established, a single TCP connection between the WFD Source and the WFD Sink shall be used for the duration of the WFD Session for all UIBC data exchange.

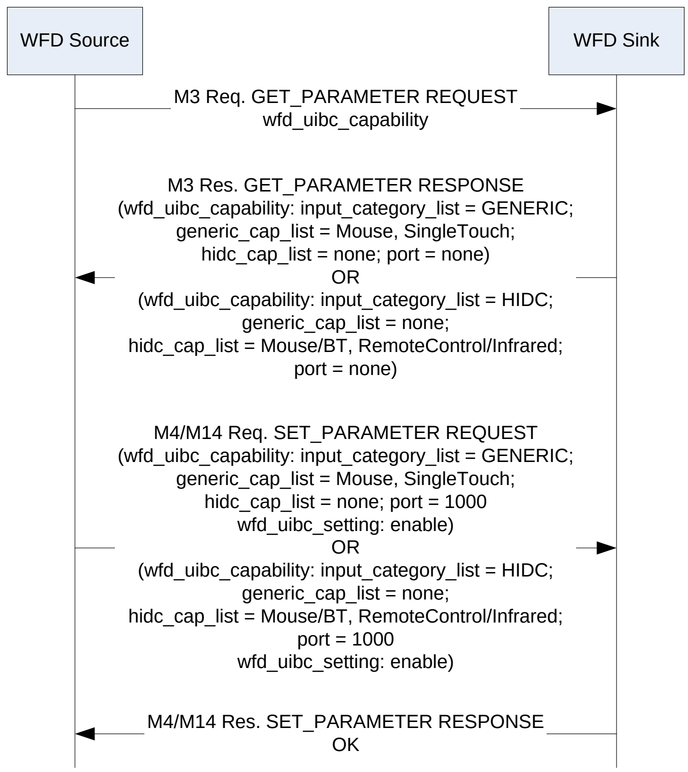


Figure 21. UIBC capability negotiation

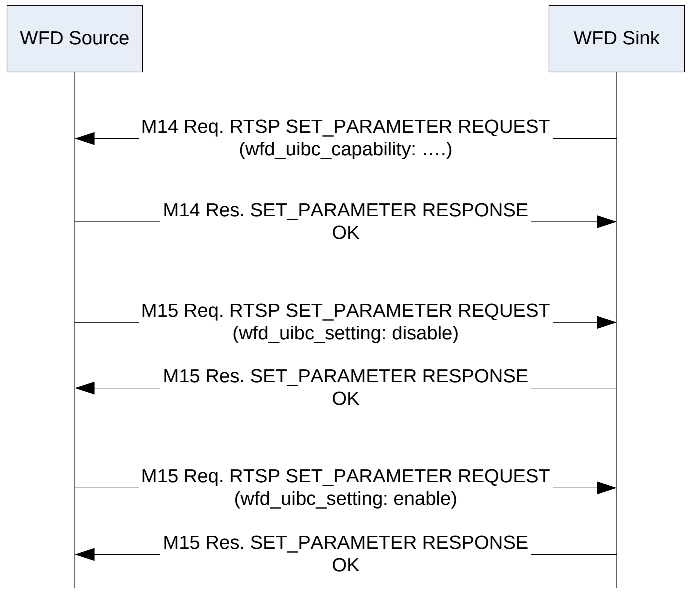


Figure 22. UIBC update

Control information is associated with a user input. Control information of user input includes a query about UIBC capabilities, identifying specific UIBC input type & category and requests to enable or to disable a UIBC. The control information is transmitted over the RTSP control plane. See sections 6.1.15, 6.1.16, 6.4.14 and 6.4.15.

4.11.3 UIBC Input Body

4.11.3.1 Generic Input Body Format

The Generic Input Body has one or more Generic Input messages. The format of each Generic Input message is shown below:

Table 14. Generic input message format

Field	Size (Octet)	Value
Generic Input Type ID	1	Input type such as Zoom In, Scroll, etc. See Table 15
Length	2	Length of the following fields in octets
Describe	Variable	The details of the user inputs

Table 15. Generic Input Type ID for user inputs of the Generic Category

Generic Input Type ID	Notes
0	Left Mouse Down/Touch Down
1	Left Mouse Up/Touch Up
2	Mouse Move/Touch Move
3	Key Down
4	Key Up
5	Zoom
6	Vertical Scroll
7	Horizontal Scroll

Generic Input Type ID	Notes
8	Rotate
9-255	Reserved

The Describe field of Generic Input message for the Left Mouse Down/Touch Down Generic Input Type ID is shown in Table 16. The coordinate origin (0, 0) is defined to be the top-left corner of the rectangular display region.

Table 16. Describe Field of the Generic Input Message for Left Mouse Down/Touch Down

Field	Size (Octet)	Notes
Number of pointers (N)	1	Number of pointers of a multi-touch motion event. When set to 0x01, it indicates a single-touch motion event.
For i = 1: N {		
Pointer ID	1	The identification number of this pointer. The value lies in [0,1,…]
X-coordinate	2	X-coordinate for mouse/touch down event normalized with respect to the negotiated resolution of the video stream.
Y-coordinate }	2	Y-coordinate for mouse/touch down event normalized with respect to the negotiated resolution of the video stream.

The Describe field of the Generic Input message for the Left Mouse Move/Touch Move Generic Input Type ID is shown in Table 17. The coordinate origin (0, 0) is defined to be the top-left corner of the rectangular display region.

Table 17. Describe Field of the Generic Input Message for Left Mouse Move/Touch Move

Field	Size (Octet)	Notes
Number of pointers (N)	1	Number of pointers of a multi-touch motion event. When set to 0x01, it indicates a single-touch motion event.
For i = 1: N {		
Pointer ID	1	The identification number of this pointer. The value lies in [0,1,…]
X-coordinate	2	X-coordinate for mouse/touch move event normalized with respect to the negotiated resolution of the video stream.
Y-coordinate }	2	Y-coordinate for mouse/touch move event normalized with respect to the negotiated resolution of the video stream.

The Describe field of the Generic Input message for the Zoom Generic Input Type ID is shown in Table 18. The coordinate origin (0, 0) is defined to be the top-left corner of the rectangular display region.

Table 18. Describe Field of the Generic Input Message for Zoom

Field	Size (Octet)	Notes
X	2	The reference X-coordinate for the zoom operation normalized with respect to the negotiated resolution of the video stream.
Y	2	The reference Y-coordinate for the zoom operation normalized with respect to the negotiated resolution of the video stream.
Integer times to zoom	1	The unsigned integer portion of the number of times to zoom
Fraction times to zoom	1	The fraction portion of the number of times to zoom. The unit of the fractional part shall be 1/256, and the sign of the fractional part is always positive.

The Describe field of the Generic Input message for the Vertical Scroll Generic Input Type ID is shown in Table 19.

Table 19. Describe Field of the Generic Input Message for Vertical Scroll

Field	Size (Octet)	Notes
Amount to vertically scroll	2	B15B14; Scroll Unit Indication bits. 0b00; the unit is a pixel (normalized with respect to the WFD Source display resolution that is conveyed in an RTSP M4 Request message). 0b01; the unit is a mouse notch (where the application is responsible for representing the number of pixels per notch). 0b10-0b11; Reserved.
		B13; Scroll Direction Indication bit. 0b0; Scrolling down. 0b1; Scrolling up.
		B12:B0; Number of Scroll bits. Number of units for a vertical scroll.

The Describe field of the Generic Input message for the Horizontal Scroll Generic Input Type ID is shown in Table 20.

Table 20. Describe Field of the Generic Input Message for Horizontal Scroll

Field	Size (Octet)	Notes
Amount to horizontally scroll	2	B15B14; Scroll Unit Indication bits. 0b00; the unit is a pixel (normalized with respect to the WFD Source display resolution that is conveyed in an RTSP M4 Request message). 0b01; the unit is a mouse notch (where the application is responsible for representing the number of pixels per notch). 0b10-0b11; Reserved.
		B13; Scroll Direction Indication bit. 0b0; Scrolling to the right. Scrolling to the right means the displayed content being shifted to the left from a user perspective. 0b1; Scrolling to the left. Scrolling to the left means the displayed content being shifted to the right from a user perspective.
		B12:B0; Number of Scroll bits. Number of units for a Horizontal scroll.

The Describe field of the Generic Input message for the Rotate Generic Input Type ID is shown in Table 21.

Table 21. Describe Field of the Generic Input Message for Rotate

Field	Size (Octet)	Notes
Integer portion of rotation amount	1	The signed integer portion of the amount to rotate in units in radians. A negative number indicates to rotate clockwise; a positive number indicates to rotate counter-clockwise
Fraction portion of rotation amount	1	The fraction portion of the amount to rotate in units of radians. The unit of the fractional part shall be 1/256, and the sign of the fractional part is always positive.

4.11.3.2 HIDC Input Body Format

The HIDC Input Body has one or more HIDC messages. The format of each HIDC message is shown in Table 22.

In this case, actual user input data is in the format as defined in an external HID specification. Refer to [21] for USB and as [22] for Bluetooth.

Table 22. HIDC message format

Field	Size (Octet)	Value
HID Input Path	1	HID Input Path. See Table 23.
HID Type	1	HID Type. See Table 24.
Usage	1	This field indicates the usage of the HIDC value field in this table. The value of this field shall be set to 0x00 if the HIDC value field contains a HID input report. The value of this field shall be set to 0x01 if the HIDC value field contains a HID report descriptor.
Length	2	Length of the HIDC value in octets.
HIDC value	Variable	This field contains a HID input report or a HID report descriptor.

Table 23. HID Input Path

Value	HID Input Path
0	Infrared
1	USB
2	Bluetooth
3	Zigbee
4	Wi-Fi
5-254	Reserved
255	Vendor Specific HID interface

Table 24. HID Type

Value	HID Type
0	Keyboard
1	Mouse
2	Single Touch
3	Multi Touch
4	Joystick
5	Camera
6	Gesture
7	Remote controller
8-254	Reserved
255	Vendor specific HID type

A HID report descriptor describes the format of its associated HID input reports. For each HID interface type and HID type combination, a WFD Sink should send its associated HID report descriptor to the WFD Source before it sends HID input reports to the WFD Source. The WFD Sink may send HID report descriptors to the WFD Source at multiple occasions to ensure that the WFD Source has the up-to-date HID report descriptors.

Default descriptors for USB keyboard and mouse HID input reports are specified. If the HID input reports that a WFD Sink sends are based on the default report descriptors for USB keyboard and mouse, the WFD Sink is not required to send HID report descriptors. The default USB HID report descriptor for keyboard is set to the one specified in section E.6 “Report Descriptor (keyboard)” in [21]. The default USB HID report descriptor for mouse is set to the one specified in section E.10 “Report Descriptor (Mouse)” in [21].

4.12 WFD Session and WFD Connection Termination

If a WFD Session has been established and if a WFD Device tries to terminate a WFD Connection, the WFD Device should initiate a WFD Session termination using RTSP TEARDOWN as specified in section 6.4.8.

There are two other cases of WFD Session termination without an explicit RTSP TEARDOWN message exchange.

- If a WFD Session has been established and if a WFD Device detects a timeout, the WFD Device terminates the WFD Session as specified in section 6.5.
- If a WFD Session has been established and a user turns WFD functionality in a WFD Device off, a WFD Session is terminated without any explicit signaling.

If possible, a WFD Device may terminate a WFD Connection as described in sections 4.12.1and 4.12.2 after a termination of a WFD Session.

4.12.1 Termination of a WFD Connection Using Wi-Fi P2P

When using Wi-Fi P2P, the WFD Source or the WFD Sink may perform an orderly termination of the WFD Connection using the connection tear-down procedure defined in [7].

4.12.2 Termination of a WFD Connection Using TDLS

When using TDLS either the WFD R1 Source or the WFD R1 Sink may perform an orderly termination of the WFD Connection using the connection tear-down procedure defined in [13].

If the TDLS link is torn down before a successful RTSP TEARDOWN, the WFD Source and the WFD Sink shall stop transmitting and shall discard any received frames (containing RTP/MPEG2-TS and/or RTSP message) related to the WFD Session that was ongoing over that TDLS link. Whether to attempt to re-establish the TDLS link before RTSP timeout to keep streaming, or to wait for the timeout to stop streaming, is implementation-specific and outside the scope of this specification.

4.12.3 Termination of WFD connection Using BSS infrastructure

When a WFD R2 Source and a WFD R2 Sink are in a WFD Session over an Infrastructure connection, the WFD R2 Source or the WFD R2 Sink may terminate the WFD Session but shall keep the Infrastructure connection.

4.13 Persistent WFD Groups

A Persistent WFD Group is a WFD Group for which required information is stored and may be made available for reuse after the initial use completes.

If Wi-Fi P2P is used as the WFD Connection mechanism, the required information to be stored is the P2P Group ID and Credentials. Such a Persistent WFD Group has a lifetime that may extend over a number of distinct sessions beyond the initial use until the group is deliberately “dissolved”.

If TDLS is used as the WFD Connection mechanism, the required information to be stored is the MAC address of the peer device. Such a Persistent WFD Group has a lifetime that may extend over a number of distinct sessions beyond the initial use until the group is deliberately terminated (described in section 4.13.2).

A Persistent WFD group eases the process of WFD Device Discovery and WFD Connection establishment with a peer WFD Device.

WFD R1 device may use a persistent WFD group over P2P (described in section 4.13.1) or TDLS (described in section 4.13.2).

WFD R2 device shall be capable of using a persistent WFD group over P2P (described in section 4.13.1)

4.13.1 Persistent WFD Group over Wi-Fi P2P

Persistence for WFD over Wi-Fi P2P is the same as persistence in Wi-Fi P2P as described in [7]. No additional Wi-Fi Display specific information is cached in the persistent store.

4.13.2 Persistent WFD Group over TDLS

Because the Persistent WFD Group over TDLS as defined in the Wi-Fi Display Technical Specification v1.0 has been deprecated, this section is removed. This feature is not applicable for a WFD R2 Device.

4.14 WFD MAC Procedures -- Concurrency

4.14.1 Concurrent WLAN access with Wi-Fi P2P

When using Wi-Fi P2P for WFD Sessions, the WFD Source and/or WFD Sink may also be associated with an AP. Such a WFD Device should support concurrent traffic access over the Wi-Fi BSS, with minimal disruption to the BSS and other devices on the Distribution System that it may be connected to.

A WFD Device that is concurrently associated with an AP shall conform to the procedures in [7] for concurrent access over the Wi-Fi and the P2P link.

4.14.2 Concurrent WLAN access with TDLS

When using TDLS for WFD Sessions, the WFD Source and WFD Sink are associated with the same AP or GO. Therefore, support for concurrent traffic access with an AP (or a GO) and with a peer TDLS STA is implicit, per IEEE802.11z [13].

4.15 WFD Standby

The feature in this section is optional, and the capability is indicated by RTSP M3 message exchange (and optional WFD Service Discovery).

WFD Standby mode in this specification is as a mode that a WFD Device can go into for power saving at an application layer. The exact function(s) which a WFD Device will perform while in this mode is implementation specific and is outside the scope of this specification.

WFD Sources and Sinks can go into WFD Standby mode for power saving for a number of reasons. For example:

- A notebook computer can go into WFD Standby mode due to lack of user input activities.
- A TV or monitor can go into WFD Standby mode due to lack of activity on display inputs.

In order for WFD devices to maximize potential power savings and for them to be able to wake up from WFD Standby mode in a timely manner, a WFD Source needs to be able to inform a WFD Sink that it is going into WFD Standby mode and vice versa.

A WFD Source/Sink indicates to the WFD Sink/Source that it is going into WFD Standby mode by sending a SET_PARAMETER Request message with the parameter wfd-standby (specified in section 6.1.18). A WFD Source/Sink indicates it is leaving or asking to leave WFD Standby mode by sending a SET_PARAMETER Request message with the parameter triggered-method set to PLAY (specified in section 6.4.5) and/or by sending PLAY (specified in section 6.4.7).

If the WFD Source and the WFD Sink support the standby/resume feature, upon receiving an RTSP SET_PARAMETER Request message with the wfd-standby parameter (RTSP M12 Request message) and the WFD Source sends an RTSP M12 Response message with a status code of RTSP OK, the WFD Source shall perform the following:

- Continue to maintain the WFD Session with the WFD Sink, and
- Continue to respond to RTSP commands from the WFD Sink.

In addition, the WFD Source should perform the following:

- Discontinue transmission of RTP packets that carry an MPEG2-TS to the WFD Sink.
- Go into WFD Standby mode at an application layer, e.g., shutdown MPEG-TS multiplexer and audio/video encoder.

If the WFD Source and the WFD Sink support the standby/resume feature, upon receiving an RTSP SET_PARAMETER Request message with the wfd-standby parameter (RTSP M12 message) and the WFD Sink sends an RTSP M12 Response message with a status code of RTSP OK, the WFD Sink shall perform the following:

- Continue to maintain the WFD Session with the WFD Source, and
- Continue to respond to RTSP commands from the WFD Source.

In addition, the WFD Sink may perform the following:

- Shut down display interface(s) or integrated display function.
- Go into WFD Standby mode at an application layer, e.g., shutdown MPEG2-TS demultiplexer and audio/video decoder.

To resume from WFD Standby mode, there are four cases.

First case:

If the WFD Source and the WFD Sink support this feature, when the WFD Source wants to leave WFD Standby mode, the WFD Source and the WFD Sink shall perform the following steps in order:

- the WFD Source in WFD Standby mode sends an RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to PLAY to the WFD Sink to indicate that the WFD Sink is being requested to send an RTSP M7 message. This implies that the WFD Source in WFD Standby mode is leaving WFD Standby mode.
- the WFD Sink sends an RTSP M5 Response message.
- upon a receiving the RTSP M5 Response message indicating a status code of RTSP OK, the WFD Source exits WFD Standby mode.
- after a successful completion of an RTSP M7 message exchange, the WFD Source starts transmitting RTP packets that carry an MPEG2-TS to the WFD Sink.
- the WFD Sink receives and decodes (and renders if it has integrated rendering device or outputs to externally connected rendering device) an MPEG2-TS transmitted by the WFD Source.

Second case:

If the WFD Source and the WFD Sink support the standby/resume feature and the WFD Sink is in WFD Standby mode, when the WFD Source wants to ask the WFD Sink to leave WFD Standby mode, the WFD Source and the WFD Sink shall perform following steps in order:

- the WFD Source sends an RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to PLAY to the WFD Sink to indicate that the WFD Sink is being requested to send an RTSP M7 Request message. This also implies that the WFD Source is asking the WFD Sink in WFD Standby mode to resume.
- upon receiving the RTSP M5 Request message and responding to it with an RTSP M5 Response message indicating a status code of RTSP OK, the WFD Sink exits WFD Standby mode.
- the WFD Sink enables its display interface(s) or internal display if currently disabled. (This step may be done in different order and is outside the scope of this specification.)
- the WFD Sink sends an RTSP M7 Request message after successful completion of the RTSP M5 message exchange.
- after successful completion of the RTSP M7 message exchange, the WFD Source starts transmitting RTP packets that carry an MPEG2-TS to the WFD Sink.
- the WFD Sink receives and decodes (and renders if it has an integrated rendering device or outputs to externally connected rendering device) an MPEG2-TS transmitted by the WFD Source.

Third case:

If the WFD Source and the WFD Sink support the standby/resume feature, when the WFD Sink wants to leave WFD Standby mode, the WFD Source and the WFD Sink shall perform the following steps in order:

- the WFD Sink in WFD Standby mode sends an RTSP M7 Request message to the WFD Source to indicate that the WFD Sink is requesting resumption of the audio and/or video stream. This implies that the WFD Sink in WFD Standby mode is leaving WFD Standby mode.
- the WFD Sink exits WFD Standby mode.
- the WFD Sink enables display interface(s) or internal display, if currently disabled. (This step may be done in different order and is outside the scope of this specification.)
- after successful completion of the RTSP M7 message exchange, the WFD Source starts transmitting RTP packets that carry an MPEG2-TS to the WFD Sink.
- the WFD Sink receives and decodes (and renders if it has an integrated rendering device or outputs to externally connected rendering device) an MPEG2-TS transmitted by the WFD Source.

Fourth case:

If the WFD Source and the WFD Sink support the standby/resume feature, when the WFD Sink wants to ask the WFD Source in WFD Standby mode to leave WFD Standby mode, the WFD Source and the WFD Sink shall perform following steps in order:

- the WFD Sink sends an RTSP M7 Request message to the WFD Source to indicate that the WFD Sink is requesting resumption of the audio and/or video stream. This implies that the WFD Sink is asking the WFD Source in WFD Standby mode to resume.
- upon receiving the RTSP M7 Request message, the WFD Source exits WFD Standby mode.
- after successful completion of the RTSP M7 message exchange, the WFD Source starts transmitting RTP packets that carry an MPEG2-TS to the WFD Sink.
- the WFD Sink receives and decodes (and renders if it has an integrated rendering device or outputs to externally connected rendering device) an MPEG2-TS transmitted by the WFD Source.

When the WFD Source re-starts transmitting RTP packets that carry an MPEEG2-TS to the WFD Sink as described in the above 4 cases, the WFD Source should transmit the PAT/PMT in the MPEG2-TS first. When the WFD Sink re-starts receiving audio and/or video payload in the MPEG2-TS from the WFD Source, the WFD Sink shall parse an MPEG2-TS without the PAT/PMT at the beginning of the resumed MPEG2-TS.

If both the WFD Source and the WFD Sink are in WFD Standby mode, the first case and the second case (or the third case and the fourth case) are combined, but only one RTSP M5 (or M7) Request message is transmitted to leave WFD Standby mode at both the WFD Source and the WFD Sink.

The WFD Source in WFD Standby mode shall not send any RTSP Request messages except for the RTSP M16 Request message and the RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to PLAY or to TEARDOWN.

If the WFD Source in WFD Standby mode receives an RTSP M7 Request message, an RTSP M8 Request message or an RTSP M12 Request message, the WFD Source shall respond as specified in section 6.4.7, 6.4.8 or 6.4.12, respectively. If the WFD Source in WFD Standby mode receives other RTSP Request messages, the WFD Source in WFD Standby mode shall send an RTSP Response message with status code “406” (meaning “not acceptable”) and reason phrase “in-standby-mode”.

The WFD Sink in WFD Standby mode shall not send any RTSP Request messages except for the RTSP M7 Request message and the RTSP M8 Request message.

If the WFD Sink in WFD Standby mode receives an RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to PLAY or to TEARDOWN, or an RTSP M12 Request message, the WFD Sink shall respond as specified in section 6.4.5 or 6.4.12, respectively. If the WFD Sink in WFD Standby mode receives other RTSP Request messages, the WFD Sink in WFD Standby mode shall send RTSP Response message with status code “406” (meaning “not acceptable”) and reason phrase “in-standby-mode”.

4.16 Direct Streaming and Transport of Auxiliary Streams

The features described in this section are applicable to WFD R2 devices.

4.16.1 Overview

When in non transcoding mode, a WFD R2 Source sends the media content in its native encoding format without the need of decoding and re-encoding. The audio, video and any auxiliary content obtained from the media application are first separated into the corresponding elementary streams. The generation of these elementary streams may require parsing of the bit-streams available in any other container formats of the content. The bit-streams are then packetized to their respective PES packets and multiplexed using the MPEG2-TS multiplexer. The processing of the audio and video streams is similar to WFD1.0, except that the audio and/or video data may just be passed through in their native encoding formats when streaming in non transcoding mode, and thus the encoding of the bit streams is not required at the WFD R2 Source.

When the WFD R2 Source needs to send graphical contents such as overlays or subtitles associated to the main video content, a separate elementary stream for the graphical content is generated and packetized as an auxiliary stream. The WFD R2 Source generates the transport stream by multiplexing the audio, video and auxiliary streams as shown in Figure 23.

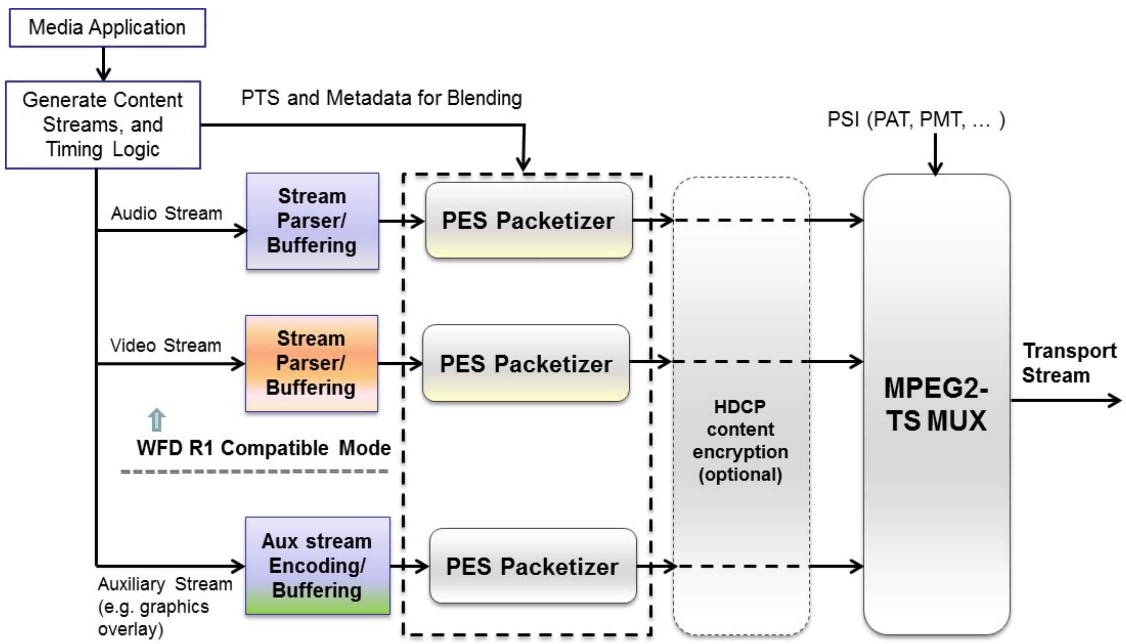


Figure 23. Overview of Audio/Video and Auxiliary Streams over MPEG2-TS

When a WFD R2 Sink, that supports auxiliary streams receives the transport stream containing an auxiliary stream from the WFD R2 Source, it blends the display content using the blending metadata included in the PES packets of the auxiliary stream as described in Appendix B.2.

4.16.2 Capability Negotiation and Session Setup

The WFD capability negotiation that uses the non-transcoding mode for streaming requires an exchange of the RTSP M3 Request and Response messages that includes the wfd2-video-formats and wfd2-audio-codecs parameter. The WFD R2 Source shall include the wfd2-video-formats and wfd2-audio-codecs parameter in the M3 Request message to query the video and audio formats supported by the WFD R2 Sink to determine if the Sink supports non-transcoding. The WFD R2 Sink shall respond with the M3 Response containing the list of supported video formats, an indication of its capability for direct streaming mode in the non-transcoding-support field in the wfd2-video-formats parameter, and the supported audio formats in the wfd2-audio-codecs parameter.

After completion of the RTSP M3 Request/Response messages with a WFD R2 Sink that has indicated support for direct streaming, the WFD R2 Source determines whether the direct steaming of video, audio or both is possible based on the list of supported audio and video formats from the Sink. It should be noted that the WFD R2 Source shall transcode the stream to compatible format if the corresponding native encoding format of the stream is not supported by the WFD R2 Sink.

Note: After the media content to be streamed is parsed and de-multiplexed to video and audio streams, it is possible that WFD R2 Source may require to use non-transcoding mode only for either the video or audio stream when the native encoding format of only one of the audio or video streams is supported by the WFD R2 Sink. In such scenario, it is left to the implementation of the WFD R2 Source to allow sending one of the audio or video streams without transcoding along with the transcoding of other associated stream or to use transcoding for both streams.

In order to start direct streaming of a media content, WFD R2 Source sends an RTSP SET_PARAMETER Request (M4 Request) message including the codec parameters to be used in the wfd2-video-formats and wfd2-audio-codecs parameter. The details for performing such AV streaming switch during a WFD Session is described in section 4.16.3.

The WFD capability negotiation to discover if auxiliary streams can carry overlay graphics involves exchanging wfd2-aux-stream-formats between the WFD R2 Source and WFD R2 Sink during the RTSP M3 Request and Response messages. The WFD R2 Source shall include wfd2-aux-stream-formats parameter in the M3 Request message, and the WFD R2 Sink shall respond with the M3 Response containing the wfd2-aux-stream-formats parameter with the list of supported auxiliary stream formats and associated information in the wfd2-aux-stream-formats parameter as described in section 6.1.23. If the WFD R2 Sink does not support the auxiliary streams, it shall respond with “none” in the wfd2-aux-stream-formats parameter when sending the M3 Response.

When both the WFD R2 Source and WFD R2 Sink support auxiliary streams, (i.e. when both support direct streaming), the WFD R2 Source may include wfd2-aux-stream-formats parameter in the RTSP M4 Request message containing the parameters to be used for the auxiliary streams during the WFD Session. The WFD R2 Source shall set the wfd2-aux-stream-formats parameter in the M4 Request to indicate the codec used for the auxiliary stream in the aux-codecs field, the maximum number of overlay layers to be included in the auxiliary stream in the max_overlay_layers field and the full-screen reference overlay resolution in the reference-overlay-resolution field (see details in Appendix B.2). The WFD R2 Source shall set the max_overlay_layers field in the M4 Request to be lower or equal to the max_overlay_layers value indicated by the WFD R2 Sink in the M3 Response. When the number of overlay layers associated to a media content is more than that can be supported by the WFD R2 Sink, the WFD R2 Source shall overlay them locally or combine some overlay components before transmitting to ensure that the auxiliary stream only includes the number of overlay layers up to the maximum value indicated in the M4 Request.

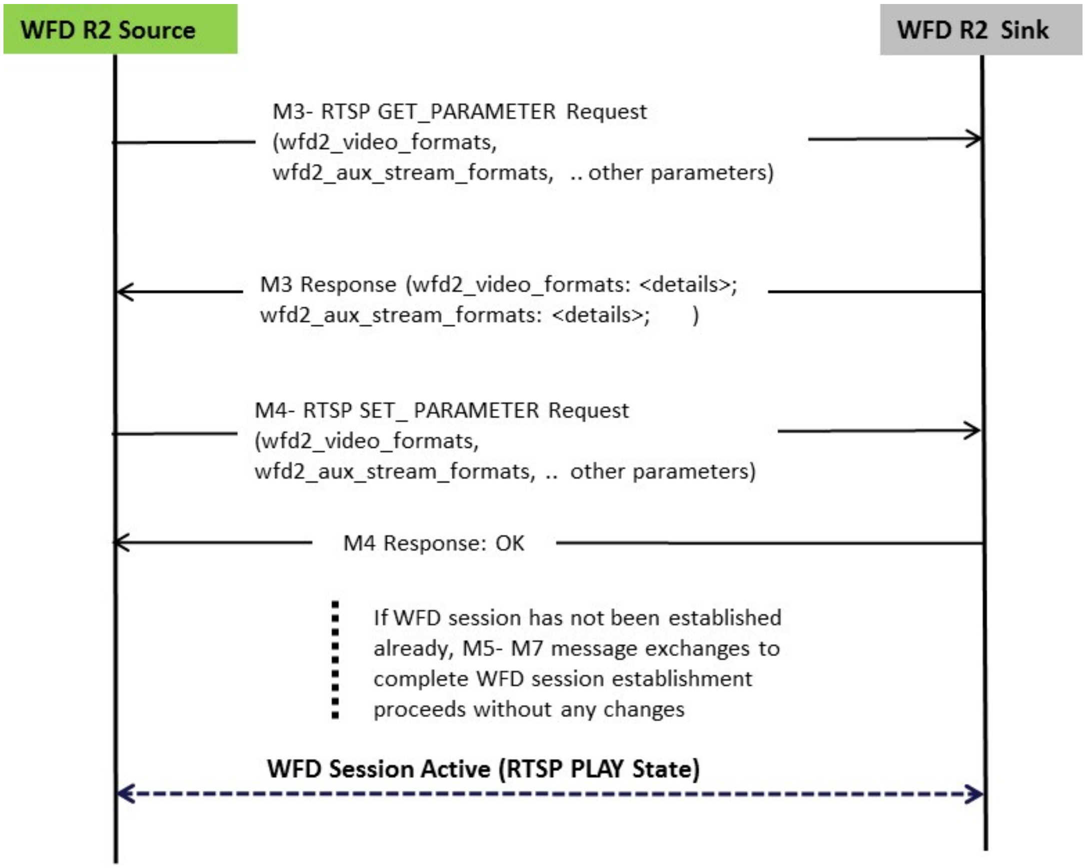


Figure 24. Message Sequence for Capability Negotiation to Setup Direct Streaming and Auxiliary Streams in a WFD Session

4.16.3 AV Streaming Switch

The WFD R2 Source and WFD R2 Sink may initiate a WFD Session by using a common A/V format supported for transcoding mode such that the display of the WFD R2 Source is mirrored to the WFD R2 Sink. When playback of a new media content is initiated, such as for a locally stored media, the WFD R2 Source may switch to stream the A/V content in its native encoding format if such mode is supported by the WFD R2 Sink. The main advantage of switching to direct streaming is to enable lower power consumption at the WFD R2 Source by avoiding the decoding and re-encoding of the content.

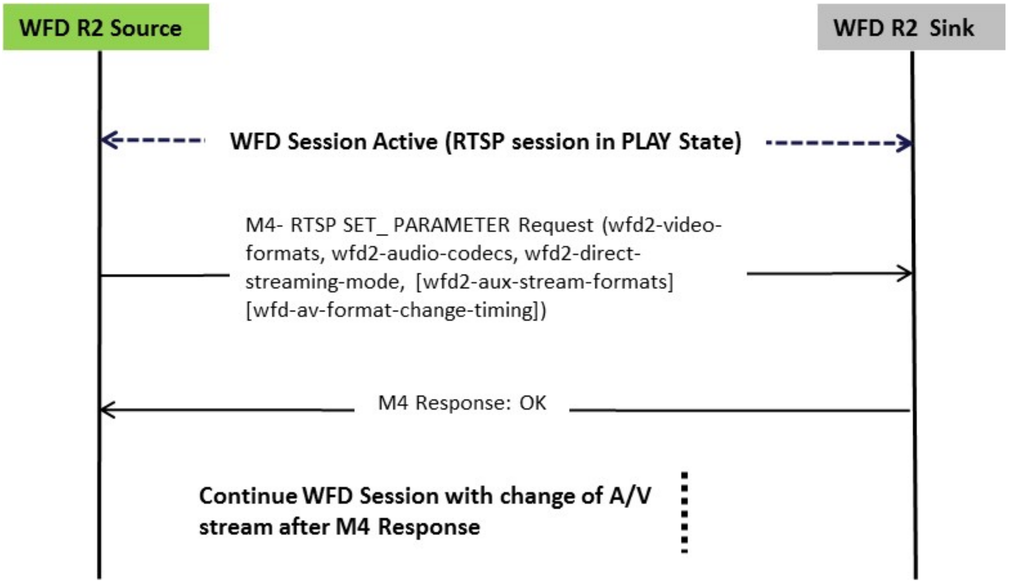


Figure 25. Message sequence for AV Streaming Switch

In order to allow the WFD R2 Sink to properly switch the decoders to receive a new bit-stream using a different format, the WFD R2 Source shall send a RTSP SET_PARAMETER message (M4 Request) that includes the audio and video codec parameters (wfd2-video-formats and wfd2-audio-codecs) to be used for the new bit-stream. In addition, the WFD R2 Source shall include the wfd2-direct-streaming-mode parameter set to “active” in the M4 Request to indicate the switch to non-transcoding (direct streaming) mode. In this mode, the WFD R2 Source shall select the codec/profile that are applicable for the non-transcoding mode for encoding of the content as listed in the Table 8. The indication to switch to direct streaming mode allows the WFD R2 Sink to handle any additional buffer resources, which otherwise may not be necessary when in the transcoding mode. The WFD R2 Source shall also include wfd-av-format-change-timing, if necessary, as described in this specification. If a WFD R2 Source needs to send auxiliary stream for any graphical overlay contents to a WFD R2 Sink capable of receiving them, it shall also provide the required codec and other settings in the wfd2-aux-stream-formats parameter.

To start switching from AV streaming to direct streaming, the WFD R2 Source shall set wfd2-video-formats parameter to include the highest display resolution, refresh rate and level that are supported by both the WFD R2 Source and the WFD R2 Sink for the selected video codec. This allows the WFD R2 Source to send the video content in the same format as it was obtained without requiring frequent explicit AV format changes. Once direct streaming has started, some of the rules for automatic AV format change(s) (without user intervention) as described in section 4.10.3.2 are not applicable. The following rules shall be satisfied within the direct streaming mode.

After switching to direct streaming mode, the resolution, refresh rate and level for the selected video codec and profile may change as indicated only in an MPEG2-TS. The WFD R2 Source is not required to update these parameter changes by performing an explicit AV format change except for the following conditions:

- If the WFD R2 Sink sets the Video Frame Rate Change Support bit (B4) of the frame-rate-control-support field in the wfd2-video-formats parameter in the RTSP M3 Response message to zero, then the WFD R2 Source shall indicate any change in video frame rate by performing explicit AV format change (i.e. by sending a RTSP M4 Request message with the appropriate parameters) with the WFD R2 Sink. In this case, after detecting a change in video frame rate of the video stream, the WFD R2 Source may switch back to transcoding mode when performing the explicit AV format change by including wfd2-direct-streaming-mode parameter set to “inactive” in the RTSP M4 Request.
- The WFD R2 Source shall explicitly indicate any change in video frame rate when the scan method of the video, i.e., progressive or interlace changes by performing explicit AV format change (i.e. by sending a RTSP M4 Request message with the appropriate parameters) with the WFD R2 Sink. In this case, after detecting a change in scan method for the video stream, the WFD R2 Source may switch back to transcoding mode when performing the explicit AV format change by including wfd2-direct-streaming-mode parameter set to “inactive” in the RTSP M4 Request.

When in direct streaming mode, the WFD R2 Source shall perform explicit AV format change by sending an RTSP M4 Request message containing the wfd2-video-formats, wfd-av-format-change-timing and other associated parameters (if applicable as described in section 4.16.2) to the WFD R2 Sink if the codec or profile of the video stream is required to be changed (e.g. to start streaming another media content). When in direct streaming mode, the WFD R2 Source shall indicate any changes in audio format (i.e., sample rate, bits per sample, the number of channels, audio codec) by sending a M4 Request message containing the wfd2-audio-codecs parameter to the WFD R2 Sink. Same rule applies if the accompanying audio stream has been sent by transcoding or without transcoding.

When in direct streaming mode, the WFD R2 Sink should not send an RTSP M13 Request (SET_PARAMETER with wfd-idr-request) to the WFD R2 Source for requesting an IDR refresh. If the direct streaming mode is active and the WFD R2 Source receives an IDR Request (RTSP M13 Request) from a WFD R2 Sink, the WFD R2 Source may respond with status code 503 (service unavailable) in the RTSP M13 Response to indicate that it is not possible to send an IDR picture at that time. In this case, the WFD R2 Source may take additional implementation specific recovery action such as switching back to transcoding mode as necessary to address future IDR requests during streaming of the media content.

After switching to the direct streaming mode, if the WFD R2 Source wants to switch back to transcoding mode, it shall perform explicit AV format change with the WFD R2 Sink by sending an RTSP M4 Request message that shall include wfd2-direct-streaming-mode parameter set to “inactive”, wfd-av-format-change-timing parameter to indicate the timing of AV format change and the audio and video codec parameters (wfd2-video-formats and wfd2-audio-codecs) to be used after the switch.

After receiving the RTSP SET_PARAMETER message (M4 Request) from the WFD R2 Source indicating a switch to the direct streaming mode, the WFD R2 Sink shall send the RTSP OK (M4 Response) to the WFD R2 Source when it is ready to receive the new bit-stream with the indicated AV format. Once the WFD R2 Source receives the RTSP OK Response (M4 Response) from the WFD R2 Sink, it shall start AV streaming using the MPEG2 TS stream to encapsulate media payload in native format from the new AV media content.

If the WFD R2 Source intends to switch to direct streaming mode but the AV content to be transmitted is subject to Link Content Protection requirements, the WFD R2 Source and WFD R2 Sink shall complete the HDCP 2.x session key establishment prior of performing the AV streaming switch, using the procedure as described in section 4.7. When the auxiliary content associated to the video is subject to Link Content Protection requirements, the WFD R2 Source shall encode the auxiliary content using H.264 CBP and then encrypt the corresponding PES payload for the auxiliary stream using the procedure described in section 4.7. When the auxiliary stream associated with the protected video content is not subject to Link Content Protection requirements, the WFD R2 Source may encode the auxiliary content using any codec type listed in Table 10 that is supported by both the WFD R2 Source and WFD R2 Sink and transmit the auxiliary stream without HDCP encryption.

When the AV media content being streamed reaches end of the stream or a user stops the playback of the media content, the WFD R2 Source may again perform an AV stream switch by sending RTSP SET_PARAMETER message (M4 Request) to set the Audio/video format to be used for the mirroring mode.

4.16.4 Auxiliary Stream over MPEG2-TS

If the media application in the WFD R2 Source requires overlaid graphical content to the video being played, it shall extract the overlay content bit-stream as a separate auxiliary elementary stream to send over the MPEG2-TS. The WFD R2 Source shall only include the image up to highest negotiated display resolution that can be supported by the WFD R2 Sink in an auxiliary stream.

Upon receiving the auxiliary content stream, the WFD R2 Sink shall blend the display image using the in-band metadata information contained in the private header of the PES packet as described in Appendix B.2. The WFD R2 Sink composites the graphics image received from the auxiliary stream to the video plane to generate the display output.

4.17 Backward Compatibility

4.17.1 Device/Service Discovery

- A WFD R2 Device shall support P2P device discovery procedures using WFD IE (for R1and R2) and WFD R2 discovery procedures based on mDNS/DNS-SD for infrastructure connection.
- A WFD R2 Device may optionally support TDLS procedures as specified in section 5.2.2.1 (Tunneled Probe Request/Response).
- A WFD R2 Device shall detect the version of peer device (R1 or R2) during discovery procedures.

- If a WFD R2 Device receives a P2P Probe Request with the WFD Device Information subelement only in the WFD IE without the WFD R2 Device Information subelement , the WFD R2 Device shall assume that the peer device is a WFD R1 Device and shall include the WFD Device Information subelement in the a WFD IE but may not include the WFD R2 Device Information subelement in its Probe Response frame.
- If a WFD R2 Device receives a P2P Probe Request the WFD R2 Device Information subelement in the WFD IE over P2P link, the WFD R2 Device shall assume that the peer Device is WFD R2 Device and shall include the WFD R2 Device Information subelement in the WFD IE in its Probe Response frame.
- If a WFD R2 Device discovers another WFD Device over infrastructure using mDNS it shall assume that the peer device is a WFD R2 Device.
- If a WFD R2 Device supports TDLS and receives a Tunneled Probe Request with WFDIE it shall assume the peer device is WFD R1 Device and shall include a WFD IE with the WFD Device Information subelementin its Tunneled Probe Response Frame.

4.17.2 WFD Connection Establishment

A WFD R2 Device shall support mandatory P2P connection procedure for WFD connection with a WFD R1 Device as specified in section 4.5.2.

A WFD R2 Device may optionally support TDLS connection procedures with a WFD R1 Device as specified in section 4.5.3.

4.17.3 WFD Capability Negotiation

A WFD R2 Device shall support all mandatory Capability Negotiation parameters (RTSP protocol) as specified in section 4.9.5 to interoperate with WFD R1 Device.

A WFD R2 Device may support optional Capability Negotiation parameters (RTSP protocol) as specified in section 4.9.5 to interoperate with a WFD R1 Device.

4.17.4 WFD Session Establishment

A WFD R2 Device shall support session establishment procedures as specified in section 4.8 to interoperate with a WFD R1 Device.

5 Frame formats and service discovery information

This section describes the information elements and frame formats used to perform the procedures described in Chapter 4. In addition, the Service Discovery Information used by WFD R2 devices to perform discovery of Display R2 Service are also described.

The WFD communication protocol is based on the use of the WFD Information Element (WFD IE) and WFD action frame formats. These utilize the Vendor Specific Information Element and Vendor Specific Action frame formats as specified in IEEE Std 802.11-2007 [14] with the Wi-Fi Alliance OUI and OUI Type indicating Wi-Fi Display. A number of WFD subelements are defined; a single WFD IE carries one or more WFD subelements.

Byte ordering within the multi-octet fields shall be in network byte order (big-endian).

5.1 WFD Information Element

5.1.1 WFD IE Format

The vendor specific information element format (as defined in IEEE Std 802.11-2007 [14]) is used to define the WFD information element (WFD IE) in this specification. The format of the WFD IE is shown in Table 25.

Table 25. WFD IE Format

Field	Size (octets)	Value (Hexadecimal)	Description
Element ID	1	DD	IEEE 802.11 vendor specific usage
Length	1	Variable	Length of the following fields in the IE in octets. The length field is variable and set to 4 plus the total length of WFD subelements.
OUI	3	50-6F-9A	Wi-Fi Alliance Specific OUI
OUI Type	1	0A	Identifying the type or version of the WFD IE. Setting to 0x0A indicates Wi-Fi Alliance WFD v1.0
WFD subelements	Variable		One or more WFD subelements appear in the WFD IE

The WFD subelements are defined to have a common general format consisting of a 1 octet WFD subelement ID field, a 2 octets Length field and variable-length subelement specific information fields as shown in Table 26.

Table 26. General Format of a WFD Subelement

Field	Size (octets)	Value (Hexadecimal)	Description
Subelement ID	1		Identifying the type of WFD subelement. The specific value is defined in Table 27.
Length	2	Variable	Length of the following fields in the subelement
Subelements body field	Variable		Subelement specific information fields

Table 27. WFD Subelement ID Definitions

Subelement ID (Decimal)	Notes
0	WFD Device Information
1	Associated BSSID
2-5	Reserved
6	Coupled Sink Information

Subelement ID (Decimal)	Notes
7	WFD Extended Capability
8	Local IP Address
9	WFD Session Information
10	Alternative MAC Address
11	WFD R2 Device Information
12-255	Reserved

A WFD Device that encounters an unknown or reserved subelement ID value within a WFD IE which was received without error shall ignore that WFD subelement and parse any remaining fields for additional WFD subelements with recognizable subelement ID values. A WFD Device that encounters a recognizable but unexpected subelement ID value in the received WFD IE may ignore that WFD subelement.

More than one WFD IE may be included in a single frame. If multiple WFD IEs are present, the complete WFD subelement data consists of the concatenation of the WFD subelement fields of the WFD IEs. The WFD subelements field of each WFD IE may be any length up to the maximum (251 octets). The order of the concatenated WFD subelement data shall be preserved in the ordering of the WFD IEs in the frame. All of the WFD IEs shall fit within a single frame and shall be adjacent in the frame. If a WFD subelement is not contained entirely within a single WFD IE, the WFD subelement ID field and Length field for that subelement occur only once at the start. This is the same rule for P2P IE as specified in section 4.1.1 of [7], and an example for P2P IE is shown in Figure 16 of [7].

5.1.2 WFD Device Information Subelement

The WFD Device Information subelement is used to signal information required by a peer WFD Device during discovery to facilitate a decision as to whether to attempt pairing with the peer WFD Device and creating a WFD Session. The format of the WFD Device Information subelement is as shown in Table 28.

Table 28. WFD Device Information Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	0	Identifying the type of WFD subelement. The specific value is defined in Table 27.
Length	2	6	Length of the following fields of the subelement.
WFD Device Information	2		Bitmap defined in Table 29 detailing WFD Device Information.
Session Management Control Port	2	Valid TCP port	Default 7236 [30]. TCP port at which the WFD Device listens for RTSP messages. (If a WFD Sink that is transmitting this subelement does not support the RTSP server function, this field is set to all zeros.) The WFD Device can choose any value other than default 7236[1]
WFD Device Maximum Throughput	2		Maximum average throughput capability of the WFD Device represented in multiples of 1Mbps
Notes. 1. When not choosing default 7236, it is recommended to choose a port number within 49152 to 65535 (as the Private or Ephemeral Ports as described in [29].			

Table 29 lists the interpretation of the WFD Device Information field.

Table 29. WFD Device Information Field

Bits	Name	Interpretation
1:0	WFD Device Type bits	0b00: WFD Source 0b01: Primary Sink 0b10: Secondary Sink 0b11: dual-role possible, i.e., either a WFD Source or a Primary Sink
2	Coupled Sink Operation Support at WFD Source bit	0b0: Coupled Sink Operation not supported by WFD Source. 0b1: Coupled Sink Operation supported by WFD Source This bit is valid for WFD Device Type bits set to value 0b00 or 0b11. When WFD Device Type bits value is 0b01 or 0b10, the value of this b2 is ignored upon receiving.
3	Coupled Sink Operation Support at WFD Sink bit	0b0: Coupled Sink Operation not supported by WFD Sink 0b1: Coupled Sink Operation supported by WFD Sink This bit is valid for WFD Device Type bits set to value 0b01, 0b10 or 0b11. When WFD Device Type bits value is 0b00, the value of this b3 is ignored upon receiving.
5:4	WFD Session Availability bits	0b00: Not available for WFD Session 0b01: Available for WFD Session 0b10, 0b11: Reserved
6	WSD Support bit	0b0: WFD Service Discovery (WSD): Not supported 0b1: WFD Service Discovery (WSD): Supported
7	PC bit	0b0: Preferred Connectivity (PC): P2P 0b1: Preferred Connectivity (PC): TDLS
8	CP Support bit	0b0: Content Protection using the HDCP system 2.x: Not supported 0b1:Content Protection using the HDCP system 2.x Supported[1]
9	Time Synchronization Support bit	0b0: Time Synchronization using 802.1AS: Not supported 0b1: Time Synchronization using 802.1AS: Supported
10	Audio un-supported at Primary Sink bit	0b0: all cases except below 0b1: If B1B0=0b01 or 0b11, and this WFD Device does not support audio rendering when acting as a Primary Sink
11	Audio only support at WFD Source bit	0b0: all cases except below 0b1: If B1B0=0b00 or 0b11, and this WFD Device supports transmitting audio only elementary stream when acting as a WFD Source
12	TDLS Persistent Group bit	0b0: TDLS persistent group not intended 0b1: TDLS persistent group intended
13	TDLS Persistent Group Re-invoke	0b0: all other cases except below 0b1: The request is for re-invocation of TDLS persistent group
15:14	Reserved	Set to all zeros

Notes.

1. Because a transition period from HDCP 2.0 to HDCP 2.1 defined by DCP LLC has been expired, this 0b1 means “Content Protection using the HDCP system 2.1 Supported”, on the WFD devices seeking WFD certification. After a transition period from the previous version of HDCP 2 to the newest version of HDCP 2 defined by DCP LLC has been expired, this 0b1 means “Content Protection using the newest version of HDCP system 2 Supported”, on the WFD devices seeking WFD certification.

WFD devices which intend or accept establishment of a WFD Connection or the Coupling operation shall set WFD Device Type bits (B5B4) of the WFD Device Information field of the WFD Device Information subelement to 0b01.

5.1.3 Associated BSSID Subelement

The Associated BSSID subelement is used to indicate the address of the AP (or the GO) that the WFD Device is associated with. The format of the Associated BSSID subelement is as shown in Table 30.

If a WFD Device is associated with more than one AP(s) and/or GO(s), the WFD Device shall select one of them to be indicated in this Associated BSSID subelement. Although it is recommended to select one that can be used in TDLS topology, selection is outside the scope of this specification.

Table 30. Associated BSSID Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	1	Identifying the type of WFD subelement. The specific value is defined in Table 27.
Length	2	6	Length of the following fields of the subelement.
Associated BSSID	6		Address of the infrastructure AP or the GO to which the WFD Device is associated.

5.1.4 Coupled Sink Information Subelement

The Coupled Sink Information subelement signals the status of a WFD Sink's coupling with another WFD Sink. The format of the Coupled Sink Information subelement is as shown in Table 31.

Table 31. Coupled Sink Information Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	6	Identifying the type of WFD subelement. The specific value is defined in Table 27.
Length	2	7	Length of the following fields of the subelement.
Coupled Sink Status bitmap	1		Bitmap defined in Table 32 detailing Coupled Sink Status.
Coupled Sink MAC Address	6		MAC address of other WFD Sink with which Coupling established. If the WFD Device has not Coupled, these six octets shall be set to all zeros.

5.1.4.1 Coupled Sink Status Bitmap

The Coupled Sink status bitmap signals the status of a WFD Sink's coupling with another WFD Sink. This bitmap is also used by the wfd-coupled-sink parameter described in section 6.1.7.

Table 32 defines the fields of the Coupled Sink Status Bitmap.

Table 32. Coupled Sink Status Bitmap

Bits	Name	Interpretation
1:0	Coupled Sink Status bits	0b00: Not coupled/Available for Coupling 0b01: Coupled 0b10: Teardown Coupling 0b11: Reserved
7:2	Reserved	Set to all zeros

5.1.5 WFD Video Formats Subelement

The WFD Video Formats subelement is no longer defined. However, because some fields of the WFD Video Formats subelement are referred to by other sections, the description of the fields is shown in Table 33.

Table 33. WFD Video Formats Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	N/A	No subelement ID is assigned
Length	2	21	Length of the following fields of the subelement.
CEA Resolutions/Refresh Rates bitmap	4		Bitmap defined in Table 34 detailing CEA resolutions and refresh rates supported by the CODEC.
VESA Resolutions/Refresh Rates bitmap	4		Bitmap defined in Table 35 detailing VESA resolutions and refresh rates supported by the CODEC.
HH Resolutions/Refresh Rates bitmap	4		Bitmap defined in Table 36 detailing HH resolutions and refresh rates supported by the CODEC.
Native Resolutions/Refresh Rates bitmap	1		Bitmap defined in Table 37 detailing native display resolutions and refresh rates supported by the WFD Sink.
Profiles bitmap	1		Bitmap defined in Table 38 detailing the H.264 profile indicated by this instance of the WFD Video Formats subelement.
Levels bitmap	1		Bitmap defined in Table 39 detailing the H.264 level indicated by this instance of the WFD Video Formats subelement.
Latency field	1		Specifying the latency of the video decoder at the Primary Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD Source in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD Source in an RTSP M4 Request message and the WFD Sink shall ignore this field upon reception. If the Primary Sink does not support this field, it shall set this field to all zeros. Otherwise the WFD Sink shall set this field to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.
Minimum slice size field	2		Specifying the smallest slice size expressed in number of macroblocks. If this field is transmitted by the WFD Source, this value shall be the smallest encoded slice it can support. If this field is transmitted by the Primary Sink, this value shall be the smallest slice size it can decode. WFD devices that do not support slice encoding in which a picture is constructed by multiple slices shall set this field to 0x00 00.
Slice Encoding Parameters bitmap	2		Bitmap defined in Table 40 detailing parameters for the slice encoding in which a picture is constructed by multiple slices.
Video Frame Rate Control Support bitmap	1		Bitmap defined in Table 41 detailing the video frame skipping support with its parameter and frame rate change support.

5.1.5.1 CEA Resolutions/Refresh Rates Bitmap

The CEA Resolutions/Refresh rates bitmap represents the set of CEA resolutions and corresponding refresh rates that a WFD Device supports. This bitmap is also used by the wfd-video-formats parameter (specified in section 6.1.3).

In the wfd-video-formats parameter that is included in RTSP M3 Responses or in WFD Service Discovery frames:

- B0 of the Supported CEA resolution/refresh-rates bitmap (Table 34) shall be set to one for all WFD devices (except for Secondary Sink) indicating all WFD devices shall support 640x480p60 as a mandatory mode of operation.
- The WFD devices that support higher resolution(s) at 60Hz family than 640x480 shall set B1 to one, and the WFD devices that support higher resolution(s) at 50Hz family than 640x480 shall set B3 to one, of Table 34.

The frame rate described in Table 34, Table 35 and Table 36 refer to the rate at which video payload is sourced from the WFD Source. The transport stream may include information that is used to determine if the video payload corresponds to

60 (59.94), 30 (29.97), 50, 25, or 24 (23.98) fps as defined in [2]. The WFD Source selects a frame rate to be used for the WFD Session. The selected frame rate is conveyed to the WFD Sink during WFD Capability Negotiation using the RTSP M4 Request message. The decoder at the WFD Sink shall have support for 59.94, 29.97, or 23.98 fps when the WFD Sink advertised it support of 60, 30, or 24 fps, respectively.

See section 3.4.2 for complete details of mandatory configurations.

Index is used for the Native Resolution/Refresh Rate bitmap.

Table 34. Supported CEA Resolution/Refresh Rates

Bits	Index	Interpretation
0	0	640x480 p60
1	1	720x480 p60
2	2	720x480 i60
3	3	720x576 p50
4	4	720x576 i50
5	5	1280x720 p30
6	6	1280x720 p60
7	7	1920x1080 p30
8	8	1920x1080 p60
9	9	1920x1080 i60
10	10	1280x720 p25
11	11	1280x720 p50
12	12	1920x1080 p25
13	13	1920x1080 p50
14	14	1920x1080 i50
15	15	1280x720 p24
16	16	1920x1080 p24
31:17	-	Reserved

5.1.5.2 VESA Resolutions/Refresh Rates Bitmap

The VESA Resolutions/Refresh Rates bitmap given in Table 35 represents the set of VESA resolutions and corresponding refresh rates that a WFD Device supports. This bitmap is also used by the wfd-video-formats parameter (specified in section 6.1.3).

When specifying support for VESA format, the WFD Sink shall indicate support for a resolution with higher refresh rate(s) if and only if it also indicates support for a corresponding lower refresh rate. For instance, support for 720x480p60 can be indicated only if support for 720x480p30 is also indicated.

Index is used for the Native Resolution/Refresh Rate bitmap.

Table 35. Supported VESA Resolution/Refresh Rates

Bits	Index	Interpretation
0	0	800x600 p30
1	1	800x600 p60

Bits	Index	Interpretation
2	2	1024x768 p30
3	3	1024x768 p60
4	4	1152x864 p30
5	5	1152x864 p60
6	6	1280x768 p30
7	7	1280x768 p60
8	8	1280x800 p30
9	9	1280x800 p60
10	10	1360x768 p30
11	11	1360x768 p60
12	12	1366x768 p30
13	13	1366x768 p60
14	14	1280x1024 p30
15	15	1280x1024 p60
16	16	1400x1050 p30
17	17	1400x1050 p60
18	18	1440x900 p30
19	19	1440x900 p60
20	20	1600x900 p30
21	21	1600x900 p60
22	22	1600x1200 p30
23	23	1600x1200 p60
24	24	1680x1024 p30
25	25	1680x1024 p60
26	26	1680x1050 p30
27	27	1680x1050 p60
28	28	1920x1200 p30
31:29	-	Reserved

5.1.5.3 HH Resolutions/Refresh Rates Bitmap

The HH Resolutions/Refresh Rates bitmap given in Table 36 represents the set of resolutions and corresponding refresh rates commonly supported in handheld devices that a WFD Device supports. This bitmap is also used by the wfd-video-formats parameter (specified in section 6.1.3).

Index is used for the Native Resolution/Refresh Rate bitmap.

Table 36. Supported HH Resolutions/Refresh Rates

Bits	Index	Interpretation
0	0	800x480 p30

Bits	Index	Interpretation
1	1	800x480 p60
2	2	854x480 p30
3	3	854x480 p60
4	4	864x480 p30
5	5	864x480 p60
6	6	640x360 p30
7	7	640x360 p60
8	8	960x540 p30
9	9	960x540 p60
10	10	848x480 p30
11	11	848x480 p60
31:12	-	Reserved

5.1.5.4 Native Resolutions/Refresh Rates Bitmap

The Native Resolution/Refresh Rate bitmap given in Table 37 represents the native resolutions and corresponding refresh rates of a WFD Device. This bitmap is used in WFD Video Formats subelement. This bitmap is also used by the wfd-video-formats parameter (specified in section 6.1.3) and the wfd-3d-formats parameter (specified in section 6.1.4). If this bitmap is included in the WFD Video Formats subelement transmitted by a WFD Source, it shall be set to all zeros. If this bitmap is included in an RTSP M4 Request message, a WFD Source should set this bitmap to all zeros and a Primary Sink shall ignore this bitmap upon reception.

Table 37. Display Native Resolution Refresh Rate

Bits	Name	Interpretation
2:0	Table Selection bits	0b000: Resolution/Refresh rate table selection: Index to CEA resolution/refresh rates (Table 34) 0b001: Resolution/Refresh rate table selection: Index VESA resolution/refresh rates (Table 35) 0b010: Resolution/Refresh rate table selection: Index HH resolutions/refresh rates (Table 36) 0b011~0b111: Reserved
7:3	Index bits	Index into resolution/refresh rate table selected by [B2:B0]

5.1.5.5 Profiles Bitmap

The Profiles Bitmap given in Table 38 represents the H.264 profiles supported by a WFD Device. This bitmap is also used in the WFD 3D Video Formats subelement. The Profiles Bitmap when included in a WFD subelement shall either have B0 or B1 (but not both) of the bitmap set to one

This bitmap is also used by the wfd-video-formats parameter (specified in section 6.1.3), the wfd-3d-formats parameter (specified in section 6.1.4) and the wfd-preferred-display-mode parameter (specified in section 6.1.14).

Table 38. Profiles Bitmap

Bits	Name	Interpretation
0	CBP bit	0b0: Constrained Baseline Profile (CBP) not supported 0b1: CBP supported
1	CHP bit	0b0: Constrained High Profile (CHP) not supported

Bits	Name	Interpretation
		0b1: CHP supported
7:2	Reserved	Set to all zeros

5.1.5.6 Levels Bitmap

The Levels bitmap given in Table 39 indicates the maximum H.264 level supported for the corresponding H.264 profile indicated in the Profiles Bitmap. Only one bit in the Levels Bitmap used in a WFD subelement shall be set to one. This bitmap is also used in the WFD 3D Video Formats subelement.

This bitmap is also used by the wfd-video-formats parameter (specified in section 6.1.3), the wfd-3d-formats parameter (specified in section 6.1.4) and the wfd-preferred-display-mode parameter (specified in section 6.1.14). In this case, the bitmap represents either:

- the maximum level supported for the H.264 profile indicated in the Profiles Bitmap supported by the WFD Device in an RTSP M3 Response message (only one bit set to one) or
- the level and the corresponding H.264 profile selected by the WFD Source in an RTSP M4 Request message (only one bit set to one).

Table 39. Maximum H.264 Level Supported

Bits	Name	Interpretation
0	H.264 Level 3.1 bit	0b0: H.264 Level 3.1 not supported 0b1: H.264 Level 3.1 supported
1	H.264 Level 3.2 bit	0b0: H.264 Level 3.2 not supported 0b1: H.264 Level 3.2 supported
2	H.264 Level 4 bit	0b0: H.264 Level 4 not supported 0b1: H.264 Level 4 supported
3	H.264 Level 4.1 bit	0b0: H.264 Level 4.1 not supported 0b1: H.264 Level 4.1 supported
4	H.264 Level 4.2 bit	0b0: H.264 Level 4.2 not supported 0b1: H.264 Level 4.2 supported
7:5	Reserved	Set to all zeros

5.1.5.7 Slice Encoding Parameters Bitmap

The slice encoding parameters bitmap given in Table 40 is two octets long and describes the parameters to be used in slice encoding in which a picture is constructed by multiple slices. This bitmap is also used in the WFD 3D Video Formats subelement. This bitmap is also used by the wfd-video-formats parameter (specified in section 6.1.3), the wfd-3d-formats parameter (specified in section 6.1.4) and the wfd-preferred-display-mode parameter (specified in section 6.1.14).

Table 40. Slice Encoding parameters bitmap

Bits	Name	Interpretation
9:0	Max Slice Num bits	Maximum number of slices per a picture, minus 1.
12:10	Max Slice Size Ratio bits	When this bitmap is used in a WFD subelement: Ratio of Maximum slice size to be used and Minimum slice size indicated in minimum-slice-size field in WFD Video Formats or WFD 3D Video Formats subelement. When this bitmap is used in an RTSP message: Ratio of Maximum slice size to be used and Minimum slice size indicated in minimum-slice-size field in wfd-video-formats or wfd-3d-formats.
15:13	Reserved	Set to all zeros

If this bitmap is used in a subelement and the Minimum slice size field in the WFD Video Formats subelement or the WFD 3D Video Formats subelement is all zeros, all bits in this bitmap shall be set to zero.

If this bitmap is used in an RTSP message and the minimum-slice-size field in the wfd-video-formats parameter or the wfd-3d-formats parameter is all zeros, all bits in this bitmap shall be set to zero.

The subfields [B9:B0] and [B12:B10] shall be set to a non-zero value in other cases.

5.1.5.8 Video Frame Rate Control Support Bitmap

The Video Frame Rate Control Support bitmap given in Table 41 indicates support for Frame Rate Change and support for Video Frame Skipping. If Video Frame Skipping is supported, the maximum time intervals in unit of 0.5 seconds that can be elapsed between two successive video frames are also indicated.

This bitmap is also used in the WFD 3D Video Formats subelement. This bitmap is also used by the wfd-video-formats parameter (specified in section 6.1.3), the wfd-3d-formats parameter (specified in section 6.1.4) and the wfd-preferred-display-mode parameter (specified in section 6.1.14).

Table 41. Video Frame Rate Control Support Bitmap

Bits	Name	Interpretation
0	Video Frame Skipping Support bit	0b0: Not supported 0b1: Supported
3:1	Max Skip Interval bits	Reserved if B0 is 0b0 (video frame skipping not supported) These bits indicate the maximum/allowable time-interval between two video frames after skipped as expressed equation as follows, except for 0b000. (value in decimal) x 0.5 second(s) 0b000 : No limitation 0b001~ 0b111 : parameter for indicating time-interval
4	Video Frame Rate Change (see section 4.10.3.2) Support bit	0b0: Dynamic video refresh rate change without user intervention not supported 0b1: Dynamic video refresh rate change without user intervention supported
7:5	Reserved	Set to all zeros

5.1.6 WFD 3D Video Formats Subelement

The WFD 3D Video Formats subelement is no longer defined.

5.1.7 WFD Audio Formats Subelement

The WFD Audio Formats subelement is no longer defined. However, because some fields of the WFD Audio Formats subelement are referred to by other sections, the description of the fields are shown in Table 42.

Table 42. WFD Audio Formats Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	N/A	No subelement ID is assigned
Length	2	15	Length of the following fields of the subelement.
LPCM Modes bitmap	4		Bitmap defined in Table 43 detailing LPCM audio formats supported by the codec on the WFD Device.

Field	Size (octets)	Value	Description
LPCM decoder latency field	1		Specifying the latency of the LPCM decoder at the WFD Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD Source in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD Source in an RTSP M4 Request message and the WFD Sink shall ignore this field upon reception. If the WFD Sink does not support this field, it shall set this field to all zeros. Otherwise the WFD Sink shall set this field to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.
AAC Modes bitmap	4		Bitmap defined in Table 44 detailing AAC audio formats supported by the codec on the WFD Device.
AAC decoder latency field	1		Specifying the latency of the AAC decoder at the WFD Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD Source in a WFD Service Discovery Response frame. This field shall be set to all zeros when transmitted by the WFD Sink that does not support AAC format at all, in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD Source in an RTSP M4 Request message and the WFD Sink shall ignore this field upon reception. If the WFD Sink does not support this field, it shall set this field to all zeros. Otherwise the WFD Sink shall set this field to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.
AC3 Modes bitmap	4		Bitmap defined in Table 45 detailing AC3 audio formats supported by the codec on the WFD Device.
AC3 decoder latency field	1		Specifying the latency of the AC3 decoder at the WFD Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD Source in a WFD Service Discovery Response frame. This field shall be set to all zeros when transmitted by the WFD Sink that does not support AC3 format at all, in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD Source in an RTSP M4 Request message and the WFD Sink shall ignore this field upon reception. If the WFD Sink does not support this field, it shall set this field to all zeros. Otherwise the WFD Sink shall set this field to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.

5.1.7.1 LPCM Modes Bitmap

The LPCM Modes bitmap represents LPCM configurations supported by the WFD Device. This bitmap is also used in the 'modes' field of wfd-audio-codecs as described in section 6.1.2.

In a wfd-audio-codecs parameter that is included in an RTSP M3 Response message or in a WFD Service Discovery frame, B1 of the LPCM Modes bitmap (Table 43) shall be set to one for all WFD devices to indicate support of 2-channel LPCM audio at 16 bits/channel at 48000 samples/second as a mandatory mode of operation (except for a Primary Sink that does not have audio rendering capability, e.g., typical office projector. Note that a device is deemed audio rendering capable if it can playback audio payload with or without the help of an external transducer, e.g., attached speaker). Other LPCM audio formats are optional at all WFD devices.

Table 43. LPCM Modes bitmap

Bits	Interpretation		
	Sampling Frequency (kHz)	Nominal bit width at decoder output (bits)	#channels
0	44.1	16	2
1	48	16	2

Bits	Interpretation		
	Sampling Frequency (kHz)	Nominal bit width at decoder output (bits)	#channels
31:2	Reserved		

5.1.7.2 AAC Modes Bitmap

The AAC Modes bitmap given in Table 44represents AAC configurations supported by the WFD Device. This bitmap is also used in the 'modes' field of wfd-audio-codecs as described in section 6.1.2.

Table 44. AAC Codec bitmap

Bits	Interpretation			
	Sampling Frequency (kHz)	Nominal bit width at decoder output (bits)	#channels	Codec Option
0	48	16	2	AAC-LC
1	48	16	4	AAC-LC
2	48	16	6	AAC-LC
3	48	16	8[1]	AAC-LC
31:4	Reserved			
Notes: 1. In ISO/IEC standard, down-mix method is not defined for 8-ch (7.1ch), and it is recommended that the WFD Sink that does not support 8-ch (7.1ch) natively should not set this bit to one.				

5.1.7.3 AC3 Modes Bitmap

The AC3 Modes bitmap given in Table 45 represents Dolby Digital (also known as AC3) configurations supported by the WFD Device. This bitmap is also used in the 'modes' field of wfd-audio-codecs as described in section 6.1.2.

Table 45. AC3 Modes bitmap

Bits	Interpretation			
	Sampling Frequency (kHz)	Nominal bit width at decoder output (bits)	#channels	Codec Option
0	48	16	2	Dolby Digital (AC-3)
1	48	16	4	Dolby Digital (AC-3)
2	48	16	6	Dolby Digital (AC-3)
31:3	Reserved			

5.1.8 WFD Extended Capability Subelement

The WFD Extended Capability subelement is used to indicate the capabilities for miscellaneous and optional functions of a WFD Device. The format of the WFD Extended Capability subelement is as shown in Table 46.

Table 46. WFD Extended Capability Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	7	Identifying the type of WFD subelement. The specific value is defined in Table 27.
Length	2	2	Length of the following fields of the subelement.

Field	Size (octets)	Value	Description
WFD Extended capabilities bitmap	2		Bitmap defined in Table 47 detailing capabilities for miscellaneous and optional functions

5.1.8.1 WFD Extended Capabilities Bitmap

The WFD Extended Capabilities bitmap given in Table 47 describes support for UIBC, I2C Read/Write, Preferred Display mode and WFD Standby/resume control by a WFD Device.

Table 47. WFD Extended Capabilities Bitmap

Bits	Name	Interpretation
0	UIBC Support bit	0b0: Not supported 0b1: Supported
1	I2C Read/Write Support bit	0b0: Not supported 0b1: Supported
2	Preferred Display mode Support bit	0b0: Not supported 0b1: Supported
3	Standby and Resume Control Support bit	0b0: Not supported 0b1: Supported
4	TDLS Persistent Support bit	0b0: Not supported 0b1: Supported
5	TDLS Persistent BSSID Support bit	0b0: Not supported 0b1: Supported
15:6	Reserved	Set to all zeros

5.1.9 Local IP Address Subelement

The Local IP Address subelement is used to convey the local IP address to a WFD peer STA during TDLS Setup. Only if a WFD IE is included in a TDLS Setup Request/Response as described in section 4.5.3, the WFD IE shall include Local IP Address subelement. Table 48 depicts the format of this subelement.

Table 48. Local IP Address Subelement

Field	Size (octets)	Value	Description
Sub-element ID	1	8	Defined in Table 27.
Length	2	5	Length of the following fields in this subelement
Version	1	1	Version 1: IPv4 address field follows
IPv4 address	4		This field is the IPv4 host address of the STA

5.1.10 WFD Session Information Subelement

The WFD Session Information subelement describes WFD Session information. Table 49 depicts the format of this subelement. A WFD Device that is acting as a GO based on P2P connection includes a list of descriptors of all WFD devices which are associated in its group and indicate that P2P Discoverability bit in Device Capability Bitmap of P2P Capability attribute equals to 0b1. WFD Device Info Descriptor field is shown in Table 50.

Table 49. WFD Session Information Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	9	Identifying the type of WFD subelement. The specific value is defined in Table 27.
Length	2	Variable	Length of the following fields of the subelement.
WFD Device Info Descriptor	Sum of all WFD Device info (24octets * number of clients)		List of WFD Device Info Descriptor in WFD group

The WFD Device Info Descriptor includes information corresponding to each WFD Device in the group. The WFD Device that is acting as a GO based on P2P connection already knows about all its clients, and it can advertise the known information for other clients of the group to other WFD devices that are not part of the group. Note that this subelement shall not include GO itself. When the GO does not have associated client that is WFD capable, this subelement shall not be included in the WFD IE.

Table 50. WFD Device Info Descriptor field

Field name	Size (octets)	Value	Description
Length	1	23	Length of the following fields.
Device address	6		Device address
Associated BSSID	6		Address of the infrastructure AP or the GO to which the WFD Device is associated. If the WFD Device described in this descriptor is not associated with the infrastructure AP, these 6 octets shall be set to all zeros. If the WFD devices associated with more than one AP(s) and/or GO(s), the WFD Device shall choose one of them to be indicated in this Associated BSSID subelement. How to choose one is out-of-scope of this specification, but it is recommended to choose one that can be used in TDLS topology.
WFD Device Information	2		Bitmap defined in Table 29 detailing WFD Device Information.
WFD Device Maximum Throughput	2		Maximum average throughput capability of the WFD Device represented in multiples of 1Mbps
Coupled Sink Information	7		Coupling Status and address of the Coupled Primary or Secondary Sink to which the WFD devices is Coupled or ready to couple. The first one byte is Coupled Sink status bitmap as defined in Table 31, and the rest six bytes are MAC address of Coupled peer WFD Sink. If the WFD Device described in this descriptor is not Coupled, these seven octets shall be set to all zeros. If the GO does not know the MAC address of the Coupled sink with that the WFD Sink described in this descriptor, the last six octets shall be set to all zeros.

The GO may set WFD Session Availability bits in the WFD Device Information field of WFD Device Info Descriptor to 0b00 or 0b01. The recipient shall interpret these bits as WFD Session availability “unknown”.

5.1.11 Alternative MAC Address Subelement

If the resolved connectivity scheme demands use of an interface different from the one used during device discovery phase, the Alternate MAC Address subelement is used to communicate the alternate P2P device address or the WLAN interface address to be used for the WFD Connection. The MAC address required for forthcoming connection establishment is then included into this subelement as an Alternative MAC address field by the WFD Device while transmitting (tunneled) Probe Response frame.

The format of the Alternate MAC Address subelement is as shown in Table 51

Table 51. Alternative MAC Address Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	10	Identifying the type of WFD subelement. The specific value is defined in Table 27.
Length	2	6	Length of the following fields of the subelement.
Alternative MAC address	6	Valid MAC address	Address of WLAN interface or P2P Device address.

5.1.12 WFD R2 Device Information Subelement

The WFD R2 Device Information subelement is used to signal information required by a peer WFD Device during Discovery to decide whether to pair with the peer WFD R2 Device and create a WFD Session. The format of the WFD R2 Device Information subelement is shown in Table 52.

Table 52. WFD R2 Device Information Subelement

Field	Size (octets)	Value	Description
Subelement ID	1	11	Identifying the type of WFD subelement. The specific value is defined in Table 27.
Length	2	variable	Length of the following fields of the subelement.
WFD R2 Device Information	2		Bitmap defined in Table 53 detailing WFD R2 Device Information.

Table 53 lists the interpretation of the WFD R2 Device Information field.

Table 53. WFD R2 Device Information

Bits	Name	Interpretation
1:0	WFD R2 Device Type bits	0b00: WFD R2 Source 0b01: WFD R2 Primary Sink 0b10: Reserved (to be ignored upon reception) 0b11: dual-role possible, i.e., either a WFD R2 Source or a WFD R2 Primary Sink
2:15	Reserved	To be ignored upon reception

All WFD R2 devices shall include the WFD R2 Device Information subelement in the WFD IE with the WFD R2 Device Type indicated by B1-B0 bits. All WFD R2 devices shall continue including WFD Device Information subelement to indicate its session management control port information and attributes in the WFD Device Information bitmap that are also applicable for WFD R2 devices (i.e. WFD Session Availability bits, CP Support bit, Time Synchronization Support bit, Audio un-supported at Primary Sink bit and Audio only support at WFD Source bit).

5.2 Management Frames

This section defines extensions to the P2P management frames specified in [7] in support of WFD Capabilities. A WFD R2 Device shall include the WFD IE in Beacon, Probe Request/Response, Association Request/Response and P2P Public Action frames in order to be interoperable with R1 devices. If a WFD R2 Device discovers that the peer device is also a WFD R2 Device, then it may include the WFD IE in Association Request/Response and P2P Public Action frames.

5.2.1 Beacon Frame Format

If a WFD Device acts as a P2P Group Owner, the WFD Device shall insert one or more WFD IEs after the other information elements in the Beacon frames it transmits. WFD subelements for a WFD IE that are included in the Beacon frame are shown in Table 54.

Table 54. WFD Subelements in a Beacon Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the Beacon frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the Beacon frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation and if the WFD Sink is the P2P GO, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the Beacon frames it transmits.

5.2.2 Probe Request Frame Format

The Probe Request frames are transmitted by any WFD Device.

A WFD Device shall insert one or more WFD IEs after the other information elements in the Probe Request frames it transmits. WFD subelements for a WFD IE that are included in the Probe Request frame are shown in the Table 55.

Table 55. WFD Subelements in a Probe Request Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the Probe Request frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the Probe Request frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the Probe Request frames it transmits.
WFD Extended Capability	7	If a WFD Device intends to advertise its TDLS persistent capability to other WFD devices during the discovery process, the WFD Device shall include the WFD Extended Capability subelement in the WFD IE in the Probe Request frames it transmits.
WFD R2 Device Information	11	A WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the Probe Request frames it transmits.

5.2.2.1 Tunneled Probe Request/Response

Probe Request frames and Response frames listed in Table 56 are tunneled via

1. The AP to which the WFD Source and Sink are associated if the TDLS Prohibited field of the Extended Capability element included in the Beacon frame from the AP is not set to one, or
2. The GO to which the WFD Source and Sink are associated if the Intra-BSS Distribution field of the Group Capability bitmap of the P2P Capability attribute in the beacon frame from the GO is not set to zero.

Table 56. Tunneled Probe Request/Response

Field in the Frame	Value (Hexadecimal)	Length (octets)	Requirement
802.11 MAC Header		Depends on the data frame	MAC header corresponding to a data frame

Field in the Frame	Value (Hexadecimal)	Length (octets)	Requirement
LLC	AA-AA-03	3	
SNAP	00-00-00-89-0D	5	
Payload Type	2	1	TDLS
Category	7F	1	Vendor Specific
OUI	50-6F-9A	3	Wi-Fi Alliance OUI
Frame Body Type	4 or 5	1	4 for Probe Request, 5 for Probe Response; other values are reserved. Corresponds to the Management Frame subtype as defined in [13]
Other fields		Variable	Information Elements that are included in the encapsulated Probe Request/Response payload

5.2.3 Probe Response Frame

The Probe Response frames are transmitted by any WFD Device as per the rules specified in [7].

When a WFD Device responds to the Probe Request frame that is transmitted by other WFD Device and contains a WFD IE, the WFD Device shall insert one or more WFD IEs after the other information elements in the Probe Response frames it transmits. WFD subelements for a WFD IE that are included in the Probe Response frame are shown in Table 57

Table 57. WFD Subelements in a Probe Response Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the Probe Response frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the Probe Response frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the Probe Response frames it transmits.
WFD Extended Capability	7	If a WFD Device intends to advertise its TDLS persistent capability to other WFD devices during the discovery process, the WFD Device shall include the WFD Extended Capability subelement in the WFD IE in the Probe Response frames it transmits.
WFD Session Information	9	If a WFD Capable GO has at least one associated client that is WFD capable, the WFD capable GO shall include the WFD Session Information subelement in the WFD IE in the Probe Response frames it transmits.
Alternative MAC Address	10	If a WFD Device intends to use a different interface for the forthcoming WFD Connection from the one on which the (tunneled) Probe Request Frame was received, the WFD Device shall include the Alternate MAC Address subelement in the WFD IE in the (tunneled) Probe Response frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device receives a Probe Request frame containing the WFD R2 Device Information subelement in the WFD IE, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the Probe Response frames it transmits.

5.2.4 Association/Reassociation Request Frame

When a WFD Device becomes a P2P client, an Association/Reassociation Request frame is transmitted by the WFD Device.

When a WFD Device tries to establish a P2P connection for a WFD Session with another WFD Device, the WFD Device shall insert one or more WFD IEs after the other information elements in the Association/Reassociation Request frames it transmits. WFD subelements for a WFD IE that is included in the Association/Reassociation Request frame are shown in Table 58.

Table 58. WFD Subelements in an Association/Reassociation Request Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the Association/Reassociation Request frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the Association/Reassociation Request frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the Association/Reassociation Request frames it transmits.
WFD R2 Device Information	11	A WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the Association/Reassociation Request frames it transmits to a peer WFD R2 Device.

5.2.5 Association/Reassociation Response Frame

Upon receiving an Association/Reassociation Request frame, an Association/Reassociation Response frame is transmitted by a WFD Device acting as a P2P GO.

When a WFD Device responds to the Association/Reassociation Request frame that is transmitted by other WFD Device and contains a WFD IE, the WFD Device shall insert one or more WFD IEs after the other information elements in the Association/Reassociation Response frames it transmits. WFD subelements for a WFD IE that are included in the Association/Reassociation Response frame are shown in Table 59.

Table 59. WFD Subelements in an Association/Reassociation Response Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement shall be present in the WFD IE in the Association/Reassociation frames it transmit.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the Association/Reassociation Response frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the Association/Reassociation Response frames it transmits.
WFD Session Information	9	If a WFD Capable GO has at least one associated client that is WFD capable, the WFD capable GO shall include the WFD Session Information subelement in the WFD IE in the Association/Reassociation Response frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device receives an Association/Reassociation Request frame containing the WFD R2 Device Information subelement in the WFD IE, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the Association/Reassociation Response frames it transmits to the peer WFD R2 Device.

5.2.6 P2P Public Action Frames

5.2.6.1 GO Negotiation Request Frame

The GO Negotiation Request frames are transmitted by any WFD Device.

When a WFD Device tries to establish a P2P connection to initiate a WFD Session with another WFD Device, the WFD Device shall insert one or more WFD IEs after the other information elements in the GO Negotiation Request frames it transmits. WFD subelements for a WFD IE that are included in the GO Negotiation Request frame are shown in Table 60.

Table 60. WFD Subelements in a GO Negotiation Request Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the GO Negotiation Request frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the GO Negotiation Request frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the GO Negotiation Request frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device transmits a GO Negotiation Request frame addressed to a peer WFD R2 Device, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the GO Negotiation Request frames it transmits.

5.2.6.2 GO Negotiation Response Frame

Upon receiving the GO Negotiation Request frame, The GO Negotiation Response frame is transmitted by a WFD Device.

When a WFD Device responds to the GO Negotiation Request frame that is transmitted by another WFD Device and contains a WFD IE, the WFD Device shall insert one or more WFD IEs after the other information elements in the GO Negotiation Response frame that it transmits. WFD subelements for a WFD IE that are included in the GO Negotiation Response frame are shown in Table 61.

Table 61. WFD Subelements in a GO Negotiation Response Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the GO Negotiation Response frame.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the GO Negotiation Response frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the GO Negotiation Response frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device receives a GO Negotiation Request frame containing the WFD R2 Device Information subelement in the WFD IE, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the GO Negotiation Response frames it transmits.

5.2.6.3 GO Negotiation Confirmation Frame

Upon receiving the GO Negotiation Response frame, the GO Negotiation Confirmation frame is transmitted by a WFD Device acting as a P2P GO.

When a WFD Device responds to the GO Negotiation Response frame that is transmitted by another WFD Device and contains a WFD IE, the WFD Device shall insert one or more WFD IEs after the other information elements in the GO Negotiation Confirmation frame that it transmits. WFD subelements for a WFD IE that are included in the GO Negotiation Response frame are shown in Table 62.

Table 62. WFD Subelements in a GO Negotiation Confirmation Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the GO Negotiation Confirmation frames it transmits.

Subelements	Subelement ID	Requirement
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the GO Negotiation Confirmation frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the GO Negotiation Confirmation frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device receives a GO Negotiation Response frame containing the WFD R2 Device Information subelement in the WFD IE, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the GO Negotiation Confirmation frames it transmits.

5.2.6.4 P2P Invitation Request Frame

The P2P Invitation Request frames are transmitted by any WFD Device.

When a WFD Device tries to establish P2P connection for a WFD Session with another WFD Device, the WFD Device shall insert one or more WFD IEs after the other information elements in the P2P Invitation Request frames it transmits. WFD subelements for a WFD IE that are included in the P2P Invitation Request frame are shown in Table 63

Table 63. WFD Subelements in a P2P Invitation Request Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the P2P Invitation Request frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the P2P Invitation Request frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the P2P Invitation Request frames it transmits.
WFD Session Information	9	If a WFD Capable GO has at least one associated client that is WFD capable, the WFD capable GO shall include the WFD Session Information subelement in the WFD IE in the P2P Invitation Request frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device transmits a P2P Invitation Request frame addressed to a peer WFD R2 Device, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the P2P Invitation Request frames it transmits.

5.2.6.5 P2P Invitation Response Frame

Upon receiving the P2P Invitation Request frame, the P2P Invitation Response frame is transmitted by a WFD Device.

When a WFD Device responds to the P2P Invitation Request frame that is transmitted by another WFD Device and contains a WFD IE, the WFD Device shall insert one or more WFD IEs after the other information elements in the P2P Invitation Response frame that it transmits. WFD subelements for a WFD IE that are included in the P2P Invitation Response frame are shown in Table 64.

Table 64. WFD Subelements in a P2P Invitation Response Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement shall be present in the WFD IE in the P2P Invitation Response frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the

Subelements	Subelement ID	Requirement
		Associated BSSID subelement in the WFD IE in the P2P Invitation Response frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the P2P Invitation Response frames it transmits.
WFD Session Information	9	If a WFD Capable GO has at least one associated client that is WFD capable, the WFD capable GO shall include the WFD Session Information subelement in the WFD IE in the P2P Invitation Response frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device receives a P2P Invitation Request frame containing the WFD R2 Device Information subelement in the WFD IE, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the P2P Invitation Response frames it transmits.

5.2.6.6 Provision Discovery Request Frame

The Provision Discovery Request frames are transmitted by a WFD Device, to establish P2P connection for WFD Session with another WFD Device, using Wi-Fi Simple Configuration.

When a WFD Device tries to establish P2P connection for a WFD Session with another WFD Device using Wi-Fi Simple Configuration, the WFD Device shall insert one or more WFD IEs after the other information elements in the Provision Discovery Request frames it transmits. WFD subelements for a WFD IE that are included in the Provision Discovery Request frame are shown in Table 65.

Table 65. WFD Subelements in a Provision Discovery Request Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the Provision Discovery Request frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID subelement in the WFD IE in the Provision Discovery Request frames it transmits.
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the Provision Discovery Request frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device transmits a Provision Discovery Request frame addressed to a peer WFD R2 Device, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the Provision Discovery Request frames it transmits.

5.2.6.7 Provision Discovery Response Frame

Upon receiving the Provision Discovery Request frames, the Provision Discovery Response frames are transmitted by a WFD Device.

When a WFD Device responds to the Provision Discovery Request frame that is transmitted by another WFD Device and contains a WFD IE, the WFD Device shall insert one or more WFD IEs after the other information elements in the Provision Discovery Response frames it transmits. WFD subelements for a WFD IE that are included in the Provision Discovery Response frame are shown in Table 66

Table 66. WFD Subelements in a Provision Discovery Response Frame

Subelements	Subelement ID	Requirement
WFD Device Information	0	A WFD Device shall include the WFD Device Information subelement in the WFD IE in the Provision Discovery Response frames it transmits.
Associated BSSID	1	If a WFD Device is associated with an infrastructure AP or a GO and the WFD Device sets its PC bit to 0b1 in WFD Device Information subelement, the WFD Device shall include the Associated BSSID in the WFD IE in the Provision Discovery Response frames it transmits.

Subelements	Subelement ID	Requirement
Coupled Sink Information	6	If a WFD Sink supports the Coupled Sink Operation, the WFD Sink shall include the Coupled Sink Information subelement in the WFD IE in the Provision Discovery Response frames it transmits.
WFD Session Information	9	If a WFD Capable GO has at least one associated client that is WFD capable, the WFD capable GO shall include the WFD Session Information subelement in the WFD IE in the Provision Discovery Response frames it transmits.
WFD R2 Device Information	11	If a WFD R2 Device receives a Provision Discovery Request frame containing the WFD R2 Device Information subelement in the WFD IE, the WFD R2 Device shall include the WFD R2 Device Information subelement in the WFD IE in the Provision Discovery Response frames it transmits.

5.2.7 Service Discovery Action Frames

5.2.7.1 WFD Service Discovery

This feature is not applicable for a WFD R2 Device.

5.3 WFD R2 Command

A WFD R2 Device may send a WFD R2 Command to other WFD R2 devices using a TCP connection, but note that the TCP connection is not for RTSP. The TCP payload structure of a WFD R2 Command is defined in Table 67.

Table 67. WFD R2 Command

Field	Size (in octet)	Description
Type	1	Type field indicates a type of the WFD R2 Command Type field definition is provided in Table 68
Length	1	Length field indicates the length in octet of Value field in this WFD R2 Command. Possible values of this field are from 0 to 255.
Value	Variable as indicated in Length field	Value field indicates content depending on Type field.

A Type of a WFD R2 Command is defined in Table 68.

Table 68. WFD R2 TCP Connection Request Command

Type field value	Description
0	TCP Connection Request Command. See section 4.5.4.
1~255	Reserved. A WFD R2 Device shall not use these reserved values in transmission and shall ignore upon reception.

5.4 WFD R2 Service Discovery Information

The capability string to be used by a WFD R2 Device may include the capability information listed in Table 69.

Table 69. WFD R2 Capability Definitions used for Service Discovery Information

Capability ID (Decimal)	Capability String	Notes
0	Display Device Information	WFD Device Information, as described in 5.1.2
1	Display Associated BSSID	Associated BSSID, as described in 5.1.3
2-5		Reserved

Capability ID (Decimal)	Capability String	Notes
6	Display Coupled Sink Information	Coupled Sink Information, as described in 5.1.4
7	Display Extended Capability	WFD Extended Capability, as described in 5.1.8
8	Display Local IP Address	Local IP Address, as described in 5.1.9
9	Display Session Information	WFD Session Information, as described in 5.1.10
10	Display Alternative MAC Address	Alternative MAC Address, as described in 5.1.11
11	Display R2 Audio Formats	WFD R2 Audio Formats, as described in 5.4.3
12	Display R2 Video Formats	WFD R2 Video Formats, as described in 5.4.1
13	Display R2 Aux Stream Formats	WFD R2 Aux Stream Formats, as described in 5.4.2
14-255		Reserved

5.4.1 WFD R2 Video Formats

The WFD R2 Video Formats indicates the video formats supported by the WFD R2 Device. The display_capability string for Service Discovery may include the WFD R2 Video Formats information. The format of the WFD R2 Video Formats information is shown in Table 70.

Table 70. WFD R2 Video Formats Information

Field	Size (octets)	Value	Description
Capability ID	1	12	Identifying the type of WFD Capability. The specific value is defined in Table 69.
Length	2	Variable	Length of the following fields listed in this information table.
WFD R2 CEA Resolutions/Refresh Rates bitmap	6		Bitmap defined in Table 71 detailing CEA resolutions and refresh rates supported by the codec.
WFD R2 VESA Resolutions/Refresh Rates bitmap	6		Bitmap defined in Table 72 detailing VESA resolutions and refresh rates supported by the codec.
WFD R2 HH Resolutions/Refresh Rates bitmap	6		Bitmap defined in Table 73 detailing HH resolutions and refresh rates supported by the codec.
WFD R2 Native Resolutions/Refresh Rates bitmap	1		Bitmap defined in Table 74 detailing native display resolutions and refresh rates supported by the WFD Sink.
WFD R2 Video Codec Info Descriptor	Sum of all supported Video Codec Info (5 octets x number of video codecs)		List of WFD R2 Video Codec Info Descriptor fields as specified in Table 75. Only one Descriptor field entry shall be included for each of the Video Codec Type supported.
Minimum slice size field	2		Specifying the smallest slice size expressed in number of macroblocks. If this field is transmitted by the WFD Source, this value shall be the smallest encoded slice it can support. If this field is transmitted by the Primary Sink, this value shall be the smallest slice size it can decode. WFD devices that do not support slice encoding in which a picture is constructed by multiple slices shall set this field to 0x00 00.
Slice Encoding Parameters bitmap	2		Bitmap defined in Table 40 detailing parameters for the slice encoding in which a picture is constructed by multiple slices.

Field	Size (octets)	Value	Description
Video Frame Rate Control Support bitmap	1		Bitmap defined in Table 41 detailing the video frame skipping support with its parameter and frame rate change support.

5.4.1.1 WFD R2 CEA Resolutions/Refresh Rates Bitmap

The WFD R2 CEA Resolutions/Refresh rates bitmap represents the set of CEA resolutions and corresponding refresh rates that a WFD R2 Device supports. This bitmap is also used by the wfd2-video-formats parameter (specified in section 6.1.22).

In the wfd2-video-formats parameter that is included in RTSP M3 Responses or in WFD Service Discovery frames:

- B0 and B15 of the Supported CEA resolution/refresh-rates bitmap (Table 71) shall be set to one for all WFD R2 devices (except for Secondary Sink) indicating all WFD R2 devices shall support 640x480p60 and 1280x720p30 as a mandatory modes of operation.
- WFD R2 Devices that support higher resolution(s) at 60Hz family than 640x480 shall set B1 to one, and WFD R2 devices that support higher resolution(s) at 50Hz family than 640x480 shall set B3 to one, see Table 5 49.

The frame rate described in Table 71, Table 72 and Table 73 refer to the rate at which video payload is sourced from the WFD R2 Source. The transport stream may include information that is used to determine if the video payload corresponds to 60 (59.94), 30 (29.97), 50, 25, or 24 (23.98) fps as defined in [2]. The WFD R2 Source selects a frame rate to be used for the WFD Session. The selected frame rate is conveyed to the WFD R2 Sink during WFD Capability Negotiation using the RTSP M4 Request message. The decoder at the WFD R2 Sink shall have support for 59.94, 29.97, or 23.98 fps when the WFD R2 Sink advertised it support of 60, 30, or 24 fps, respectively.

See section 3.4.2 for complete details of mandatory configurations.

Index is used for the Native Resolution/Refresh Rate bitmap.

Table 71. Supported CEA Resolution/Refresh Rates

Bits	Index	Interpretation
0	0	640x480 p60
1	1	720x480 p60
2	2	720x480 i60
3	3	720x576 p50
4	4	720x576 i50
5	5	1280x720 p30
6	6	1280x720 p60
7	7	1920x1080 p30
8	8	1920x1080 p60
9	9	1920x1080 i60
10	10	1280x720 p25
11	11	1280x720 p50
12	12	1920x1080 p25
13	13	1920x1080 p50
14	14	1920x1080 i50
15	15	1280x720 p24
16	16	1920x1080 p24

Bits	Index	Interpretation
17	17	3840x2160 p24
18	18	3840x2160 p25
19	19	3840x2160 p30
20	20	3840x2160 p50
21	21	3840x2160 p60
22	22	4096x2160 p24
23	23	4096x2160 p25
24	24	4096x2160 p30
25	25	4096x2160 p50
26	26	4096x2160 p60
47:27	-	Reserved

5.4.1.2 WFD R2 VESA Resolutions/Refresh Rates Bitmap

The WFD R2 VESA Resolutions/Refresh Rates bitmap given in Table 72 represents the set of VESA resolutions and corresponding refresh rates that a WFD R2 Device supports. This bitmap is also used by the wfd2-video-formats parameter (specified in section 6.1.22).

When specifying support for VESA format, the WFD R2 Sink shall indicate support for a resolution with higher refresh rate(s) if and only if it also indicates support for a corresponding lower refresh rate. For instance, support for 720x480p60 can be indicated only if support for 720x480p30 is also indicated.

Index is used for the Native Resolution/Refresh Rate bitmap.

Table 72. Supported VESA Resolution/Refresh Rates

Bits	Index	Interpretation
0	0	800x600 p30
1	1	800x600 p60
2	2	1024x768 p30
3	3	1024x768 p60
4	4	1152x864 p30
5	5	1152x864 p60
6	6	1280x768 p30
7	7	1280x768 p60
8	8	1280x800 p30
9	9	1280x800 p60
10	10	1360x768 p30
11	11	1360x768 p60
12	12	1366x768 p30
13	13	1366x768 p60
14	14	1280x1024 p30
15	15	1280x1024 p60

Bits	Index	Interpretation
16	16	1400x1050 p30
17	17	1400x1050 p60
18	18	1440x900 p30
19	19	1440x900 p60
20	20	1600x900 p30
21	21	1600x900 p60
22	22	1600x1200 p30
23	23	1600x1200 p60
24	24	1680x1024 p30
25	25	1680x1024 p60
26	26	1680x1050 p30
27	27	1680x1050 p60
28	28	1920x1200 p30
29	29	1920x1200 p60
30	30	2560x1440 p30
31	31	2560x1440 p60
32	32	2560x1600 p30
33	33	2560x1600 p60
47:34		Reserved
Notes: 1. Entries from Table 34 (CEA) are not duplicated in Table 35 (VESA). In order to use a CEA entry for a VESA monitor, apply VESA CVT timings instead of CEA timings. 2. All entries use DMT timings if defined in VESA DMT, otherwise use the CVT computed value for timings as specified in [38].		

5.4.1.3 WFD R2 HH Resolutions/Refresh Rates Bitmap

The WFD R2 HH Resolutions/Refresh Rates bitmap given in Table 73 represents the set of resolutions and corresponding refresh rates commonly supported in handheld devices that a WFD R2 Device supports. This bitmap is also used by the wfd2-video-formats parameter (specified in section 6.1.22).

Index is used for the WFD R2 Native Resolution/Refresh Rate bitmap.

Table 73. Supported HH Resolutions/Refresh Rates

Bits	Index	Interpretation
0	0	800x480 p30
1	1	800x480 p60
2	2	854x480 p30
3	3	854x480 p60
4	4	864x480 p30
5	5	864x480 p60
6	6	640x360 p30

Bits	Index	Interpretation
7	7	640x360 p60
8	8	960x540 p30
9	9	960x540 p60
10	10	848x480 p30
11	11	848x480 p60
47:12	-	Reserved

5.4.1.4 WFD R2 Native Resolutions/Refresh Rates Bitmap

The WFD R2 Native Resolution/Refresh Rate bitmap given in Table 74 represents the native resolutions and corresponding refresh rates of a WFD R2 Device. This bitmap is used in WFD R2 Video Formats information. This bitmap is also used by the wfd2-video-formats parameter (specified in section 6.1.22). If this bitmap is included in the WFD R2 Video Formats Information transmitted by a WFD R2 Source, it shall be set to all zeros. If this bitmap is included in an RTSP M4 Request message, a WFD R2 Source should set this bitmap to all zeros and a Primary Sink shall ignore this bitmap upon reception.

Table 74. Display Native Resolution Refresh Rate

Bits	Name	Interpretation
1:0	Table Selection bits	0b00: Resolution/Refresh rate table selection: Index to CEA resolution/refresh rates (Table 34) 0b01: Resolution/Refresh rate table selection: Index VESA resolution/refresh rates (Table 35) 0b10: Resolution/Refresh rate table selection: Index HH resolutions/refresh rates (Table 36) 0b11: Reserved
7:2	Index bits	Index into resolution/refresh rate table selected by [B1:B0], values 47-63 are Reserved.

5.4.1.5 WFD R2 Video Codec Info Descriptor

The WFD R2 Video Codec Info Descriptor field includes format information for the optional Video Codecs that a WFD R2 Device supports. The format for the WFD R2 Video Codec Info Descriptor field is specified in Table 75

Table 75. WFD Video Codec Info Descriptor field

Field	Size (octets)	Value	Description
Video Codec Type	1		Identification of the video codec for which the WFD Video Codec Info Descriptor is provided. The specific values are defined in Table 76.
Profiles bitmap	1		Bitmap defined in Table 77 detailing the profiles supported for the Video Codec indicated in the Video Codec Type field of this Descriptor.
Levels bitmap	2		Bitmap defined in Table 78 detailing the levels supported for the Video Codec indicated in the Video Codec Type field of this Descriptor.
Latency field	1		Specifies the latency of the video decoder at the Primary Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD Source in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD Source in an RTSP M4 Request message and the WFD Sink shall ignore this field upon reception. If the Primary Sink does not support this field, it shall set this field to all zeros. Otherwise the WFD Sink shall set this field to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.

Table 76. Video Codec Type field values in WFD R2 Video Codec Info Descriptor

Bits	Codec name	Interpretation
0	H.264	0b0: not supported 0b1: supported
1	H.265	0b0: not supported 0b1: supported
7-2	Reserved	Set to all zeros

The Profiles Bitmap given in Table 77 represents the video profiles for the specified video codec type supported by a WFD R2 Device.

This bitmap is also used by the wfd2-video-formats parameter (specified in section 6.1.22).

Table 77. Video Codec Profile Bitmap values in WFD Video Codec Info Descriptor

Bitmap applicable for the video codec type (set as per Table 76)	Bits	Profile name	Interpretation
0 (H.264)	0	Constrained Baseline Profile (CBP)	0b0: not supported 0b1: supported
	1	Restricted High Profile (CHP)	0b0: not supported 0b1: supported
	2	Restricted Notes: The RHP (formerly known as CHP) is specified by Wi-Fi Alliance in this specification with some restrictions to standard H.264 high profile as listed in Table 7. High Profile 2 (RHP2)	0b0: not supported 0b1: supported
	3	Baseline Profile (BP)	0b0: not supported 0b1: supported
	4	Main Profile (MP)	0b0: not supported 0b1: supported
	5	High Profile (HiP)	0b0: not supported 0b1: supported
	6	Progressive High Profile	0b0: not supported 0b1: supported
	7	Progressive High Still Picture	0b0: not supported 0b1: supported
1 (H.265)	0	Main Profile	0b0: not supported 0b1: supported
	1	Main 10 Profile	0b0: not supported 0b1: supported
	2	Main 444 Profile	0b0: not supported 0b1: supported
	3	Main Still Picture Profile	0b0: not supported 0b1: supported
	4	Screen Content Coding Profile	0b0: not supported

Bitmap applicable for the video codec type (set as per Table 76)	Bits	Profile name	Interpretation
			0b1: supported
	5	Main 444 10 Profile	0b0: not supported 0b1: supported
	7:6	Reserved	Set to all zeros
Notes: 1. When the R2 Primary Sink device provides the codecs/profiles it supports in the wfd2-video-formats parameter in the M3 Response, it shall include all the codecs/profiles it is capable of decoding. The WFD R2 Source shall interpret the supported format as applicable for transcoding or non-transcoding mode based on its capability to encode the content, as per the Table 7 and Table 8. 2. When the WFD R2 Source device includes the codec/profile in the wfd2-video-formats parameter in the M4 Request during establishment of the WFD Session or anytime during an active WFD Session, it shall select the codec/profile based on the transcoding or non-transcoding mode for encoding of the content and set it according to the list in Table 7 and Table 8.			

The Levels bitmap given in Table 78 indicates the maximum level supported for the corresponding video codec profile indicated in the Profiles Bitmap. Only one bit in the Levels bitmap used in a WFD R2 Video Formats information shall be set to one. The maximum supported display resolution should map to the maximum level supported. If any mismatch occurs between the maximum display resolution indicated by the maximum level and the bit map carrying the supported display resolutions (CEA, VESA or HH) supported by the Sink, the Source may select the lower value of the display resolution derived from these two parameters.

The Levels bitmap is also used by the wfd2-video-formats parameter (specified in section 6.1.22).

Table 78. Video Codec Level Bitmap values in WFD R2 Video Codec Info Descriptor

Video Codec Type Value	Bits	Profile name	Interpretation
0 (H.264)	0	Level 3.1	0b0: not supported 0b1: supported
	1	Level 3.2	0b0: not supported 0b1: supported
	2	Level 4	0b0: not supported 0b1: supported
	3	Level 4.1	0b0: not supported 0b1: supported
	4	Level 4.2	0b0: not supported 0b1: supported
	5	Level 5	0b0: not supported 0b1: supported
	6	Level 5.1	0b0: not supported 0b1: supported
	7	Level 5.2	0b0: not supported 0b1: supported
	15:8	Reserved	Set to all zeros
1 (H.265)	0	Level 3.1	0b0: not supported 0b1: supported
	1	Level 4	0b0: not supported 0b1: supported
	2	Level 4.1	0b0: not supported 0b1: supported

Video Codec Type Value	Bits	Profile name	Interpretation
	3	Level 5	0b0: not supported 0b1: supported
	4	Level 5.1	0b0: not supported 0b1: supported
	15:5	Reserved	Set to all zeros

5.4.2 WFD R2 Aux Stream Formats

The WFD R2 Aux Stream Formats information is used to indicate the capability of a WFD R2 Device to support the codec and formats for encoding the content to be sent over auxiliary stream. The display_capability string for service discovery may include the WFD R2 Aux Stream Formats information. The format of the WFD R2 Aux Stream Formats information is shown in Table 79.

Table 79. WFD R2 Aux Stream Formats Information

Field	Size (octets)	Value	Description
Capability ID	1	13	Identifying the type of WFD R2 capability. The specific value is defined in Table 69.
Length	2	Variable	Length of the following fields in this information table.
Aux Stream Codecs bitmap	1		Bitmap defined in Table 80 detailing the supported codecs for auxiliary streams.
Supported Overlay Layers	1		When this field is transmitted by a WFD R2 Sink, it shall specify the maximum number of overlay layers it can handle when receiving auxiliary stream. When this field is transmitted by a WFD R2 Source, it shall specify the maximum number of overlay layers it may be using when sending auxiliary stream.

5.4.2.1 Aux Stream Codecs Bitmap

The Aux Stream Codecs bitmap represents the codecs supported by a WFD Device for encoding and decoding the content in an auxiliary stream. This bitmap is also used by the wfd2-aux-stream-formats parameter (specified in section 6.1.23).

If the wfd2-aux-stream-formats parameter is included in RTSP M3 Responses or in the WFD2 Aux Stream Formats Information that is included in Service Discovery frames, then B0 of the Aux Stream Codecs bitmap (Table 80) shall be set to one for all WFD R2 devices (except for R2 Secondary Sink) indicating all WFD R2 devices shall support PNG as a mandatory codec when capable of handling auxiliary streams.

Table 80. Aux Stream Codecs Bitmap

Bits	Codec name	Interpretation
0	PNG	0b0: not supported 0b1: supported
1	JPEG	0b0: not supported 0b1: supported
2	H.264 CBP	0b0: not supported 0b1: supported
7-3	Reserved	Set to all zeros

5.4.2.2 Reference Overlay Resolutions Bitmap

The Reference Overlay Resolution bitmap given in Table 81 represents the resolutions used by the WFD R2 Source for generating the overlay graphics sent in the auxiliary stream. The WFD R2 Source shall select the full-screen reference overlay resolution to be the minimum of the maximum video resolution supported by the Source, the maximum video resolution supported by the Sink, and the Sink's native resolution as described in Appendix B.1. This bitmap is used in the wfd2-aux-stream-formats parameter specified in section 6.1.23.

This bitmap may be included in the M4 Request message by a WFD R2 Source. If this bitmap is included in an RTSP M3 Response message, a WFD R2 Sink should set this bitmap to all zeros and a WFD R2 Source shall ignore this bitmap upon reception.

Table 81. Reference Overlay Resolution Refresh Rate

Bits	Name	Interpretation
1:0	Table Selection bits	0b00: Resolution/Refresh rate table selection: Index to CEA resolution/refresh rates (Table 34) 0b01: Resolution/Refresh rate table selection: Index VESA resolution/refresh rates (Table 35) 0b10: Resolution/Refresh rate table selection: Index HH resolutions/refresh rates (Table 36) 0b11: Reserved
7:2	Index bits	Index into resolution/refresh rate table selected by [B1:B0], values 47-63 are Reserved.

5.4.3 WFD R2 Audio Formats

The WFD R2 Audio Formats is used to indicate the audio capabilities of a WFD R2 Device. The format of the WFD R2 Audio Formats information is shown in Table 82, which extends the audio codecs and formats supported by a WFD R1 Device.

Table 82. WFD R2 Audio Formats Information

Field	Size (octets)	Value	Description
Capability ID	1	2	Identifies the type of WFD Capability. The specific value is defined in Table 69.
Length	2	15	Length of the following fields of this Information Table.
LPCM Modes bitmap	4		Bitmap defined in Table 83 detailing LPCM audio formats supported by the codec on the WFD R2 Device.
LPCM decoder latency field	1		Specifies the latency of the LPCM decoder at the WFD Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD Source in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD Source in an RTSP M4 Request message and the WFD Sink shall ignore this field upon reception. If the WFD Sink does not support this field, it shall set this field to all zeros. Otherwise the WFD Sink shall set this field to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.
AAC Modes bitmap	4		Bitmap defined in Table 44 detailing AAC audio formats supported by the codec on the WFD Device.
AAC decoder latency field	1		Specifies the latency of the AAC decoder at the WFD Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD Source in a WFD Service Discovery Response frame. This field shall be set to all zeros when transmitted by the WFD Sink that does not support AAC format at all, in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD Source in an RTSP M4 Request message and the WFD Sink shall ignore this field upon reception. If the WFD Sink does not support this field, it shall set this field to all zeros. Otherwise the WFD Sink shall set this field to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the

Field	Size (octets)	Value	Description
			presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.
AC3 Modes bitmap	4		Bitmap defined in Table 45 detailing AC3 audio formats supported by the codec on the WFD Device.
AC3 decoder latency field	1		Specifies the latency of the AC3 decoder at the WFD Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD Source in a WFD Service Discovery Response frame. This field shall be set to all zeros when transmitted by the WFD Sink that does not support AC3 format at all, in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD Source in an RTSP M4 Request message and the WFD Sink shall ignore this field upon reception. If the WFD Sink does not support this field, it shall set this field to all zeros. Otherwise the WFD Sink shall set this field to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.
CTA Audio Modes bitmap	4		Bitmap defined in Table 84 detailing CTA Audio supported by the WFD R2 Device.
CTA Audio latency	1		Specifies the latency of CTA Audio at the WFD R2 Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD R2 Source in a WFD Service Discovery Response frame. This field shall be set to all zeros when transmitted by the WFD R2 Sink that does not support any CTA format at all, in a WFD Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD R2 Source in an RTSP M4 Request message and the WFD R2 Sink shall ignore this field upon reception. If the WFD R2 Sink does not support this field, it shall set this field to all zeros, in an RTSP M3 Response message. Otherwise the WFD R2 Sink shall set this field, in an RTSP M3 Response message, to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.
AAC-ELDv2 Modes bitmap	4		Bitmap defined in Table 85 detailing AAC-ELDv2 formats supported by the codec on the WFD R2 Device.
AAC-ELDv2 decoder latency	1		Specifies the latency of the AAC-ELDv2 decoder at the WFD R2 Sink as an integer multiple of 5 msec. This field shall be set to all zeros when transmitted by the WFD R2 Source in a WFD Service Discovery Response frame. This field shall be set to all zeros when transmitted by the WFD R2 Sink that does not support any AAC-ELDv2 format at all, in a WFD2 Service Discovery Response frame. This field should be set to all zeros when transmitted by the WFD R2 Source in an RTSP M4 Request message and the WFD R2 Sink shall ignore this field upon reception. If the WFD R2 Sink does not support this field, it shall set this field to all zeros, in an RTSP M3 Response message. Otherwise the WFD R2 Sink shall set this field, in an RTSP M3 Response message to a best-effort estimate of the worst-case time between the availability of source data at the input interface of the decoder, and the presentation of the corresponding decoded data at the input interface of the rendering device, rounded up to the next higher multiple of 5 msec.

5.4.3.1 LPCM Modes Bitmap

The LPCM Modes bitmap represents LPCM configurations supported by the WFD R2 Device. This bitmap is also used in the ‘modes’ field of wfd2-audio-codecs as described in section 6.1.21.

If a wfd2-audio-codecs parameter is included in an RTSP M3 Response message or in a WFD Service Discovery frame, B1 of the LPCM Modes bitmap (Table 83shall be set to one for all WFD devices to indicate support of 2-channel LPCM audio at 16 bits/channel at 48000 samples/second as a mandatory mode of operation (except for a Primary Sink that does not have audio rendering capability, e.g., typical office projector. A device is deemed audio rendering capable if it can playback audio payload with or without the help of an external transducer, e.g., attached speaker). Other LPCM audio formats are optional at all WFD devices.

Table 83. LPCM Modes bitmap for WFD R2 devices

Bits	Interpretation		
	Sampling Frequency (kHz)	**Bit width (bits)**	**#channels**
0	44.1	16	2
1	48	16	2
2	48	16	1
3	48	16	6
4	48	16	8
5	48	24	2
6	96	16	2
7	96	24	2
8	96	24	6
9	96	24	8
10-31	Reserved		
Notes: If a WFD R2 Sink supports CTA Audio, the WFD R2 Sink shall only include LPCM modes with bits b0-b2 as LPCM audio formats in the wfd2-audio-codecs when indicating the supported audio formats to a WFD R2 Source capable of CTA audio during WFD capability negotiation. All other LPCM modes shall be indicated as CTA audio format with the details for sampling frequency, bit-width and channels included in the wfd2-cta-datablock-collection parameter (see section 6.1.29 for details).			

5.4.3.2 Capability negotiation for use of CTA Audio

A WFD R2 Device may support new audio formats specified in CTA-861-G [47], including advanced LPCM configurations and an extended number of compressed audio codecs. These audio formats are described collectively as CTA Audio in this specification.

If a WFD R2 Source supports CTA Audio for LPCM modes, then it shall include both wfd2-audio-codecs and wfd2-cta-datablock-collection in the M3 Request during capability negotiation with a WFD R2 Sink.

If a WFD R2 Sink supports CTA Audio and receives an M3 Request containing wfd2-audio-codecs and wfd2-cta-datablock-collection parameters, then it shall send an M3 Response containing both the wfd2-audio-codecs and wfd2-cta-datablock-collection parameters. The wfd2-audio-codecs shall include at least the mandatory audio format (2 channel 48 kHz, 16 bit LPCM) and may include only the LPCM modes with bit 0 (44.1 kHz, 16 bits 2 channel LPCM), and bit 2 in Table 83 as supported LPCM audio format. All other supported LPCM or compressed audio formats by the WFD R2 Sink shall be included as CTA audio formats in the wfd2-audio-codecs parameter and the details of all supported CTA audio modes shall be included in the wfd2-cta-datablock-collection parameter. The Audio Data Block shall be parsed to determine which compressed audio formats are supported and what forms of L-PCM audio are supported.

If a WFD R2 Source that supports CTA Audio receives an M3 Response containing both the wfd2-audio-codecs and wfd2-cta-datablock-collection parameters, it shall send an M4 Request with a wfd2-audio-codec parameter indicating the audio format to be used for the session. The WFD R2 Source shall only include the wfd2-audio-codecs parameter to indicate the mandatory audio format (i.e. 2 channel 48 kHz, 16 bit LPCM), or LPCM audio mode with bit 0 (44.1 kHz, 16 bits 2 channel LPCM), or bit 2 from Table 83 as the audio format for the WFD R2 Session. For any other audio format among the list of CTA audio modes supported by the WFD R2 Sink to be used in the WFD R2 session, the WFD R2 Source shall include the CTA audio format in the wfd2-audio-codecs parameter and a wfd2-audio-inforframe parameter to indicate the details of the CTA Audio format to be used in the WFD R2 session.

Note that CTA-861-G specification has two audio format codes for L-PCM audio. The legacy mode, (Audio Format Code = 1) supports from 2 to 8 channels. L-PCM 3D using extended format code = 0x0D supports from 1 to 32 channels. L-PCM 3D mode is a superset and includes legacy mode and therefore only L-PCM 3D mode is needed to cover all channel configurations up to 32 channels. Therefore, WFD R2 devices that support CTA Audio shall only use L-PCM 3D audio mode to indicate CTA L-PCM configurations.

If L-PCM 3D audio is supported, a Room Configuration Descriptor Data Block, (CTA-861.2, Table 3), shall be used to indicate the supported channel locations. The Room Configuration Data Block shall be followed by one or more Speaker Location Data Blocks, which contains a Speaker Location Descriptor for each available channel.

An example of how wfd2-cta-datablock-collection and wfd2-cta-audio-infoframe are used is shown in Appendix G.

5.4.3.3 CTA Audio Modes Bitmap

The CTA Audio Modes bitmap given in Table 84 indicates whether CTA Audio is supported by the WFD R2 Device.

Table 84. CTA Audio Modes Bitmap

Bits	Interpretation
0	CTA Audio support 0b0: not supported 0b1: supported
31:1	Reserved

CTA Audio indicates whether advanced multichannel codec support is available, including both compressed and Linear PCM formats. If CTA Audio is supported, bit 0 will be set to 1, otherwise it is set to 0.

5.4.3.4 AAC-ELDv2 Modes Bitmap

The AAC_ELDv2 Modes bitmap given in Table 85 represents whether AAC-ELDv2 compressed formats are supported by the WFD R2 Device. If the format is supported, the corresponding bit will be set to one.

Table 85. AAC-ELDv2 Modes Bitmap

Bits	Interpretation
0	AAC-ELD (AOT39) mono is supported
1	AAC-ELD (AOT39) stereo is supported
2	AAC-ELD (AOT39) + LD-MPS(AOT44) stereo is supported
3	AAC-ELD (AOT39) 5.1 channels is supported
4	AAC-ELD (AOT39) + LD-MPS(AOT44) 5.1 channels is supported
5	AAC-ELD (AOT39) 7.1 channels is supported
6	AAC-ELD (AOT39) + LD-MPS(AOT44) 7.1 channels is supported
31:7	Reserved

5.4.4 WFD R2 Extended Capabilities bitmap

The WFD R2 Extended Capabilities bitmap indicates whether UIBC, I2C Read/Write, Preferred Display mode, WFD Standby/resume control are supported by the WFD R2 Device. The display_capability string for service discovery may include the WFD R2 Extended Capabilities information as shown in Table 86.

Table 86. WFD R2 Extended Capabilities Bitmap

Bits	Name	Interpretation
0	UIBC Support bit	0b0: Not supported 0b1: Supported
1	I2C Read/Write Support bit	0b0: Not supported 0b1: Supported
2	Preferred Display mode Support bit	0b0: Not supported 0b1: Supported
3	Standby and Resume Control Support bit	0b0: Not supported 0b1: Supported
4	TDLS Persistent Support bit	0b0: Not supported 0b1: Supported
5	TDLS Persistent BSSID Support bit	0b0: Not supported 0b1: Supported
15:6	Reserved	Set to all zeros

5.5 TXT Record

The TXT record contains the relevant capabilities of the WFD Device in the form of key-value pairs.

Each key-value pair is encoded as its own constituent string within the DNS TXT record in the form of “key=value”. The key and value are both UTF-8 encoded. Table 87 below shows the mapping of the WFD Sub-element to the TXT record.

Table 87. WFD R2 Synchronous Streaming Information field bit mapping

WFD Sub-element	WFD Sub-element Field	Key	Value	Size (byte)
WFD Device Information	WFD Device Type	Type	p-sink	0x0B
			s-sink	0x0B
			Source	0x0B
			Dual	0x09
	WFD Session Availability Bit	ses_avail	1	0x0B
			0	0x0B
	CP Support Bit	Hdcp	0	0x06
			HDCP2.0	0x0C
			HDCP2.1	0x0C
			HDCP2.2	0x0C
	Audio un-supported at Primary Sink bit	aud_psink	0	0x0A
			1	0x0A
	Audio only support at WFD Source bit	aud_src	0	0x0A
			1	0x0A
N/A	N/A	proto_ver	1.0 or 2.0	0x0D
N/A	N/A	dev_name	up to 32 bytes UTF-8 String stringcharacters	variable up to 0x29
N/A	N/A	UUID	16 bytes Hex value	0x15

WFD Sub-element	WFD Sub-element Field	Key	Value	Size (byte)
N/A	N/A	1AS	0	0x0A
N/A	N/A	1AS	1	0x0A

6 RTSP based WFD control plane

This chapter defines the methods and messages that are used to establish, maintain, manage and teardown WFD Sessions.

A WFD Sink shall establish a layer 3 connection with a WFD Source before beginning WFD Session establishment and management. WFD Session management shall use RTSP [20] (RFC2326) over TCP as the communication protocol. The WFD Sink shall use the WFD Session management Control Port value (contained in the WFD IE) for all session management communication.

Since the RTSP specification [20] does not allow an RTSP server to initiate the SETUP, PLAY, PAUSE or TEARDOWN methods, this specification uses SET_PARAMETER messages with a wfd-trigger-method parameter to enable the RTSP server to trigger the client into initiating control operations while still maintaining compliance with [20].

6.1 RTSP data structures

This section (and sub-section) defines the data structures used by the WFD control plane. The data structures are WFD specific RTSP parameters. All definitions are in Augmented Backus-Naur Form (ABNF).

6.1.1 ABNF Definitions

The data structure definitions below use ABNF as defined in RFC 2234 [28] to define some elements. Some common ABNF elements used in this specification are summarized below.

```
DIGIT         = %x30-39                                                ; 0-9
HEXDIG        = DIGIT / %x41-46
SP            = %x20                                                   ; space
CR            = %x0D                                                   ; carriage return
LF            = %x0A                                                   ; linefeed
CRLF          = CR LF
IPADDRESS     = 1*3(DIGIT) “.” 1*3(DIGIT) “.” 1*3(DIGIT) “.” 1*3(DIGIT)
IPPORT        = 1*5(DIGIT)                                             ; must be between 0 and
                                                                       65535
```

6.1.2 wfd-audio-codecs

The wfd-audio-codecs parameter specifies the audio formats supported in the WFD Session. Valid audio codecs are listed in section 3.4.1.

```
wfd-audio-codecs      = “wfd_audio_codecs:” SP sink-audio-cap CRLF
sink-audio-cap        = “none” / sink-audio-list; “none” if not supported at a Primary Sink
sink-audio-list       = audio-format SP modes SP latency *(“,” SP sink-audio-list)
audio-format          = “LPCM” / “AAC” / “AC3”
modes                 = 8*8HEXDIG; see Table 43, Table 44 and Table 45
latency               = 2*2HEXDIG; decoder latency in units of 5 msecs, see LPCM decoder
                        latency field for LPCM, see AAC decoder latency field for AAC and see
                        AC3 decoder latency field in Table 42 for more detail
```

The sink-audio-list is a list of one or more <audio-format, modes, latency> tuples for each audio CODEC supported when included in RTSP M3 Response messages. The sink-audio-list is just one <audio-format, modes, latency> tuple when included in RTSP M4 Request messages.

Tuples for LPCM, AAC and/or AC-3 can appear in any order, in an RTSP M3 Response message.

6.1.3 wfd-video-formats

The wfd-video-formats parameter specifies the supported video resolutions (Table 34, Table 35, Table 36), H.264 codec profile (Table 38), level (Table 39), decoder latency, minimum slice size, slice encoding parameters and support for video frame rate control (including explicit Frame Rate Change and implicit video frame skipping). Valid H.264 codec configurations supported in this specification are listed in Table 6.

wfd-video-formats	= “wfd_video_formats:” SP sink-video-list CRLF
sink-video-list	= “none” / (native SP preferred-display-mode-supported SP H.264-codec);the Secondary Sink shall return “none”
native	= 2*2HEXDIG; see Table 37
preferred-display-mode-supported	= 2*2HEXDIG; 0-not supported, 1-supported, 2-255 reserved
H.264-codec	= profile SP level SP misc-params SP max-hres SP max-vres *(“,” SP H.264-codec)
profile	= 2*2HEXDIG; see Table 38, only one bit set
level	= 2*2HEXDIG; see Table 39, only one bit set
max-hres	= “none” / (4*4HEXDIG); in M3 response and if preferred-display mode-supported is 0, then “none”. in M3 response and if preferred-display-mode-supported is 1, specifies the maximum horizontal resolution that the H.264 decoder supports in pixels. in M4 request, it is “none” and the recipient shall ignore this subparameter
max-vres	= “none” / (4*4HEXDIG); in M3 response and if preferred-display mode-supported is 0, then “none”. in M3 response and if preferred-display-mode-supported is 1, specifies the maximum vertical resolution that the H.264 decoder supports in pixels. in M4 request, it is “none” and the recipient shall ignore this subparameter
misc-params	= CEA-Support SP VESA-Support SP HH-Support SP latency SP min-slice-size SP slice-enc-params SP frame-rate-control-support
CEA-Support	= 8*8HEXDIG; see Table 34 when used inside preferred display mode in M4 request, this subfield should be set to 0x00000000 and the recipient shall ignore this subfield
VESA-Support	= 8*8HEXDIG; see Table 35 when used inside preferred display mode in M4 request, this subfield should be set to 0x00000000 and the recipient shall ignore this subfield
HH-Support	= 8*8HEXDIG; see Table 36 when used inside preferred display mode in M4 request, this subfield should be set to 0x00000000 and the recipient shall ignore this subfield
latency	= 2*2HEXDIG; decoder latency in units of 5 msecs, see latency field in Table 33 for more detail
min-slice-size	= 4*4HEXDIG; number of macroblocks
slice-enc-params	= 4*4HEXDIG; see Table 40
frame-rate-control-support	= 2*2HEXDIG; see Table 41

The H.264-codec is a list of one or more <profile, level, misc-params, max-hres, max-vres> tuples for each H.264 profile, corresponding maximum level, miscellaneous parameters, maximum horizontal resolution, and maximum vertical resolution supported when included in RTSP M3 Response messages. In this case, level indicates the maximum level support for the specified profile. Tuples for CBP and RHP can appear in any order, in an RTSP M3 Response message.

A WFD Sink shall set the Video Frame Rate Change Support bit in the frame-rate-control-support field to one, in the RTSP M3 Response message, if it supports this functionality. If the Sink has not set this bit to one, then the WFD Source shall not set this bit to one in the RTSP M4 Request message. If the WFD Source does not support the functionality, it shall set this bit to zero.

The H.264-codec is just one <profile, level, misc-params, max-hres, max-vres> tuple when included in an RTSP M4 Request message. In this case level refers to the actual level to be used with the selected profile.

The min-slice-size field is expressed in number of macroblocks. WFD devices that do not support slice encoding in which a picture is constructed by multiple slices shall set this field to 0x00 00.

A WFD Source shall set the min-slice-size value to zero in the RTSP M4 Request message and shall not transmit an encoded picture constructed by multiple slices to a WFD Sink that does not support decoding picture constructed by multiple slices (i.e., the WFD Sink sets the min-slice-size value to zero in the RTSP M3 Response message).

A WFD Sink sets the min-slice-size value to the smallest slice size it can decode. Slices smaller than the min-slice-size value may not reduce latency. A WFD Source sets this value to the smallest encoded slice it may transmit.

Note that the actual realizable maximum horizontal resolution and maximum vertical resolutions are a function of horizontal resolution, vertical resolution and required macro block rate and may be smaller than the max-hres and/or the max-vres values indicated in this parameter.

6.1.4 wfd-3d-formats

Because the Stereoscopic 3D video as defined in the Wi-Fi Display Technical Specification v1.0 has been deprecated, this section is removed. This feature is not applicable for a WFD R2 Device.

6.1.5 wfd-content-protection

The wfd-content-protection parameter specifies whether the WFD Sink supports the HDCP system 2.x for content protection. (Note that after a transition period from the HDCP system 2.0 to the HDCP system 2.1 defined by DCP LLC has expired, “HDCP 2.0” shall not be used by the WFD devices seeking WFD certification. Refer to section 5.2 of [31] and its addendum [32] for more detail.) If content protection is not supported or is not currently possible for any reason, the parameter is set to “none”. If the WFD Sink supports HDCP 2.0 only, the parameter is set to “HDCP2.0” (Supporting HDCP 2.0 only is prohibited due to transition period has been expired). If the WFD Sink supports HDCP 2.0 and 2.1, the parameter is set to “HDCP2.1”. If the WFD Sink supports HDCP 2.0. 2.1 and higher version, the parameter is set to “HDCP 2.1”.

If the WFD Sink supports HDCP system 2.x, the TCP port number to be used on the WFD Sink for the HDCP 2.x AKE connection is included. The port number shall be between 1 and 65535.

```
wfd-content-protection    = “wfd_content_protection:” SP cp-spec CRLF
cp-spec                   = “none” / hdcp2-spec
hdcp2-spec                = (“HDCP2.0” / “HDCP2.1”) SP “port=” IPPORT ; TCP port
```

6.1.6 wfd-display-edid

The wfd-display-edid parameter specifies the EDID of the display on which the content will be rendered.

EDID data comes in multiples of 128-byte blocks as shown in Figure 26.

Display devices may contain 1 to 256 128-byte blocks of EDID data depending on the EDID structure that it supports.

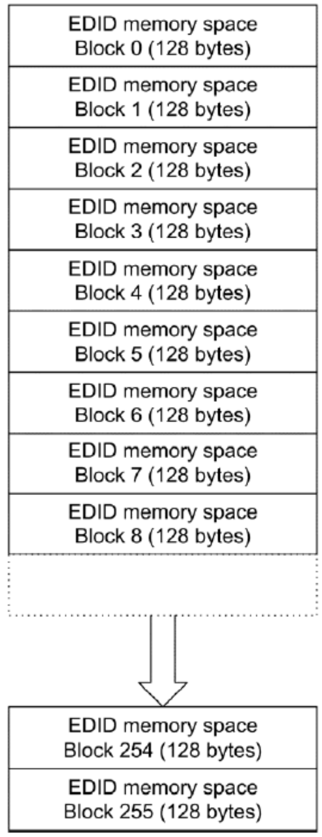


Figure 26. Block Structure of EDID data

The content and format of the EDID are not defined in this specification. Refer to the following standards for EDID structure:

- VESA Enhanced Extended Display Identification Data Standards (E-EDID) (e.g., E-EDID 1.4 [10])
- The various VESA E-EDID extension block standards
- The CEA E-EDID extension defined in the CTA-861 standard [11] and [2]

The WFD Sink shall follow the procedure for EDID access specified in VESA E-DDC v1.2 [12] (and later version if exists). When a WFD Sink responds to the query of wfd-display-edid parameter in an RTSP GET_PARAMETER Request message, the following rules are applied.

If a WFD Sink reports wfd-connector-type as HDMI or DP or UDI, the WFD Sink should return the EDID of the display that renders the streamed video.

A WFD Sink that supports the wfd-display-edid parameter shall include the entire EDID data structure that is available from the display device in a edid-payload field with indicating its length by a edid-count field in a unit of a number of 128-bytes block(s).A WFD Sink that does not support wfd-display-edid parameter shall set the edid field of the wfd-display-edid parameter to “none”, except for a WFD Sink dongle without an integrated display or with an integrated display that is not being used to render streamed video.

The WFD Sink dongle without an integrated display or with an integrated display that is not being used to render streamed video shall not set the edid field of the wfd-display-edid parameter to “none” regardless of whether an external display device is attached or not.

If EDID data is not available at the WFD Sink dongle without an integrated display or with an integrated display that is not being used to render streamed video, the edid-block-count field shall be set to 0x00 00 and the edid-payload field shall be “none”, instead of EDID data.

If EDID data is available at the WFD Sink dongle without an integrated display or with an integrated display that is not being used to render streamed video, the edid-block-count field and the edid-payload field shall be set to include entire EDID structure. In this case, The WFD Sink dongle without an integrated display or with an integrated display that is not being used to render streamed video should pass the EDID blocks from the connected external display device to the WFD Source as is, regardless of checksum failure and other error conditions.

```
wfd-display-edid        = "wfd_display_edid:" SP edid CRLF

edid                    = "none" / (edid-block-count SP edid-payload)

edid-block-count        = 4*4HEXDIG;
                              0: if EDID is not available at the time that
                                 the WFD Sink responses to the parameter
                                 request.
                             1-256: number of 128-byte EDID blocks in edid-payload
                             Other values: reserved
edid-payload            = "none" / 256*65536HEXDIG;
                              "none" if edid-block-count is 0. Otherwise, edid-payload
                          contains the entire EDID data structure available from the display
                          device. Length of edid-payload shall be a multiple of 128 bytes.
                          (length of edid-payload = edid-block-count * 128-byte)
```

6.1.7 wfd-coupled-sink

The wfd-coupled-sink parameter is used by a WFD Sink to convey its Coupled status and if Coupled to another WFD Sink, the Coupled WFD Sink's MAC address.

```
wfd-coupled-sink        = "wfd_coupled_sink:" SP coupled-sink-cap CRLF

coupled-sink-cap        = "none" /(status SP sink-address); "none" if Coupled Sink
                          Operation is not supported
status                  = 2*2HEXDIG; see Table 32

sink-address            = "none" / (12*12HEXDIG); WFD Sink's MAC address if status is
                          Coupled otherwise "none"
```

6.1.8 wfd-trigger-method

The wfd-trigger-method parameter is used by a WFD Source to trigger the WFD Sink to initiate an operation with the WFD Source.

```
wfd-trigger-method    = "wfd_trigger_method:" SP ("SETUP" / "PAUSE" / "TEARDOWN" / "PLAY") CRLF
```

6.1.9 wfd-presentation-url

The wfd-presentation-url parameter describes the Universal Resource Identifier (URI) to be used in the RTSP Setup (RTSP M6) Request message in order to setup the WFD Session from the WFD Sink to the WFD Source. The wfd-url0 and wfd-url1 values indicated in this parameter correspond to the rtp-port0 and rtp-port1 values from the wfd-client-rtp-ports parameter in the RTSP M4 Request message from the WFD Source to the WFD Sink at the end of the WFD Capability Negotiation phase.

```
wfd-presentation-url    = "wfd_presentation_URL:" SP wfd-url0 SP wfd-url1 CRLF

wfd-url0                = "none" / ("rtsp://" source-ip-address "/wfd1.0/streamid=0")

wfd-url1                = "none" / ("rtsp://" source-ip-address "/wfd1.0/streamid=1")

source-ip-address       = IPADDRESS
```

The values for wfd-url0 and wfd-url1 in the wfd-presentation-url in the RTSP M4 Request message are determined by Table 88.

Table 88. wfd-url0 and wfd-url1 values in wfd-presentation-url

Type of WFD Session	Type of sink(s)	wfd-client-rtp-ports in RTSP M4 Request		wfd-presentation-url in RTSP M4 Request	
		rtp-port0	rtp-port1	wfd-url0 specified?	wfd-url1 specified?
No Secondary Sink in WFD Session Single (audio only, or video only, or audio and video multiplexed) MPEG2_TS to a Primary Sink.	Primary Sink	Non zero	0	Yes	“none”
No Primary Sink in WFD Session Single (audio only) MPEG2_TS to a Secondary Sink.	Secondary Sink	0	Non zero	“none”	Yes
Two independent MPEG2_TS, one for video destined to the Primary Sink and one for audio destined to the Secondary Sink.	Primary Sink	Non zero	0	Yes	“none”
	Secondary Sink	0	Non zero	“none”	Yes
Note: Other combinations are reserved.					

6.1.10 wfd-client-rtp-ports

The wfd-client-rtp-ports parameter is used by a WFD Sink to convey the RTP port(s) that the WFD Sink is listening on and by the a WFD Source to indicate how audio, video or both audio and video payload will be encapsulated in the MPEG2-TS stream transmitted from the WFD Source to the WFD Sink.

```
wfd-client-rtp-ports    = “wfd_client_rtp_ports:” SP profile SP rtp-port0 SP rtp-port1 SP mode CRLF
profile                 = “RTP/AVP/UDP;unicast”
rtp-port0               = IPPORT ; UDP port
rtp-port1               = IPPORT ; UDP port
mode                    = “mode=play”
```

Note : MPEG2-TS (MP2T) described at section 5.7 in RFC3551 [6] is used with the AVP/ profile in this specification. Other profiles described at other sections in [6] are not relevant to this specification.

When a WFD Sink receives an M3 Request message querying the wfd-client-ports parameter, the WFD Sink shall set the rtp_port0 and rtp_port1 values in the wfd-client-rtp-ports parameter in the RTSP M3 Response message as shown in Table 89.

Table 89. wfd-client-rtp-ports parameter values in M3 Response message

rtp_port0	rtp_port1	Description
Non-zero	Non-zero	This combination is reserved.
Non-zero	0	Primary Sink that supports a single MPEG2-TS stream.
0	Non-zero	Secondary Sink that supports a single MPEG2-TS stream
0	0	This combination is reserved.

When a WFD Source sends a RTSP M4 Request message that includes the wfd-client-rtp-ports parameter as a part of WFD Capability Negotiation, the rtp-port0 and rtp-port1 values in the wfd-client-rtp-ports parameter shall be set as shown in Table 90. In addition, the client-port value in Transport header line of an RTSP M6 Request message shall be set as shown in Table 90.

Table 90. wfd-client-rtp-ports parameter values in M3 Response message and corresponding values in the subsequent M4/M6 Request message

Received RTSP M3 Response		Possible RTSP M4 Request		Possible RTSP M6 Request	Description
rtp-port0	rtp-port1	rtp-port0	rtp-port1	client-port	
Non-zero	0	Non-zero	0	rtp-port0	WFD Source transmits an MPEG2-TS stream to the Primary Sink. Audio and video session, audio-only or video-only session. WFD Source streams corresponding content to rtp-port0.
0	Non-zero	0	Non-zero	rtp-port1	WFD Source transmits an MPEG2-TS stream containing audio only to the Secondary Sink.
Note: Other combinations are reserved.					

6.1.11 wfd-route

The wfd-route parameter provides a mechanism to specify the destination to which the audio stream is to be routed.

```
wfd-route       = “wfd_route:” SP destination CRLF
destination     = “primary” / “secondary”
```

6.1.12 wfd-I2C

The wfd-I2C parameter is used by a WFD Source to inquire whether a WFD Sink supports remote I2C Read/Write function or not.

If the WFD Sink does not support remote I2C Read/Write function, it shall set the value of this parameter to “none” when responding to the query of this parameter.

If the WFD Sink supports remote I2C Read/Write function, it shall set the value of this parameter to the TCP port number to be used by the WFD Source to exchange remote I2C Read/Write messaging transactions with the WFD Sink.

Refer to section 7 for details of Remote I2C Read/Write Messaging Transactions.

```
wfd-I2C         = “wfd_I2C:” SP I2C-port CRLF
I2C-port        = “none” / IPPORT; port where the device listens for I2C commands, “none” if
                  not supported
```

6.1.13 wfd-av-format-change-timing

The wfd-av-format-change-timing parameter is used to signal the actual AV format change timing of the streaming data to the WFD Sink. It shall not be included in an RTSP M4 Request message for the first WFD Capability Negotiation before the establishment of RTSP connection. It shall be included in an RTSP M4 Request message for WFD Capability Re-negotiation after a WFD Session has been established.

The PTS field represents PTS values that are included in the PES header of the PES corresponding to AV format change. The DTS field represents the DTS values that are included in the PES header of the PES corresponding to AV format change. The least-significant 7 bits of the PTS and DTS fields are reserved and are set to all zeros.

```
wfd-av-format-change-   = “wfd_av_format_change_timing:” SP PTS SP DTS CRLF
timing
PTS                     = 10*10HEXDIG; most-significant 33 bits indicating PTS value
DTS                     = 10*10HEXDIG; most-significant 33 bits indicating DTS value
```

6.1.14 wfd-preferred-display-mode

Because the Preferred Display Mode as defined in the Wi-Fi Display Technical Specification v1.0 has been deprecated, this section is removed. This feature is not applicable for a WFD R2 Device.

6.1.15 wfd-uibc-capability

The wfd-uibc-capability parameter describes support for the user input back channel (UIBC) and related attributes. Support for UIBC is indicated using the UIBC Support bit (B0) of the WFD Extended Capability bitmap in the WFD Extended Capability subelement. Note that “none” indicates that the corresponding sub-parameter value is not supported.

```
wfd-uibc-capability       = “wfd_uibc_capability:” SP (“none” / (input-category-val “;”
                            generic-cap-val “;” hidc-cap-val “;” tcp-port)) CRLF; “none” if
                            not supported
input-category-val        = “input_category_list=” (“none” / input-category-list)
input-category-list       = input-cat * (“,” SP input-category-list)
input-cat                 = “GENERIC” / “HIDC”
generic-cap-val           = “generic_cap_list=” (“none” / generic-cap-list)
generic-cap-list          = inp-type *(“,” SP generic-cap-list)
inp-type                  = “Keyboard” / “Mouse” / “SingleTouch” / “MultiTouch” / “Joystick”
                            / “Camera” / “Gesture” / “RemoteControl”
hidc-cap-val              = “hidc_cap_list=” (“none” / hidc-cap-list)
hidc-cap-list             = detailed-cap *(“,” SP hidc-cap-list)
detailed-cap              = inp-type “/” inp-path
inp-path                  = “Infrared” / “USB” / “BT” / “Zigbee” / “Wi-Fi” / “No-SP”; “No-
                            SP” means vendor specific
tcp-port                  = “port=” (“none” / IPPORT)
```

The WFD Source indicates the TCP port number to be used for UIBC in the tcp-port field of the wfd-uibc-capability parameter in RTSP M4 and/or M14 Request messages. The WFD Sink uses “none” for the tcp-port field of the wfd-uibc-capability parameter, in RTSP M3 Response and M14 Request messages.

6.1.16 wfd-uibc-setting

The wfd-uibc-setting parameter is used to enable and disable the UIBC.

```
wfd-uibc-setting          = “wfd_uibc_setting:” SP uibc-setting CRLF
uibc-setting              = “disable” / “enable”
```

The wfd-uibc-setting parameter may be included in the first RTSP M4 Request message during the WFD Capability Negotiation, provided that the RTSP M4 Request message contains the wfd-uibc-capability parameter.

6.1.17 wfd-standby-resume-capability

The wfd-standby-resume-capability parameter describes support of both standby control using a wfd-standby parameter and resume control using PLAY and using triggered-method setting PLAY.

```
wfd-standby-resume-capability     = “wfd_standby_resume_capability:” SP standby-resume-cap
                                    CRLF
standby-resume-cap                = “none” / “supported”; “none” if not supported
```

6.1.18 wfd-standby

The wfd-standby parameter is used to indicate that the sender of this parameter in an RTSP SET_PARAMETER Request message is entering WFD Standby mode.

```
wfd-standby              = "wfd_standby" CRLF
```

6.1.19 wfd-connector-type

A WFD Source may send an RTSP GET_PARAMETER Request message to a WFD Sink with the wfd-connector-type parameter to inquire about the connector type that is currently active in the WFD Sink.

After a change of the active connector type, the WFD Sink may send an RTSP SET_PARAMETER Request message to the WFD Source with the wfd-connector-type parameter. This is to inform the WFD Source about the new active connector type now in use by the WFD Sink. A WFD Sink shall not send an RTSP SET_PARAMETER Request message to the WFD Source with the wfd-connector-type parameter unless the WFD Source supports this parameter. Support of this parameter by the WFD Source is indicated when the WFD Source sends an RTSP GET_PARAMETER Request message with the wfd-connector-type parameter at least once after the successful RTSP M1 and M2 message exchanges.

This mechanism allows 0 or 1 active connector to be reported by the WFD Sink.

Handling of multiple connector types simultaneously is implementation specific and is out of the scope of this specification.

```
wfd-connector-type       = "wfd_connector_type:" SP connector-type CRLF
connector-type           = "none" / (2*2HEXDIG); specifies the active display connector
                           type (see Table 91), "none" if the WFD Sink dongle that is not
                           acting as a WFD Sink with an integrated display is not connected
                           to an external display
```

The connector-type field is used to indicate the Display Connector Type currently in use by the WFD Sink. When a WFD Sink responds to a query of the wfd-connector-type parameter in an RTSP GET_PARAMETER Request message or when a WFD Sink sends this parameter in an RTSP SET_PARAMETER Request message, the following rules apply.

The WFD Sink dongle that is not connected to an external display and it is not acting as a WFD Sink with embedded display (to render streamed content) shall return a value of "none".

Otherwise, the WFD Sink shall choose a non-reserved value from Table 91.

Table 91. Display connector type field

Value	Description
0	Video Graphics Array (VGA) Connector
1	S-Video connector
2	Composite video connector
3	Component video connector
4	Digital Video Interface (DVI) connector
5	High-Definition Multimedia Interface (HDMI) connector
6	Reserved
7	Wi-Fi Display
8	Japanese D connector. (A connector conforming to the EIAJ RC-5237 standard)
9	Serial Digital Image (SDI) connector
10	A Display Port connector (DP)

Value	Description
11	Reserved
12	A Unified Display Interface (UDI)
13-254	Reserved
255[1]	Indicates a type of physical connector that is not listed from value 0 to 254
Notes: 1. If the value 255 as connector-type is reported from the WFD Sink, WFD Sources may not be able to unambiguously identify the connector type that is in use. Due to this reason, some WFD Sources may not be able to recognize the WFD Sink at all or may interoperate in a sub-optimal manner.	

6.1.20 wfd-idr-request

The wfd-idr-request parameter is used by the WFD Sink to request an IDR picture from the WFD Source.

```
wfd-idr-request          = "wfd_idr_request" CRLF
```

6.1.21 wfd2-audio-codecs

The wfd2-audio-codecs parameter specifies the audio formats that may be supported in a WFD R2 Session. Valid audio codecs are listed in section 3.4.1.

```
wfd2-audio-codecs          = "wfd2_audio_codecs:" SP sink-audio-cap CRLF
sink-audio-cap             = "none" / sink-audio-list; "none" if not supported at a Primary Sink
sink-audio-list            = audio-format SP modes SP latency *("," SP sink-audio-list)
audio-format               = "LPCM" / "AAC" / "AC3" / "CTA" / "AAC-ELDv2"
modes                      = 8*8HEXDIG; Table 44, Table 45, Table 82, Table 83
latency                    = 2*2HEXDIG; decoder latency in units of 5 msecs, see LPCM decoder
                             latency field for LPCM, see AAC decoder latency field for AAC, see AC3
                             decoder latency field, see CTA decoder latency field and see AAC-ELDv2
                             decoder latency field in Table 82 for more detail
```

The sink-audio-list is a list of one or more <audio-format, modes, latency> tuples for each audio codec supported when included in RTSP M3 Response messages. The sink-audio-list is just one <audio-format, modes, latency> tuple when included in RTSP M4 Request messages.

Tuples for LPCM, AAC and/or AC-3 may appear in any order in an RTSP M3 Response message.

If both the wfd2-audio-codecs and wfd-audio-codecs parameters are included in the RTSP M3 Request message, a WFD R2 Sink shall ignore the wfd-audio-codecs parameter and only include the wfd2-audio-codecs parameter in the subsequent RTSP M3 Response message to indicate the codec parameters that the WFD R2 Sink supports for the WFD Session. A WFD R2 Source shall only include the wfd2-audio-codecs parameter (i.e. not use wfd-audio-codecs parameter) in the RTSP M4 Request message to set the audio codec parameters for the WFD Session with the WFD R2 Sink.

6.1.22 wfd2-video-formats

The wfd2-video-formats parameter specifies the video codecs and formats that may be supported by a WFD R2 Device with the codec type (Table 76), codec profiles (Table 77), level (Table 78), decoder latency, minimum slice size, slice encoding parameters and support for video frame rate control (including explicit Frame Rate Change and implicit video frame skipping).

```
Wfd2-video-formats         = "wfd2_video_formats:" SP sink-video-cap (SP portrait-mode) CRLF
sink-video-cap             = "none" / (native SP video-codecs SP non-transcoding-support); the
                             Secondary Sink shall return "none"
```

native	= 2*2HEXDIG; see Table 74
video-codecs	= codec SP profile SP level SP misc-params *(“,” SP video-codecs)
codec	= 2*2HEXDIG; see Table 76
profile	= 2*2HEXDIG; see Table 77, only one bit set per video-codec
level	= 4*4HEXDIG; see Table 78, only one bit set per video-codec
misc-params	CEA-Support SP VESA-Support SP HH-Support SP latency SP min-slice-size SP slice-enc-params SP frame-rate-control-support
CEA-Support	= 12*12HEXDIG; see Table 71
VESA-Support	= 12*12HEXDIG; see Table 72
HH-Support	= 12*12HEXDIG; see Table 73
latency	= 2*2HEXDIG; decoder latency in units of 5 msecs, see latency field in Table 33 for more detail
min-slice-size	= 4*4HEXDIG; number of macroblocks
slice-enc-params	= 4*4HEXDIG; see Table 40
frame-rate-control-support	= 2*2HEXDIG; see Table 41
non-transcoding-support	= 2*2HEXDIG; 0-not supported, 1-supported, 2-255 reserved
portrait-mode	= “enabled”; this field is present only in M4 Request message to indicate that current orientation is portrait

The video-codecs is a list of one or more <codec, profile, level, misc-params> tuples for each video codec, profile, corresponding maximum level and miscellaneous parameters supported when included in RTSP M3 Response messages. In this case, level indicates the maximum level support for the specified profile. Tuples for video-codecs can appear in any order, in an RTSP M3 Response message.

A WFD Sink shall set the Video Frame Rate Change Support bit in the frame-rate-control-support field to one, in the RTSP M3 Response message, if it supports this functionality. If the Sink has not set this bit to one, then the WFD Source shall not set this bit to one in the RTSP M4 Request message. If the WFD Source does not support the functionality, it shall set this bit to zero.

The video-codecs is just one <codec, profile, level, misc-params> tuple when included in an RTSP M4 Request message. In this case level refers to the actual level to be used with the selected profile of the selected codec. If both the wfd2-video-formats and wfd-video-formats parameters are included in the RTSP M3 Request message, a WFD R2 Sink shall ignore the wfd-video-formats parameter and only include the wfd2-video-formats parameter in the subsequent RTSP M3 Response message to indicate the codec parameters that the WFD R2 Sink supports for the WFD Session. A WFD R2 Source shall only include the wfd2-video-formats parameter (i.e. not use wfd-video-formats parameter) in the RTSP M4 Request message to set the video format parameters for the WFD Session with the WFD R2 Sink.

The min-slice-size field is expressed in number of macroblocks. WFD devices that do not support slice encoding in which a picture is constructed by multiple slices shall set this field to 0x00 00.

A WFD Source shall set the min-slice-size value to zero in the RTSP M4 Request message and shall not transmit an encoded picture constructed by multiple slices to a WFD Sink that does not support decoding picture constructed by multiple slices (i.e., the WFD Sink sets the min-slice-size value to zero in the RTSP M3 Response message).

A WFD Sink sets the min-slice-size value to the smallest slice size it can decode. Slices smaller than the min-slice-size value may not reduce latency. A WFD Source sets this value to the smallest encoded slice it may transmit.

A WFD Sink shall set the non-transcoding-support field value to 0x01 in the RTSP M3 Response if it is capable to support the non-transcoding mode. A WFD Source shall always set this field to 0x00 in the RTSP M4 Request.

6.1.23 wfd2-aux-stream-formats

The wfd2-aux-stream-formats parameter specifies the codecs and formats that may be supported by a WFD R2 Device for the content to be sent over auxiliary stream with the supported codec type (Table 76) and associated parameters.

wfd2-aux-stream-formats	= "wfd2_aux_stream_formats:" SP sink-aux-stream-cap CRLF
sink-aux-stream-cap	= "none" /sink-aux-stream-info; "none" if auxiliary stream is not supported by an R2 Primary Sink. Secondary Sink shall return "none"
sink-aux-stream-info	= aux-codecs SP max_overlay_layers SP reference-overlay-resolution;
aux-codecs	= 2*2HEXDIG; see Table 80
max_overlay_layers	= 2*2HEXDIG; maximum number of overlay layers that can be handled by the WFD R2 Sink
reference-overlay-resolution	= 2*2HEXDIG; indexed display resolution using bitmap from Table 81

The wfd2-aux-stream-formats parameter is included in the RTSP M3 Request message sent by a WFD R2 Source capable to support auxiliary streams to query about the auxiliary stream capability of a WFD R2 Sink. The wfd2-aux-stream-formats parameter may be included in the RTSP M4 Request message when the WFD R2 Sink has indicated support for handling of auxiliary stream in its RTSP M3 Response message.

6.1.24 wfd2-direct-streaming-mode

The wfd2-direct-streaming-mode parameter is sent by the WFD R2 Source to a WFD R2 Sink to indicate whether the direct streaming mode will be activated after completion of the M4 Request/Response message exchange.

wfd2-direct-streaming-mode	= "wfd2_direct_streaming_mode:" SP direct-streaming CRLF
direct-streaming	= "active" / "inactive"; "active" to indicate that the streaming mode is direct streaming, and "inactive" to indicate that direct streaming mode is not active, see section 4.16.3 for more details

The wfd2-direct-streaming-mode parameter may be included in the RTSP M4 Request message during establishment of WFD Session or any time during an active WFD Session.

6.1.25 wfd2-transport-switch

The wfd2-transport-switch parameter is used to switch the transport for streaming over the TCP or UDP transport.

wfd2-transport-switch	= "wfd2_transport_switch:" SP profile SP mode CRLF
profile	= ("RTP/AVP/TCP;unicast" / "RTP/AVP/UDP;unicast")
mode	= "mode=play"

NOTE: The existing wfd-client-rtp-ports parameter is not reused for sending TCP or UDP port information after the initial WFD Session establishment since this parameter is included in the M3 Request during initial capability negotiation and the Sink returns it with the RTP/AVP/UDP profile and ports to be used for UDP. The wfd2-transport-switch parameter is used to perform a transport switch from UDP to TCP or vice versa after the WFD Session has been established, the exchange of the port information for the requested transport mode is performed through the RTSP M6 (SETUP) Request/Response messages in this case.

6.1.26 wfd2-buffer-length

The wfd2-buffer-length parameter specifies the initial buffer size the WFD R2 Sink is able to support and the amount of buffer the WFD R2 Source requests to be set at the WFD R2 Sink. The WFD R2 Sink may respond with the actual amount of buffer length it will use.

wfd2-buffer-length	= "wfd2_buffer_length:" SP buffer-length CRLF
buffer-length	= 4*4DIGIT; buffer length in milliseconds

After a WFD Session has been established, the WFD R2 Source may include the wfd2-buffer-length parameter in an M3 Request (GET_PARAMETER Request) any time in order to query the buffer length the WFD R2 Sink can support. The WFD R2 Sink shall set the wfd2-buffer-length parameter in the M3 Response (GET_PARAMETER Response) message to indicate the amount of buffer length it can allocate at that time for a subsequent transport switch after the M3 Request/Response messages.

After receiving an M3 Response message containing the wfd2-buffer-length parameter during an ongoing WFD Session, the WFD R2 Source shall include the buffer length the WFD R2 Source prefers the WFD R2 Sink to use in the wfd2-buffer-length parameter in the subsequent M4 Request (SET_PARAMETER Request) message for initiating a transport switch. If the WFD R2 Sink is only able to support lower amount of buffer length, the WFD R2 Sink may respond with RTSP OK and include the wfd2-buffer-length parameter with the other buffer length value in the M4 Response message. If the wfd2-buffer-length parameter is received in the M4 Response with RTSP OK and the WFD R2 Source determines that the lower amount of buffer length is sufficient to switch to TCP, the WFD R2 Source shall use the buffer length value provided by the WFD R2 Sink for the streaming over TCP .If the WFD R2 Source determines that the amount of buffer length supported by the WFD R2 Sink is not sufficient to switch to TCP, the WFD R2 Source shall send a subsequent RTSP M4 Request message with the wfd-client-rtp-ports parameter to continue streaming over UDP after receiving the M4 Response with RTSP OK from the WFD R2 Sink.

NOTE: If the Sink cannot support the requested amount of buffer, it may return a different value in the wfd2-buffer-length parameter in the M4 Response message with the status code 200 (RTSP OK). The value included in the wfd2-buffer-length parameter in the message-body of the M4 Response message is a suggested value supported by the Sink.

6.1.27 wfd2-audio-playback-status

The wfd2-audio-playback-status parameter is used by the WFD R2 Source to query the current status of the audio playback to include the amount of audio data currently buffered and the PTS value of the audio currently being played at the WFD Sink. The WFD R2 Sink returns the duration of the buffered audio data in milliseconds. The PTS value returned is in the original 90 kHz clock domain.

```
wfd2-audio-playback-     =  "wfd2_audio_playback_status:" SP audio-buffer-size  SP audio-
status                   playback-pts CRLF
audio-buffer-size        = "none" / buffered-audio; "none" if the Sink is not able to
                         indicate the buffer size
buffered-audio           = 4*4DIGIT; buffered audio in milliseconds

audio-playback-pts       = ""none" / audio-PTS; "none" if the Sink is not able to indicate
                         the playback PTS
audio-PTS                = 10*10HEXDIG; most-significant 33 bits indicating PTS value
```

Once a WFD Session has been established, the WFD R2 Source may include the wfd2-audio-playback-status parameter in an M3 Request (GET_PARAMETER Request) any time in order to query the current status of the audio playback at the WFD R2 Sink.

After receiving an M3 Request message containing the wfd2-audio-playback-status parameter, the WFD R2 Sink shall include the wfd2-audio-playback-status parameter in the M3 Response (GET_PARAMETER Response) message. The WFD R2 Sink shall set the audio-buffer-size value of the wfd2-audio-playback-status parameter to indicate the amount of currently buffered audio data when this information is available, otherwise the audio-buffer-size value shall be set to "none". The WFD R2 Sink shall set the audio-playback-pts value of the wfd2-audio-playback-status parameter to indicate the PTS of the audio data currently being played when this information is available; otherwise the audio-playback-pts value shall be set to "none".

6.1.28 wfd2-video-playback-status

The wfd2-video-playback-status parameter is used by the WFD Source to query the current status of the video playback to include the amount of video data currently buffered and the PTS value of the video currently being played at the WFD Sink. The WFD Sink returns the duration of the buffered video data in milliseconds. The PTS value returned is in the original 90 kHz clock domain.

```
video-buffer-size        = "none" / buffered-video; "none" if the Sink is not able to
                         indicate the buffer size
wfd2-video-playback-     =  "wfd2_video_playback_status:" SP video-buffer-size SP  video-
status                   playback-pts CRLF
buffered-video           = 4*4DIGIT; buffered video in milliseconds
```

```
video-playback-pts        = "none" / video-PTS ; "none" if the Sink is not able to indicate
                            the playback PTS
video-PTS                 = 10*10HEXDIG; most-significant 33 bits indicating PTS value
```

Once a WFD Session has been established, the WFD R2 Source may include the wfd2-video-playback-status parameter in an M3 Request (GET_PARAMETER Request) any time in order to query the current status of the video playback at the WFD R2 Sink.

After receiving an M3 Request message containing the wfd2-video-playback-status parameter, the WFD R2 Sink shall include the wfd2-video-playback-status parameter in the M3 Response (GET_PARAMETER Response) message. The WFD R2 Sink shall set the video-buffer-size value of the wfd2-video-playback-status parameter to indicate the amount of currently buffered video data when this information is available; otherwise the video-buffer-size value shall be set to "none". The WFD R2 Sink shall set the video-playback-pts value of the wfd2-video-playback-status parameter to indicate the PTS of the video data currently being played when this information is available; otherwise the video-playback-pts value shall be set to "none".

6.1.29 wfd2-cta-datablock-collection

The wfd2-cta-datablock-collection parameter is used to convey CTA-861-F (and later) capabilities. These include the following:

- Audio Data Block
- Speaker Allocation Data Block
- Room Configuration Data Block
- Vendor Specified Data Block

The format for a CTA Data Block collection encapsulated in an RTSP message is shown below.

```
wfd2-cta-datablock-          = "wfd2_cta_datablock_collection:" SP
collection                           cta-datablock-collection CRLF
cta-datablock-collection     = "none" /
                                  (cta-datablock-collection-length [SP cta-datablock-collection-
                               payload])
cta-datablock-collection-    = 4*4HEXDIG;
length                            0: if not available, 2-65534: Length of cta-datablock-collection-
                               payload
cta-datablock-collection-    2*65534HEXDIG;
payload                           cta-datablock-collection-payload contains a collection of CTA-
                               861-F (and CTA-861.2) data blocks
```

The cta-datablock-collection-payload is constructed using the data structures described in CTA-861-F. The payload is made up of a collection of CTA data blocks.

Byte 1 of each CTA data block is a data block header as shown in Table 92.

Table 92. CTA data block header

Byte #	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Tag Code			Length of following data block payload (in bytes)				

CTA Data Block Tag Codes are defined in CTA-861-F, Table 44. It is possible to use an Extended Tag Format, which is signaled by a Tag Code value of 7, and further defined in Byte 2 of the data block as shown in Table 93.

Table 93. CTA Extended Tag Code

Byte #	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2	Extended Tag Code							

Extended Tag Codes are defined in CTA-861-F and extended in CTA-861.2.

Multiple data blocks may be combined into a collection. A parser is responsible for doing the following:

- Read the Tag Code and Length fields
- If Tag Code is known, parse accordingly
- If Tag Code is unknown, skip by an equal number of bytes as specified in the Length field to the next data block
- Repeat until the length of the data block collection (as signaled in the RTSP message) is reached

Descriptions of the CTA data blocks are in the following sections.

6.1.29.1 Audio Data Block

(CTA-861-F, Section 7.5.2 and CTA-861.2) - Tag Code 1

The Audio Data Block shall be used to indicate which CTA audio formats (beyond the formats signaled using wfd2-audio-codecs), are supported in a WFD R2 Sink.

Allowed audio formats in WFD R2 are shown in Table 94.

Table 94. CTA audio formats allowed in WFD R2

Format Code	CXT Code	Audio Coding Type
0x02	n/a	AC-3
0x06	n/a	AAC-LC
0x0A	n/a	Enhanced AC-3
0x0B	n/a	DTS-HD
0x0F	0x04	MPEG-4 HE AAC
0x0F	0x05	MPEG-4 HE AAC v2
0x0F	0x06	MPEG-4 AAC LC
0x0F	0x0A	MPEG-H 3D Audio
0x0F	0x0C	AC-4
0x0F	0x0D	L-PCM 3D Audio

Note that Format Code 0x01 is not supported. L-PCM 3D Audio (extended Audio shall be used to indicate support for optional L-PCM modes not supported in the LPCM Modes bitmap).

6.1.29.2 Room Configuration Data Block

(CTA-861.2, Section 3.3) - Tag Code 7, Extended Tag Code 0x13

If L-PCM 3D Audio is declared in the Audio Data Block, wfd2-cta-datablock-collection shall include the Room Configuration Data Block. In the Room Configuration Data Block, the SLD parameter shall be set to 1, Speaker shall be set to 1, as described in CTA-861.2, Section 3.3.

6.1.29.3 Speaker Location Data Block

(CTA-861.2, Section 3.4.1) - Tag Code 7, Extended Tag Code 0x14

If the Room Configuration Data Block is included in the wfd2-cta-datablock-collection, then one or more Speaker Location Data Blocks shall follow the Room Configuration Data Block. Speaker Count in the Room Configuration Data Block shall indicate the total number of Speaker Location Descriptors that are included in Speaker Location Data Blocks.

Note that a CTA Data Block is (by definition) limited to 31 bytes. It is recommended to fully utilize Data Blocks rather than sending each descriptor in a separate Data Block. In the case of Speaker Location Descriptors, since they are either two bytes or 5 bytes, as few as 6 or as many as 15 descriptors can be sent in one Data Block.

6.1.29.4 Vendor-Specific Data Block

The Vendor-Specific Data Block (VSDB) is an optional Data Block that contains information not defined by CTA-861 that may be of interest to the vendor's own sources.

Appendix G shows examples of CTA Audio data blocks and how they are used with wfd2-cta-datablock-collection.

6.1.30 wfd2-cta-audio-infoframe

The wfd2-cta-audio-infoframe parameter specifies information about the audio when using CTA Audio.

The format for a CTA Audio InfoFrame encapsulated in an RTSP message is shown below.

```
wfd2-cta-audio-infoframe    = "wfd_cta_audio_infoframe:" SP cta-audio-infoframe CRLF

cta-audio-infoframe         =  "none" / (cta-audio-infoframe-length [SP cta-infoframe-payload])

cta-audio-infoframe-        = 4*4HEXDIG;
length                      0: if not available, 2-65534: Length of cta-audio-infoframe-payload
cta-audio-infoframe-        = 2*65534HEXDIG; cta-audio-infoframe-payload contains Enhanced Audio
payload                     InfoFrame as defined in CTA-861.2, described in section 3.7, Delivery
                            by Channel Index
```

A WFD R2 Source indicates information about the audio to the WFD R2 Sink in the cta-audio-infoframe-payload field of the wfd2-cta-audio-infoframe parameter in an RTSP M4 Request message.

An example of how to construct a cta-audio-infoframe-payload is shown in Appendix G.

6.1.30.1 CTA Audio - Delivery by Channel Index

(CTA-861.2, Section 3.7)

A WFD R2 Source shall indicate, using wfd2-cta-audio-infoframe, the L-PCM stream being sent to a WFD R2 Sink, using the Delivery by Channel Index Audio InfoFrame format as specified in CTA-861.2, Section 3.7.

6.2 WFD RTSP methods

Table 95 lists RTSP methods (including OPTIONS) from RFC2326 [20] that WFD devices shall support.

- A WFD Device shall be able to send and respond to the OPTIONS, SET_PARAMETER, GET_PARAMETER, and org.wfa.wfd1.0 RTSP methods.
- A WFD Sink shall be able to send the SETUP, PLAY, PAUSE and TEARDOWN RTSP methods.
- A WFD Source shall be able to respond to the SETUP, PAUSE, PLAY, and TEARDOWN RTSP method. Note that a PLAY after a PAUSE may have different behavior from the typical usage in RTSP. See the notes underneath Table 95 for details.
- A WFD Source shall use "rtsp://localhost/wfd1.0" as the URI for requests sent to a WFD Sink. This does not correspond to a stream as per usual RTSP convention, but instead is used to indicate that the request is for the capabilities of the WFD Sink.
- A WFD Sink shall use the wfd-presentation-url value (see section 6.1.9) as the URI for requests sent to a WFD Source.
- A WFD Device should ignore any parameter not specified in Table 97, in GET_PARAMETER or SET_PARAMETER.

6.2.1 WFD RTSP OPTIONS

The RTSP OPTIONS request is used by a WFD Device to verify that the WFD peer supports the required methods. A WFD Device shall abort the RTSP M1 and/or M2 message exchange(s) if any of the required methods are missing.

Note: There is no need for to explicitly communicate that the RTSP M1 and/or M2 message exchange(s) has been terminated. If a WFD Device aborts the RTSP M1 and/or M2 message exchange(s) on receipt of a response from the peer WFD Device, future RTSP requests from the peer WFD Device will result in RTSP Responses with a status code indicating error.

A WFD Device shall use the tag “org.wfa.wfd1.0” to query a peer WFD Device for the options that the peer WFD Device supports.

In response to an OPTIONS request, a WFD Sink shall respond indicating support for org.wfa.wfd1.0, SET_PARAMETER, and GET_PARAMETER.

In response to an OPTIONS request, a WFD Source shall respond indicating support for org.wfa.wfd1.0, SET_PARAMETER, GET_PARAMETER, SETUP, PLAY, PAUSE and TEARDOWN.

Table 95 summarizes the requirements for support of each RTSP method. For example, when it is marked as “Required” in “WFD Source->WFD Sink direction”, it means that the WFD Source shall support transmission of this method and the WFD Sink shall support reception of this method. Note that listed methods in RTSP messages shown in the example are lists of supported methods at reception, as specified in [20].

*When streaming content encoded in real-time, the definition of PAUSE is different from what it is in RTSP. When real-time content is rendered using Wi-Fi Display, PAUSE causes the WFD Source to stop streaming. When the session is resumed streaming does not continue from the point where it was paused but from the ‘current’ content. The WFD Source does not cache the real-time content from the point where the stream was paused.

Table 95. RTSP methods that the WFD Source and/or WFD Sink can invoke

RTSP Method	From WFD Source to WFD Sink	From WFD Sink to WFD Source	From Primary Sink to Secondary Sink	From Secondary Sink to Primary Sink
OPTIONS	Required	Required	Required	Required
org.wfa.wfd1.0	Required	Required	Required	Required
SET_PARAMETER	Required	Required	Required	Required
GET_PARAMETER	Required	Required	Required	Required
SETUP	Not Allowed[1]	Required	Not Allowed[1]	Not used[2]
PLAY	Not Allowed	Required	Not Allowed	Not used[2]
TEARDOWN	Not Allowed	Required	Not Allowed	Required
PAUSE	Not Allowed	Required*	Not Allowed	Not used[2]
RECORD	Not Allowed	Not Allowed	Not Allowed	Not Allowed
DESCRIBE	Not Allowed	Not Allowed	Not Allowed	Not Allowed
REDIRECT	Not Allowed	Not Allowed	Not Allowed	Not Allowed
ANNOUNCE	Not Allowed	Not Allowed	Not Allowed	Not Allowed

Notes:

1. Not allowed during RTSP procedures to establish a WFD Session, if the WFD Device is acting as specified in the column.

2. SETUP, PAUSE, and PLAY are optional in an RTSP M2 Response message from a Secondary Sink to the Primary Sink during a coupling procedure. These SETUP, PAUSE and PLAY methods are not used from a Secondary Sink to a Primary Sink.

Example: The following RTSP exchange shows a WFD Source querying its peer WFD Sink to verify that it has WFD support.

```
Source->Sink  OPTIONS * RTSP/1.0
    CSeq: 0
    Require: org.wfa.wfd1.0

Sink->Source RTSP/1.0 200 OK
    CSeq: 0
```

```
    Public: org.wfa.wfd1.0, SET_PARAMETER, GET_PARAMETER
```

Conversely, below is an example of a WFD Sink querying its peer WFD Source to verify that it has WFD support.

```
Sink->Source  OPTIONS * RTSP/1.0
    CSeq: 0
    Require: org.wfa.wfd1.0
Source->Sink  RTSP/1.0 200 OK
    CSeq: 0
    Public: org.wfa.wfd1.0, SET_PARAMETER, GET_PARAMETER, SETUP, PLAY, PAUSE, TEARDOWN
```

6.2.2 GET_PARAMETER

In response to a GET_PARAMETER Request from a WFD Source, a WFD Sink shall respond with the current value of the requested parameter or parameters. A WFD Device shall be able to parse RTSP parameters in any order. A WFD Sink shall support responding to a GET_PARAMETER Request at any time after a successful RTSP M2 message exchange.

A GET_PARAMETER without body can be used for keep-alive function. See sections 6.5.1 and 6.4.16.

The GET_PARAMETER Response may include responses to the parameters in a different order from that which was requested in the corresponding GET_PARAMETER Request. In addition, the responder may ignore parameters in the request that the responder does not recognize.

6.2.3 SET_PARAMETER

In response to a SET_PARAMETER Request from a WFD Device, a peer WFD Device shall either:

a. update all of the values of the specified parameters to the provided values and respond with an RTSP SET_PARAMETER Response message containing a status code of RTSP OK, or

b. respond with an RTSP SET_PARAMETER Response message containing a status code “303 See Other” in the message header. The SET_PARAMETER Response message body shall contain the parameter or parameters that were not set successfully, followed by a separator of “: ” and one or more reason codes, which inform the sender of the RTSP SET_PARAMETER Request of the reason for the failure. The reason code shall be separated by “, ”.

The ABNF of the response is as follows:

```
line-at-message-body     = “name-of-failed-parameter:” SP reason-code *(“,” SP reason-code) CRLF
Reason-code              = 3*3DIGIT; see Table 96
```

For example, if the wfd-video-formats cannot be set at the Primary Sink because the WFD Source requested both an unsupported resolution and an unsupported codec profile, the message body of the response is as follows;

```
Wfd_video_formats: 415, 457
```

The reason codes to be used in the message body are specified in Table 96. These reason codes are WFD specific and are not identical to the status code specified in RTSP.

Table 96. Reason Code for RTSP M4 Response in WFD

Reason Code	WFD Detail
400	RTSP syntax violation
401	Unacceptable RTP port being used
404	The requester attempts setting a parameter that the responder did not advertise in Capability Negotiation or cannot find a parameter
415	The requester attempts to use an unsupported A/V format on the responder

Reason Code	WFD Detail
451	The receiver of the request does not understand a parameter
453	The responder cannot set a given parameter because expected bit rate for the requested parameter may exceed available bit rate
457	The responder does not support a profile and/or level for a given codec
458	A responder cannot update a requested parameter because it is not allowed in this specification
465	An unknown reason other than the ones specified here
552	WFD R2 Source does not accept suggested resolution change
553	WFD R2 Source does not accept suggested fps change

In this case (b) above, the recipient of the SET_PARAMETER Request shall update the parameters that are acceptable. The transmitter of the SET_PARAMETER Request may send a second SET_PARAMETER Request after modifying parameters based on the error codes returned by the recipient in its SET_PARAMETER Response. An example is shown in Table 123 in Appendix E.2.

6.2.4 SETUP

RTSP M6 Request and Response messages for SETUP shall include Transport header as specified herein.

During initial WFD Session establishment, the RTSP SETUP Request shall use the values from the wfd-client-rtp-ports parameter (specified in section 6.1.10) for the transport. When switching to TCP or UDP any time after the initial WFD Session establishment, the WFD R2 Sink indicates the profile and the rtp-port0 or rtp-port1 values applicable for the respective transport The ABNF for Transport header is:

```
Transport          = “Transport:” SP profile port-numbers CRLF
profile            = “RTP/AVP/UDP;unicast;” / “RTP/AVP/TCP;unicast;”)
port-numbers       = client-port [“;” server-port]; client-port only at M6 request message,
                     (client-port “;” server-port) at M6 response message
client-port        = “client_port=” (rtp-port0 / rtp-port1) [“-” IPPORT]; rtp-port0 from a
                     Primary Sink, rtp-port1 from a Secondary Sink. IPPORT here is rtp-port0
                     (or rtp-port1) plus 1 to be used for RTCP. [“-” IPPORT] can only appear
                     in M6 request if the WFD Sink wants to use optional RTCP. [“-” IPPORT]
                     can only appear in M6 response if the WFD Source accepts to use optional
                     RTCP.
server-port        = “server_port=” IPPORT [“-” IPPORT]; server’s UDP port number for RTP.
                     Optional IPPORT is server’s UDP port number for RTP plus 1 to be used
                     for optional RTCP. [“-” IPPORT] can only appear in M6 response if the
                     WFD Source accepts to use optional RTCP
```

Note : MPEG2-TS (MP2T) described at section 5.7 in RFC3551 [6] is used with the AVP/ profile in this specification. Other profiles described at other sections in [6] are not relevant to this specification.

The profile parameter in the Transport header shall be set to RTP/AVP/UDP;unicast; when sending an M6 Request or Response during the initial WFD Session establishment. When the WFD R2 Sink sends an M6 Request after receiving an M4 Request containing the wfd2-transport-switch parameter from the WFD R2 Source, the profile parameter in the Transport header shall be set based on the profile indicated in the M4 Request as described in section 4.10.6.

The WFD Source can determine and inform the timeout value to the WFD Sink. In the RTSP M6 Response message, the WFD source may set a timeout value in units of 1 second. See section 6.5.1 for more detail.

6.2.5 PLAY

The RTSP PLAY Request shall not include a “Range” parameter.

6.2.6 PAUSE

The RTSP PAUSE Request shall not include a “Range” parameter.

6.2.7 TEARDOWN

The RTSP TEARDOWN Request has no additional WFD specific requirements.

6.3 RTSP Parameters

This section defines WFD specific parameters that are used within RTSP SET_PARAMETER and GET_PARAMETER methods. All parameters referenced are those that describe or control capabilities of the WFD Sink, except for the wfd-connector-type, wfd-standby, wfd-idr-request, wfd-uibc-capability and wfd-uibc-setting, which may be sent by the WFD Sink to set parameter in the WFD source.

The URL specified by the WFD Source in GET_PARAMETER and SET_PARAMETER for accessing WFD Sink parameters shall be “rtsp://localhost/wfd1.0”. Conversely, the WFD Sink shall always reference the WFD Source in RTSP messages using the URL given to it by the WFD Source in the wfd-presentation-url parameter.

Table 97 contains a summary of RTSP parameters used in this specification:

Table 97. Summary of RTSP parameters

Parameter name	Parameter description	Method	Definition	Usage
wfd-audio-codecs	Audio format(s) supported by the WFD R1 Sink.	GET	6.1.2	Mandatory in RTSP M3 Request and Response
	Audio format selected for the WFD Session between WFD R1 devices.	SET		Mandatory in RTSP M4 Request sent by a WFD R2 Source if the stream includes audio
wfd-video-formats	video format(s) support by the WFD R1 Sink.	GET	6.1.3	Mandatory in RTSP M3 Request and Response
	Video format selected for WFD Session between WFD R1 devices.	SET		Mandatory in RTSP M4 Request if the stream includes 2D video.
wfd-3d-formats	3D formats supported by the WFD Sink.	GET	6.1.4	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
	3D format selected for the WFD Session.	SET		Mandatory in RTSP M4 Request if the stream includes 3D video.
wfd-content-protection	The HDCP system 2.x support/control port.	GET	6.1.5	Mandatory in RTSP M3 Request if the WFD Device supports HDCP system 2.x. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
wfd-display-edid	Available EDID information of display attached to the WFD Sink.	GET	6.1.6	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
wfd-coupled-sink	Coupled WFD Sink information.	GET	6.1.7	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
wfd-trigger-method	Trigger to cause a SETUP, PLAY, PAUSE or TEARDOWN message to be initiated by WFD Sink to WFD Source.	SET	6.1.8	Mandatory in RTSP M5 Request.
wfd-presentation-url	Accompanies wfd-trigger-method for SETUP trigger. Defines the URI that shall be used for SETUP Request from the WFD Sink.	SET	6.1.9	Mandatory in RTSP M4 Request when WFD Session has not been established.

Parameter name	Parameter description	Method	Definition	Usage
wfd-client-rtp-ports	RTP port(s) the WFD Sink(s) listen on.	GET	6.1.10	Mandatory in RTSP M3 Request when WFD Session has not been established.
		SET		Mandatory in RTSP M4 Request when WFD Session has not been established.
wfd-route	Route audio from Primary Sink to Secondary Sink or vice-versa.	SET	6.1.11	Mandatory in RTSP M10 Request if RTSP M10 is supported.
wfd-I2C	Supports I2C commands and port number.	GET	6.1.12	Optional in RTSP M3 Request. Mandatory in M3 Response if RTSP M3 Request includes it.
wfd-av-format-change-timing	Signals timing related to AV format change to the WFD Sink.	SET	6.1.13	Mandatory in RTSP M4 Request when it is sent during WFD Session.
wfd-uibc-capability	UIBC capability supported.	GET	6.1.15	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
	Select UIBC to be used.	SET		Optional in RTSP M4 Request. Mandatory in RTSP M14 Request if RTSP M14 is supported.
wfd-uibc-setting	Enable or disable the UIBC.	SET	6.1.16	Optional in RTSP M4 Request. Mandatory in RTSP M15 Request if RTSP M15 is supported.
wfd-standby-resume-capability	Indicate the support for standby and resume control using RTSP.	GET	6.1.17	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
wfd-standby	Sender is entering WFD Standby mode.	SET	6.1.18	Mandatory in RTSP M12 Request if RTSP M12 is supported.
wfd-connector-type	WFD Source uses this parameter to obtain the connector type currently active on the WFD Sink.	GET	6.1.19	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
	The WFD Sink uses this parameter to inform the WFD Source of a connector type change and report the currently active connector type.	SET		Mandatory in RTSP M11 Request if RTSP M11 is supported.
wfd-idr-request	WFD Sink requests an IDR picture from the WFD Source.	SET	6.1.20	Mandatory in RTSP M13 Request if sending RTSP M13 Request is supported by the Primary Sink.
wfd2-audio-codecs	audio codec(s) and format(s) support by the WFD R2 Sink.	GET	6.1.21	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
	Audio codec and format selected for WFD Session between WFD R2 devices.	SET		Mandatory in RTSP M4 Request sent by a WFD R2 Source if the stream includes audio.
wfd2-video-formats	video codec(s) and format(s) support by the WFD R2 Sink.	GET	6.1.22	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
	Video codec and format selected for WFD Session between WFD R2 devices.	SET		Mandatory in RTSP M4 Request sent by a WFD R2 Source if the stream includes 2D video.
wfd2-aux-stream-formats	Codec(s) and format(s) supported by the WFD R2 Sink for auxiliary streams.	GET	6.1.23	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
	Codec(s) and format(s) selected by the WFD R2 Source for auxiliary streams.	SET		Optional in RTSP M4 Request when the WFD R2 Source requires to send auxiliary streams.

Parameter name	Parameter description	Method	Definition	Usage
wfd2-direct-streaming-mode	Indicates if direct streaming (non-transcoding) mode is active or not from a WFD Source to a WFD R2 Sink.	SET	6.1.24	Optional in RTSP M4 Request if the stream is sent without re-encoding at the WFD R2 Source.
Wfd2-transport-switch	A WFD R2 Source uses this parameter to switch the transport for streaming over the TCP or UDP transport	SET	6.1.25	Optional in RTSP M4 Request Optional in M4 Response if the RTSP M4 Request includes it
wfd2-buffer-length	A WFD R2 Source uses this parameter to query the buffer size supported by the WFD R2 Sink or to set the buffer size required during AV streaming.	GET	6.1.26	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
		SET		Optional in RTSP M4 Request if RTSP M3 Response includes it. Optional in M4 Response if the RTSP M4 Request includes it.
wfd2-audio-playback-status	The WFD R2 Source uses this parameter to query the status of the audio playback to include the amount of audio data currently buffered and the PTS value of the audio currently being played at the WFD R2 Sink.	GET	6.1.27	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
wfd2-video-playback-status	The WFD R2 Source uses this parameter to query the status of the video playback to include the amount of video data currently buffered and the PTS value of the video currently being played at the WFD R2 Sink.	GET	6.1.28	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.
wfd2-cta-audio-infoframe	Channel layout and assignment, audio format	SET	6.1.30	Mandatory in RTSP M4 Request sent by a WFD R2 Source if wfd2-audio-codecs is CTA.
wfd2-cta-datablock-collection	Speaker location, room configuration, and extended codec details supported in the WFD R2 Sink	GET	6.1.29	Optional in RTSP M3 Request. Mandatory in RTSP M3 Response if RTSP M3 Request includes it.

6.4 RTSP Messages

This section (and sub-section) defines RTSP messages that are exchanged between WFD devices for WFD Capability Negotiation, Coupling, WFD Session Establishment, and WFD Session Management.

The transmission of an RTSP M1 Request message from a WFD Source initiates WFD Capability Negotiation, as described in section 4.6. Upon a successful exchange of RTSP M4 Request and Response messages, WFD Capability Negotiation is successfully completed.

The transmission of an RTSP M5 Request message from a WFD Source initiates a WFD Session Establishment, as described in section 4.8. Upon a successful exchange of RTSP M6 Request message and Response messages, an RTSP session is established as described in [20]. Upon the first successful exchange of RTSP M7 Request message and Response messages just after an RTSP session establishment, a WFD Session is established as described in section 4.8. A WFD Session ends when the WFD Sink sends a TEARDOWN, when an RTSP timeout occurs as specified in section 6.1.5, or when an error condition occurs as described later in this section.

When both a Primary and a Secondary Sink are in a WFD Session with a WFD Source for a Coupled Sink Operation, the WFD Source shall perform the sequence described in section 4.9.6, once with the Primary Sink, and once with the Secondary Sink. A WFD Session is considered as established when the first exchange of RTSP M7 messages between both the WFD Primary Sink and WFD Secondary Sink has been completed successfully with the WFD Source. Note that RTSP sessions are individually established, i.e., the RTSP session between the WFD Source and the Primary Sink is established when the exchange of RTSP M6 Request and Response messages between them is completed, and the RTSP session between the WFD Source and the Secondary Sink is established when the exchange of RTSP M6 Request and Response messages between them is completed.

Upon the establishment of the TCP connection, the WFD Source shall send an RTSP M1 Request message to the WFD Sink. The WFD Sink shall respond with an RTSP Response message and then shall send an RTSP M2 Request message to the WFD Source. The messages shall be transmitted in sequential order as described here.

The RTSP procedures are defined as all RTSP message exchanges listed in this specification, including RTSP messages before establishing RTSP session. An RTSP message exchange is defined as sending an RTSP Request message and receiving a corresponding RTSP Response message. A successful RTSP message exchange is one where the corresponding RTSP Response message contains a status code of RTSP OK. A failed RTSP message exchange is one where the corresponding RTSP Response message contains a status code that is not a status code of RTSP OK. See 7.1.1 in [20] for details.

When the WFD Device is unable to parse a request or unable to accept a request in an RTSP Request message, the WFD Device receiving the RTSP Request message shall send an RTSP Response message that includes a corresponding RTSP status code indicating an associated error.

If the WFD Device that sent the RTSP Request message receives an RTSP Response message with an RTSP status code indicating an error from the peer WFD Device, the WFD Device shall not execute any behavior that is not accepted by the peer WFD Device as indicated in a parameter (or parameters) in the received RTSP Response message. Depending on the returned value of a status code other than RTSP OK, the message exchange may be retried with a different set of parameters or the RTSP procedures may be aborted, and the RTSP session may be aborted if it has been established.

If an RTSP Request or RTSP Response does not match the syntax specified in this specification, the recipient of the RTSP message shall ignore the RTSP message and abort the RTSP procedures, and the RTSP session shall be aborted if the RTSP session has been established.

When aborting the RTSP procedures before a successful exchange of an RTSP M6 message exchange (i.e., RTSP session has not been established yet), a WFD Sink should not send an RTSP M8 Request message, nor should a WFD Source send an RTSP M5 Request message containing wfd-trigger-method parameter with the trigger method set to TEARDOWN.

If the RTSP procedures are aborted before establishing an RTSP session, the WFD Source and the WFD Sink stay in an RTSP Init state, and the WFD Source may send an RTSP M1 Request message to a WFD Sink. When aborting the RTSP procedures after a successful exchange of RTSP M6 messages (i.e., RTSP session has been established), a WFD Sink shall send an RTSP M8 Request message, or a WFD Source shall send an RTSP M5 Request message containing wfd-trigger-method parameter with the trigger method set to TEARDOWN.

Timeout rules are specified in section 6.5.

Examples of RTSP messages are shown in Appendix E.

Table 98 lists the RTSP messages and their identifiers defined in this specification.

Table 98. List of defined RTSP messages and their identifiers

Id	Requester	Payload	Behavior	Description	RTSP State
M1	WFD Source	6.2.1	6.4.1	Query WFD Sink’s options	Init
M2	WFD Sink	6.2.1	6.4.2	Query WFD Source’s options	Init
M3	WFD Source	6.1.2 - 6.1.7, 6.1.10, 6.1.12, 6.1.15, 6.1.17, 6.1.19	6.4.3	Query WFD Sink’s capabilities	Init (for WFD Capability Negotiation) Playing (for WFD Capability Re-negotiation)
M4	WFD Source	6.1.2 - 6.1.7, 6.1.9, 6.1.10, 6.1.13 - 6.1.16	6.4.4	Set WFD Sink’s parameters	Init (for WFD Capability Negotiation) Playing (for WFD capability Re-negotiation)
M5	WFD Source	6.1.8	6.4.5	Trigger WFD Sink to issue a SETUP, PLAY, TEARDOWN, or PAUSE Request.	Init (for SETUP) Ready (for PLAY and TEARDOWN) Playing (for PAUSE and TEARDOWN)

Id	Requester	Payload	Behavior	Description	RTSP State
M6	WFD Sink	6.2.4	6.4.6	Send SETUP Request to WFD Source.	Ready
M7	WFD Sink		6.4.7	Send PLAY Request to WFD Source. WFD Source begins audio and/or video streaming.	Ready (for initiating streaming and for returning from pause or WFD Standby)
M8	WFD Sink		6.4.8	Send TEARDOWN Request to WFD Source	Playing
M9	WFD Sink		6.4.9	Send PAUSE Request to WFD Source. WFD Source pauses the audio video stream(s)	Playing
M10	WFD Sink	6.1.11	6.4.10	Send RTSP SET_PARAMETER with wfd-route to change the WFD Sink at which audio is rendered. Applies only when both a Primary and a Secondary Sinks are in WFD Session with a WFD Source.	Playing (optional)
M11	WFD Sink	6.1.19	6.4.11	Send RTSP SET_PARAMETER with wfd-connector-type to indicate change of active connector type, when the WFD Source and the WFD Sink support content protection.	Playing (optional)
M12	WFD Source WFD Sink	6.1.18	6.4.12	Send RTSP SET_PARAMETER with wfd-standby to indicate that the sender is entering WFD Standby mode.	Playing (optional)
M13	WFD Sink	6.1.20	6.4.13	Send RTSP SET_PARAMETER with wfd-idr-request to request IDR refresh	Playing (optional at the Primary Sink to send request, and mandatory at the WFD Source to respond)
M14	WFD Source WFD Sink	6.1.15	6.4.14	Send RTSP SET_PARAMETER with wfd-uibc-capability to select UIBC to be used.	Playing (optional)
M15	WFD Source WFD Sink	6.1.16	6.4.15	Send RTSP SET_PARAMETER with wfd-uibc-setting to enable or disable the UIBC.	Playing (optional)
M16	WFD Source		6.4.16	Send GET_PARAMETER to confirm active RTSP session	Ready Playing

6.4.1 RTSP M1 Message

A WFD Source shall send an RTSP M1 Request to a WFD Sink to begin the RTSP procedures and a WFD Capability Negotiation. The RTSP M1 Request shall contain an RTSP OPTIONS Request. A WFD Sink shall respond with an RTSP M1 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. If the RTSP M1 Response message contains a status code of RTSP OK, the RTSP M1 Response message shall contain an RTSP OPTIONS Response.

6.4.2 RTSP M2 Message

After sending an RTSP M1 Response including a status code of RTSP OK to the WFD Source, the WFD Sink shall send an RTSP M2 Request to the WFD Source. The RTSP M2 Request shall contain an RTSP OPTIONS Request. The WFD Source shall respond with an RTSP M2 Response indicating an appropriate status code specified in section 7.1.1 of [20].

If the RTSP M2 Response message contains a status code of RTSP OK, the RTSP M2 Response message shall contain an RTSP OPTIONS Response.

6.4.3 RTSP M3 Message

After sending an RTSP M2 Response including a status code of RTSP OK to the WFD Sink, the WFD Source shall send an RTSP M3 Request to the WFD Sink to query the WFD Sink's attributes and capabilities. The RTSP M3 Request shall contain an RTSP GET_PARAMETER Request (see section 6.2.2). The parameters that may be requested are listed in section 6.3. The WFD Sink shall respond with an RTSP M3 Response indicating an appropriate status code specified in section 7.1.1 of [20]. If the RTSP M3 Response message contains a status code of RTSP OK, the RTSP M3 Response shall contain the values of the requested parameters.

When an optional parameter is included in the RTSP M3 Request message from the WFD Source, it implies that the WFD Source supports the optional feature corresponding to the parameter.

The WFD Sink shall be able to parse all the RTSP parameters that are listed in section 6.1 in the RTSP M3 Request message. If the WFD Sink receives an RTSP M3 Request including any of these parameters, it shall respond to them in the RTSP M3 Response.

The WFD Sink shall support responding to an RTSP M3 Request at any time after a successful RTSP M2 message exchange. The WFD Source may query all parameters at once with a single RTSP M3 Request message or may send separate RTSP M3 Request messages.

The WFD Sink shall only respond with formats and settings that it can accept in the following RTSP M4 message exchange.

If the WFD Sink has indicated support for the HDCP system 2.x (see section 6.1.5) in the most recent RTSP M3 Response message and if the HDCP 2.x authentication key exchange has not been successfully completed for a WFD Session, the WFD Source and the WFD Sink shall start the HDCP 2.x authentication key exchange upon successful completion of the RTSP M3 message exchange. The WFD Sink should start listening upon transmission of an RTSP M3 Response message, and the WFD Source should start sending AKE_Init upon receipt of the RTSP M3 Response message.

The WFD Sink shall only respond with formats and settings that the WFD Sink can accept in the following RTSP M4 message exchange.

6.4.4 RTSP M4 Message

If a WFD Session has not been established, after receiving an RTSP M3 Response including a status code of RTSP OK, the WFD Source shall send an RTSP M4 Request to the WFD Sink to set parameters for the WFD Sink. The WFD Sink shall respond with an RTSP M4 Response indicating an appropriate status code specified in section 7.1.1 of [20].

The WFD Sink shall support responding to an RTSP M4 Request at any time after a successful RTSP M3 message exchange. The WFD Source may set parameters all at once with a single RTSP M4 Request message or may send separate RTSP M4 Request messages.

The format of the M4 Request message varies depending on the WFD Session:

1. If the WFD R1 Source is trying to initiate the establishment of an audio-only WFD Session with the WFD R1 Sink, the RTSP M4 Request message (or a series of RTSP M4 Request messages) shall include the wfd-audio-codecs parameter and shall not include any of the following parameters: wfd-video-formats, wfd-3d-formats, wfd2-video-formats or wfd2-aux-stream-formats.

   If the WFD R2 Source is trying to initiate the establishment of an audio-only WFD Session with the WFD R2 Sink, the RTSP M4 Request message (or a series of RTSP M4 Request messages) shall include the wfd2-audio-codecs parameter and shall not include any of the following parameters: wfd-video-formats, wfd-3d-formats, wfd2-video-format, or wfd2-aux-stream-formats.

2. If the WFD R1 Source is trying to initiate the establishment of a video-only WFD Session with the WFD R1 Sink, the RTSP M4 Request message (or a series of RTSP M4 Request messages) shall not include the wfd-audio-

codecs parameter nor wfd2-audio-codecs parameter and shall include only one of the following parameters: wfd-video-formats or wfd-3d-formats.

If the WFD R2 Source is trying to initiate the establishment of a video-only WFD Session with the WFD R2 Sink, the RTSP M4 Request message (or a series of RTSP M4 Request messages) shall not include the wfd-audio-codecs parameter nor the wfd2-audio-codecs parameter and shall include only one of the following parameters: wfd-3d-formats, or wfd2-video-formats.

3. If the WFD R1 Source is trying to initiate the establishment of an audio and video WFD Session with an R1 Primary Sink, the RTSP M4 Request message (or a series of RTSP M4 Request messages) shall include the wfd-audio-codecs parameter and only one of the following parameters: wfd-video-formats or wfd-3d-formats.

   If the WFD R2 Source is trying to initiate the establishment of an audio and video WFD Session with an R2 Primary Sink, the RTSP M4 Request message (or a series of RTSP M4 Request messages) shall include the wfd2-audio-codecs parameter and only one of the following parameters: wfd-3d-formats, or wfd2-video-formats.

4. If the WFD R1 Source is trying to initiate the establishment of an audio and video WFD Session with an R1 Primary Sink and an R1 Secondary Sink during a Coupled Sink Operation:

   a. If both audio and video payloads are destined for the R1 Primary Sink, the RTSP M4 Request message (or a series of RTSP M4 Request messages) to the R1 Primary Sink shall include the wfd-audio-codecs parameter and only one of the following parameters: wfd-video-formats or wfd-3d-formats.

   b. If video payload is destined for the R1 Primary Sink and audio payload is destined for the R1 Secondary Sink, then:
   - The RTSP M4 Request message (or a series of RTSP M4 Request messages) to the R1 Primary Sink shall not include the wfd-audio-codecs parameter nor the wfd2-audio-codecs parameter and shall include one of the following parameters: wfd-video-formats or wfd-3d-formats.

      - The RTSP M4 Request message (or a series of RTSP M4 Request messages) to the R1 Secondary Sink shall include the wfd-audio-codecs parameter and shall not include any of the following parameters: wfd-video-formats or wfd-3d-formats.

   If the WFD R2 Source is trying to initiate the establishment of an audio and video WFD Session with an R2 Primary Sink and an R2 Secondary Sink during a Coupled Sink Operation:

   c. If both audio and video payloads are destined for the R2 Primary Sink, the RTSP M4 Request message (or a series of RTSP M4 Request messages) to the R2 Primary Sink shall include the wfd2-audio-codecs parameter and only one of the following parameters: wfd-3d-formats or wfd2-video-formats.

   d. If video payload is destined for the R2 Primary Sink and audio payload is destined for the R2 Secondary Sink, then:
   - The RTSP M4 Request message (or a series of RTSP M4 Request messages) to the R2 Primary Sink shall not include the wfd2-audio-codecs parameter and shall include one of the following parameters: wfd-3d-formats or wfd2-video-formats.

      - The RTSP M4 Request message (or a series of RTSP M4 Request messages) to the R2 Secondary Sink shall include the wfd2-audio-codecs parameter and shall not include any of the following parameters: wfd-3d-formats, wfd2-video-format, or wfd2-aux-stream-formats.

5. If the WFD R2 Source is trying to initiate the establishment of a video-only WFD R2 Session with multiple monitors in a multi-mon capable WFD R2 Sink:

   a. it shall include at least one wfd2-desired-port-number parameter in an M4 Request in order to determine monitor-specific capabilities of the monitor on the specified port. An M4 Request containing wfd2-desired-port-number parameter shall not contain additional RTSP parameters.

   b. it shall include only the wfd2-port-numbers-for-stream parameter in its last M4 Request before sending an M5 Request (Trigger SETUP). An M4 Request containing wfd2-port-numbers-for-stream parameter shall not contain additional RTSP parameters.

The RTSP M4 Request message may include a wfd-uibc-setting parameter during the WFD Capability Negotiation if the RTSP M4 Request message contains a wfd-uibc-capability parameter.

If a WFD Sink has indicated unsupported for feature(s) in its RTSP M3 Response message, then the WFD Source shall not include RTSP parameter(s) that are related to those unsupported feature(s). It is highly recommended that the WFD Sink should ignore an RTSP parameter that is unknown or is not supported.

Based on the wfd-client-rtp-ports parameter in the M3 Response and depending on the WFD Session being setup, the WFD Source determines the configuration of the MPEG2-TS stream(s) to be used in the WFD Session and the WFD Source shall include the corresponding wfd-client-rtp-ports parameter in the RTSP M4 Request message sent to the WFD Sink. How to set this parameter is defined in Table 90 under section 6.1.10.

The RTSP M4 Request message that is for WFD Capability Negotiation shall contain the wfd-presentation-url parameter (specified in section 6.1.9) that describes the Universal Resource Identifier (URI) to be used in the RTSP Setup Request (RTSP M6 Request) in order to setup the WFD Session. The wfd-presentation-url specifies the URI that a WFD Sink shall use in an RTSP M6 Request message to a WFD Source. The values of wfd-url0 and wfd-url1 fields specified in this parameter correspond to the values of rtp-port0 and rtp-port1 field in the wfd-client-rtp-ports parameter in the RTSP M4 Request message from the WFD Source to the WFD Sink at the end of the Capability Negotiation. The WFD Sink uses information in this parameter in RTSP SETUP Request (RTSP M6 Request) message.

If the RTSP M4 Request message is sent to the WFD Sink to change one or more parameters in wfd-audio-codec, wfd2-audio-codecs, wfd-video-formats, wfd2-video-formats and/or wfd-3d-formats parameter(s) used in the WFD Session, it shall include a wfd-av-format-change-timing (specified in section 6.1.13) parameter.

6.4.5 RTSP M5 Message

The M5 Request message is used by a WFD Source to trigger an RTSP session establishment when an RTSP session has not been established, or to trigger play, pause, or teardown during an RTSP session.

If an RTSP session has not been established, after receiving an RTSP M4 Response with a status code of RTSP OK, the WFD Source shall send an RTSP M5 Request message containing wfd-trigger-method parameter with the trigger method set to SETUP to the WFD Sink to indicate that the WFD Sink is requested to send an RTSP M6 Request message. The WFD Sink shall respond with an RTSP M5 Response indicating an appropriate status code specified in section 7.1.1 of [23].

Once an RTSP session has been established, the WFD Source may send an RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to PLAY to the WFD Sink to indicate that the WFD Sink is requested to send an RTSP M7 Request message. The WFD Sink shall respond with an RTSP M5 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. The WFD Source in an RTSP Playing state should not send an RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to PLAY to the WFD Sink.

Once a WFD Session has been established, the WFD Source may send an RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to PAUSE to the WFD Sink to indicate that the WFD Sink is requested to send an RTSP M9 Request message. The WFD Sink shall respond with an RTSP M5 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. The WFD Source in an RTSP Ready state should not send an RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to PAUSE to the WFD Sink.

Once an RTSP session has been established, the WFD Source may send an RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to TEARDOWN to the WFD Sink to indicate that the WFD Sink is requested to send an RTSP M8 Request message. The WFD Sink shall respond with an RTSP M5 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

In general, an RTSP M5 Request message is used to trigger the recipient to send an RTSP message corresponding to the parameter contained in the RTSP M5 Request message. Figure 27 illustrates the message exchange between WFD devices.

In the event that a WFD Source receives an RTSP message requesting the WFD Source to perform an action that contradicts a previously initiated TRIGGER operation by the WFD Source, then the WFD Source shall obey the action indicated in the RTSP Request message from the WFD Sink.

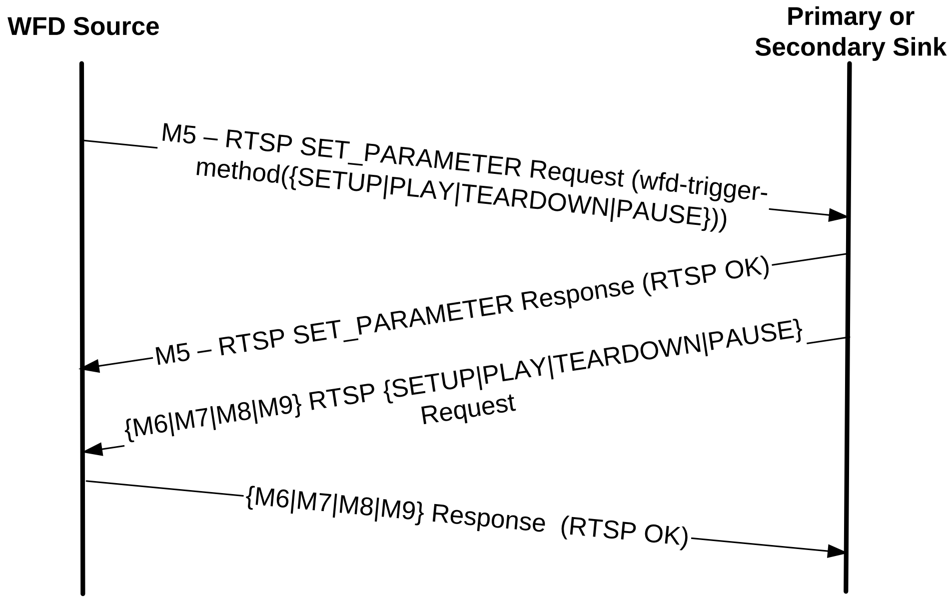


Figure 27. Triggering the WFD Sink to send an RTSP Request message

6.4.6 RTSP M6 Message

After sending an RTSP M5 Response message with a status code of RTSP OK as a response to the RTSP M5 Request message containing a wfd-trigger-method parameter with the trigger method set to SETUP, the WFD Sink shall send an RTSP M6 Request message to the WFD Source. The RTSP M6 request message shall contain an RTSP SETUP Request. The request shall use information indicated in a wfd-presentation-url (specified in section 6.1.9) in the RTSP M4 Request message as an URI for SETUP (this does not mean that the RTSP M6 Request shall include identical text string in the wfd-presentation-url parameter in an RTSP M4 Request message). The WFD Source shall respond with an RTSP M6 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. When the WFD Source sends the RTSP M6 Response message indicating a status code of RTSP OK, a session id shall be included.

RTSP M6 Request and Response messages include Transport header as specified in section 6.2.4.

To apply HDCP 2.x encryption from the beginning of the stream after verifying that the WFD Source and the WFD Sink support the HDCP system 2.0/2.1 or higher version in the RTSP M3 Request/Response messages (see section 6.1.5), the WFD Sink shall transmit an RTSP M7 Request message only after successful completion of the HDCP 2.x authentication and key exchange as specified in clause 2 of the corresponding specification references indicated in Appendix F that maps to the HDCP version in use. For higher version than HDCP 2.2, refer to corresponding specification released by DCP LLC.

6.4.7 RTSP M7 Message

A WFD Sink sends an RTSP M7 Request message to the WFD Source in the following cases:

1. While in the process of establishing a WFD Session, after receiving an RTSP M6 Response message with a status code of RTSP OK, the WFD Sink shall send an RTSP M7 Request message to the WFD Source.
2. Once a WFD Session has been established, in response to the receipt of an RTSP M5 Request message from the WFD Source containing a wfd-trigger-method parameter with the trigger method set to PLAY, the WFD Sink sends an RTSP M5 Response message to the WFD Source. If the RTSP M5 Response message included a status code of RTSP OK, the WFD Sink shall send an RTSP M7 Request message to the WFD Source.
3. Once a WFD Session has been established, if the WFD Sink intends to resume streaming from the WFD Source, the WFD Sink sends an RTSP M7 Request message to the WFD Source.

The RTSP M7 Request message contains an RTSP PLAY Request.

The WFD Source shall respond with an RTSP M7 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

If the RTSP M7 Response message includes a status code of RTSP OK, the WFD Source shall start/resume transmitting the audio/video stream to the RTP client port numbers specified in the RTSP M6 Request message. If the WFD Source starts/resumes transmitting an MPEG2-TS that contains a video elementary stream, the first video frame shall be an IDR picture with SPS and PPS. Also, if the WFD Source starts/resumes transmitting an MPEG2-TS that contains an audio elementary stream, the first Access Unit (AU) for audio shall be a new AU, and the first audio frame shall be a new frame if the audio format is not LPCM. In LPCM case, a WFD Sink can start rendering form any point and it is not necessary to consider frame boundary.

If the WFD Source in an RTSP Playing state receives an RTSP M7 Request message, the WFD Source should send the RTSP M7 Response message with status code “406” (means “not acceptable”) and reason phrase of “in-play-state”.

6.4.8 RTSP M8 Message

A WFD Sink sends an RTSP M8 Request message to the WFD Source in the following cases:

1. Once the RTSP session has been established, in response to the receipt of an RTSP M5 Request message from the WFD Source containing wfd-trigger-method parameter with the trigger method set to TEARDOWN, the WFD Sink sends an RTSP M5 Response message to the WFD Source. If the RTSP M5 Response message included a status code of RTSP OK, the WFD Sink shall send an RTSP M8 Request message to the WFD Source.
2. Once the RTSP session has been established, if the WFD Sink intends to end the RTSP session with the WFD Source, the WFD Sink sends an RTSP M8 Request message to the WFD Source.

The RTSP M8 Request message contains an RTSP TEARDOWN Request.

If an RTSP M8 Request message is received during the RTSP procedures, the WFD Source shall send an RTSP M8 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

If the WFD Source sends an RTSP M8 Response message with a status code of RTSP OK, the WFD Source shall abort the RTSP procedures, and shall stop the audio/video streaming and terminate the corresponding RTP session(s) if exists.

If the RTSP M8 Request and Response messages are correctly exchanged with a status code of RTSP OK, the WFD Source and WFD Sink shall release corresponding resources committed to the RTP session (if exists) and RTSP procedures. After the teardown, in order to establish a new WFD Session, the WFD devices shall start with WFD Capability Negotiation (see section 4.6) and WFD Session establishment (see section 4.8) procedures.

6.4.9 RTSP M9 Message

Once a WFD Session has been established and the WFD Sink has started decoding the corresponding stream received from the WFD Source, the WFD Sink sends the RTSP M9 Request message to the WFD Source in the following cases:

1. In response to the receipt of an RTSP M5 Request message from the WFD Source containing wfd-trigger-method parameter with the trigger method set to PAUSE, the WFD Sink sends an RTSP M5 Response message to the WFD Source. If the RTSP M5 Response message included a status code of RTSP OK, the WFD Sink shall send an RTSP M9 Request message to the WFD Source.
2. If the WFD Sink intends to pause the current stream being received from the WFD Source, the WFD Sink sends an RTSP M9 Request message to the WFD Source.

The RTSP M9 Request message contains an RTSP PAUSE Request.

If an RTSP M9 Request message is received during the WFD Session with “PLAY” state, the WFD Source shall respond with an RTSP M9 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

If the WFD Source sends an RTSP M9 Response message with a status code of RTSP OK, the WFD Source shall stop transmission of the content stream to the RTP client port numbers specified in the RTSP M6 Response message.

If the WFD Source in an RTSP Ready state receives an RTSP M9 Request message, the WFD Source should send an RTSP M9 Response message with status code “406” (means “not acceptable”) and reason phrase if “in-pause-state”.

6.4.10 RTSP M10 Message

During a WFD Session for Coupled Sink Operation (see section 4.9), the Primary Sink may send an RTSP M10 Request message to the WFD Source in order to change the WFD Sink where the audio stream corresponding to the WFD Session is rendered (see section 4.10.4).

During a WFD Session for Coupled Sink Operation (see section 4.9), the Secondary Sink may send an RTSP M10 Request message to the WFD Source in order to change the WFD Sink where the audio stream corresponding to the WFD Session is rendered (see section 4.10.4).

The RTSP M10 Request message is an RTSP SET_PARAMETER Request containing a wfd-route parameter. The destination field in the wfd-route parameter identifies the WFD Sink where the audio stream is to be rendered.

If an RTSP M10 Request message is received during the WFD Session, the WFD Source shall respond with an RTSP M10 Response message indicating an appropriate status code specified in section 7.1.1 of [20] to the WFD Sink that sent an RTSP M10 Request message.

If the WFD Source sends an RTSP M10 Response message with a status code of RTSP OK, the WFD Source shall switch the destination of the audio stream to the WFD sink identified in the M10 Request message.

6.4.11 RTSP M11 Message

During a WFD Session, only if the WFD Source had indicated support for this feature by querying a wfd-connector-type parameter (specified in section 6.1.19) in the RTSP M3 Request message and the WFD Sink had indicated the support of this feature in the RTSP M3 Response message, the WFD Sink may send an RTSP M11 Request message to the WFD Source in order to inform the change of active connector type.

The RTSP M11 Request message is an RTSP SET_PARAMETER Request containing the wfd-connector-type parameter.

If an RTSP M11 Request message is received during the WFD Session, the WFD Source shall respond with an RTSP M11 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

6.4.12 RTSP M12 Message

During a WFD Session, a WFD Device may send an RTSP M12 Request message to indicate that the WFD Device is trying to enter into a WFD Standby mode. The RTSP M12 Request message shall only be sent if the paired WFD Device had indicated the support for this feature in an exchange of the RTSP M3 Request and Response messages.

The RTSP M12 Request message is an RTSP SET_PARAMETER Request containing the wfd-standby parameter.

If an RTSP M12 Request message is received during the WFD Session, the WFD Device shall respond with an RTSP M12 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

If the WFD Device receives the RTSP M12 Response message including a status code of RTSP OK, the WFD Device shall enter into a WFD Standby mode.

The WFD Device should not send an RTSP M12 Request message when it is in a WFD Standby mode.

See section 4.15 for more detail.

6.4.13 RTSP M13 Message

During a WFD Session with streaming video content, the WFD Sink may send an RTSP M13 Request message to the WFD Source in order to request the IDR picture.

The RTSP M13 Request message is an RTSP SET_PARAMETER Request containing the wfd-idr-request parameter.

If an RTSP M13 Request message is received during the WFD Session with streaming video content, the WFD Source shall respond with an RTSP M13 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

See section 4.10.5 for more detail.

6.4.14 RTSP M14 Message

During the WFD Session, the WFD Device may send an RTSP M14 Request message to establish the UIBC. The RTSP M14 Request message shall only be sent if the paired WFD Device had indicated the support for UIBC in an exchange of the RTSP M3 Request and Response messages.

The RTSP M14 Request message is an RTSP SET_PARAMETER Request containing the wfd-uibc-capability parameter. During the WFD Session and when the WFD Source tries to establish a UIBC, the WFD Source sends an RTSP M14 Request message to establish the UIBC. If the RTSP M14 Request message is received, the WFD Sink shall respond with an RTSP M14 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. If the exchange of RTSP M14 Request and Response messages is successfully completed with a status code of RTSP OK in the RTSP M14 Response message, then the UIBC is established.

During the WFD Session and when the UIBC had been established, the WFD Device may send an RTSP M14 Request message to update parameters in the UIBC. If the RTSP M14 Request message is received, the paired WFD Device shall respond with an RTSP M14 Response message indicating an appropriate status code specified in section 7.1.1 of [23]. If the exchange of RTSP M14 Request and Response messages is successfully completed with a status code of RTSP OK in the RTSP M14 Response message, then the parameters in the UIBC are updated.

The WFD Sink shall not update any UIBC parameter(s) now in use, before the next exchange of RTSP M14 Request and Response messages is successfully completed with a status code of RTSP OK in the RTSP M14 Response message, independent on which side transmits the RTSP M14 Request message.

When the WFD Source attempts to update the UIBC parameter, the wfd-uibc-capability parameter in the RTSP M14 Request message shall be compliant with the capability of the WFD Sink indicated in the most recent RTSP M3 Response message.

During a WFD Session, if previously the WFD Sink had received the RTSP M14 Response message with a status code other than RTSP OK on the RTSP M14 Request message, the WFD Sink should not send the RTSP M14 Request message that includes same UIBC parameter (e.g., input category or input type to be used) setting with previously unaccepted UIBC parameter setting by the WFD Source.

To reduce one RTSP M14 message exchange, an RTSP M4 message exchange before establishing an RTSP session may be used for establishing UIBC. The RTSP M4 Request message to establish the UIBC may contain the wfd-uibc-capability parameter. If an RTSP M4 Request message containing wfd-uibc-capability parameter is received, the WFD Sink shall respond with an RTSP M4 Response message including the same (wfd-uibc-capability parameter indicating an appropriate status code specified in section 7.1.1 of [20]. If the exchange of RTSP M4 Request and Response messages is successfully completed with a status code of RTSP OK in the RTSP M14 Response message, then the UIBC is established.

See sections 4.11, 6.1.15, and 6.1.16 for more detail.

6.4.15 RTSP M15 Message

During a WFD Session and after a successful establishment of the UIBC by an exchange of RTSP M4 or M14 Request and Response messages, a WFD Source may send an RTSP M15 Request message to enable/disable the UIBC. During a WFD Session and after a successful establishment of the UIBC by an exchange of RTSP M4 or M14 Request and Response messages, a WFD Sink may send an RTSP M15 Request message to enable/disable the UIBC.

The RTSP M15 Request message is an RTSP SET_PARAMETER Request containing the wfd-uibc-setting parameter.

On receipt of the RTSP M15 Request message, the WFD Sink shall respond with an RTSP M15 Response message indicating an appropriate status code specified in section 7.1.1 of [20]. On receipt of the RTSP M15 Request message, the WFD Source shall respond with an RTSP M15 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

If the exchange of the RTSP M15 Request and Response messages is successfully completed with a status code of RTSP OK in the RTSP M15 Response message, then the UIBC is enabled or disabled as the parameter value indicates.

See section 4.11 and 6.1.16 for more detail.

6.4.16 RTSP M16 Message

During a WFD Session, a WFD Source shall send RTSP M16 messages to the WFD Sink in order to confirm that the RTSP session is active.

The RTSP M16 Request message is an RTSP GET_PARAMETER request without a body.

If an RTSP M16 Request message is received during a WFD Session, the WFD Sink shall respond with an RTSP M16 Response message indicating an appropriate status code specified in section 7.1.1 of [20].

See section 6.5.1 for usage of an RTSP M16 message.

6.5 RTSP Timeout

During RTSP procedures, timeout rules are applied to ensure the completion of a specific operation (e.g., WFD Capability Negotiation, WFD Session establishment, session management, coupling, etc) in a timely manner. This section (and sub-section) defines timeout rules.

The following are RTSP timeout rules:

1. Upon the successful establishment of a TCP connection between a WFD Source and a WFD Sink, the WFD Source shall send an RTSP M1 Request message to the WFD Sink within 6 second.
2. The timeout for an RTSP message exchange (i.e. between a Request message and a corresponding Response message) by a WFD Device is 5 seconds (e.g., time between an RTSP M1 Request message and an RTSP M1 Response message), except for an RTSP M16 message exchange. The timeout rules for an RTSP M16 message exchange are described in section 6.5.1.
3. Until a WFD Session has been established, the timeout between the receipt of a Response message and the transmission of a successive request is 6 seconds (e.g. time between upon receiving an RTSP M3 Response message and sending an RTSP M4 Request message).
4. Until a WFD Session has been established, the timeout between the transmission of a previous Response message and the transmission of a successive request is 6 seconds (e.g. between sending an RTSP M2 Response message and sending an RTSP M3 Request message).
5. The timeout between the receipt of an RTSP M6 Response message and the transmission of an RTSP M7 Request message takes two values depending on the support of link content protection (see section 4.7):
   - If link content protection is not supported, the timeout value is 6 seconds (same as rule #3).
   - If link content protection is supported, the timeout value is 9 seconds in order to accommodate the HDCP locality check (including retries) and key negotiation that shall be done before transmitting an RTSP PLAY.
6. After a WFD Session establishment, the timeout between the transmission of an RTSP M5 Response message and the transmission of the corresponding RTSP M7, M8, or M9 Request message is 6 seconds.
7. After a WFD Session establishment, the timeout between the transmission of an RTSP M4 Response message and the transmission of the corresponding RTSP M6 Request message for switching the transport as described in section 4.10.6 is 6 seconds.

If any of the RTSP timeout rules is not satisfied, the RTSP procedures (and RTSP session if it has been established) shall be aborted, and a WFD Source and a WFD Sink returns to an RTSP Init state.

6.5.1 WFD keep-alive

The WFD keep-alive function is used to periodically ensure the status of WFD Session. This WFD keep-alive function has following steps, and this timeout rule is applied to both a WFD Source and a WFD Sink (WFD Sinks in the case of Coupled Sink Operation).

1. A WFD Source indicates the timeout value via the “Session:” line in the RTSP M6 Response message.

2. A WFD Source shall send RTSP M16 Request messages and the time interval of two successive RTSP M16 Request messages shall be smaller than the timeout value set by the RTSP M6 Response message minus 5 seconds.
3. A WFD Sink shall respond with an RTSP M16 Response message upon successful receiving the RTSP M16 Request message.
4. If the WFD Source does not receive at least one RTSP M16 Response message within the period of the assigned timeout, the WFD Source shall abort the RTSP session and RTP session. The WFD Source recognizes that the WFD Sink is still alive as far as any RTSP M16 Response from that WFD Sink is received, independent on the RTSP status code.
5. If the WFD Sink does not receive at least one RTSP M16 Request message from the WFD Source within the period of the assigned timeout, the WFD Sink shall abort the RTSP session and RTP session.

The timeout value shall not be smaller than 10 seconds. The default value is 60 seconds when the value is not specified in the RTSP M6 (SETUP) Response message (refer to RFC 2326 [20] “12.37 session”).

6.5.2 Timeout before RTSP Procedure

If Wi-Fi P2P is used as WFD Connectivity, the WFD Device shall successfully establish a TCP connection with the peer WFD Device for the purpose of RTSP procedures within 90 seconds of a successful transmission of the M4 Message of the Four-Way WPA2 handshake.

If TDLS is used as WFD Connectivity, the WFD Device shall successfully establish a TCP connection with the peer WFD Device for the purpose of RTSP procedure within 90 seconds of a successful transmission of the TDLS Setup Confirm message.

A successful TCP connection establishment constitutes the following steps;

- For Wi-Fi P2P case, a WFD Device shall assign itself or obtain an IP address depending on its role in the P2P Group. For TDLS case, IP address assignment shall occur prior to TDLS link establishment.
- The WFD Source (or the Primary Sink in the case of initiating RTSP procedure for Coupling) shall start the RTSP server.
- In the event of an unsuccessful attempt at establishing the TCP connection, the WFD Sink shall retry the TCP connection attempt at regular intervals until the expiration of the 90 seconds timeout period.

In the event of an unsuccessful TCP connection within 90 seconds, then the WFD devices may teardown the Wi-Fi P2P or TDLS link.

6.6 RTSP Syntax remark

This subsection describes additional rules and clarification for RTSP syntax.

6.6.1 CSeq

The CSeq number in each RTSP header shall be assigned by the sender of the RTSP request message, i.e., separate counters for WFD Source requested messages and for WFD Sink requested messages. Same rule is applied during a Coupling procedure, i.e., separate counters for Primary Sink requested messages and for Secondary Sink requested messages). The CSeq number shall be incremented by one as specified in [20]. The initial value of CSeq at the WFD Source and the initial value of CSeq at the WFD Sink may not be identical.

6.6.2 Delimiter for parameters

In GET_PARAMETER Requests, each parameter name shall have “CRLF” at the end of the name.

A new line, i.e., an additional CRLF after the CRLF of the last body line, is not required at the end of the content of the RTSP message. If a new line, i.e., additional CRLF, exists at the end of the content of the RTSP message, the Content-Length shall be calculated to incorporate this. As a result, the recipient parses the received message body correctly. At the end of the header section, two CRLF (one is a delimiter for the last header line) shall be inserted, i.e., one blank line exists.

6.6.3 Message header

In the RTSP message header, the message header field shall be separated from the message header value by one colon (":") and one space (e.g., "CSeq: 1" or "Session: ABCDEF"). Note that the HTTP 1.1 specification allows multiple spaces (and/or Linear White Space: LWS), but this Specification uses one space for easier implementation for parsing.

6.6.4 Content-Encoding

Although the RTSP specification [20] requires inclusion of the Content-Encoding header for SET_PARAMETER Requests, the default encoding in this specification is UTF8 and it is not mandatory to include Content-Encoding header when using default encoding.

6.6.5 Case Sensitivity

In all RTSP messages, all alphabet characters are case insensitive, except for the RTSP methods, such as GET_PARAMETER, PLAY and so on, as specified in [20] (RFC 2326) and its referring document of HTTP 1.1.

6.6.6 Content-Length and Content-Type

The Content-Type and the Content-Length header lines can appear in any order.

If an RTSP message carries a message body, both the Content-Type and the Content-Length header lines shall be included in the header of the RTSP message.

The ABNF for the Content-Type header line is:

```
Content-Type            = “Content-Type:” SP “text/parameters”
```

7 Remote I2C Read/Write Messaging Transaction

A WFD Source and a WFD Sink exchange Remote I2C Read/Write messaging transactions using the TCP port for I2C Read/Write.

The TCP port number to be used at the WFD Sink for I2C Read/Write message transaction is included in the wfd-I2C parameter (specified in section 6.1.12) in the RTSP M3 Response message. The WFD Sink shall be ready to accept incoming connections from the WFD Source at the TCP port for I2C Read/Write before replying to the RTSP M3 Request message containing the wfd-I2C parameter.

Once established, the WFD devices shall use a single TCP connection for the duration of the WFD Session for all Remote I2C Requests and replies between them.

There are two types of messaging transactions:

- Remote I2C Request sent by a WFD Source
- Remote I2C Reply sent by a WFD Sink.

A Request-and-Reply Message Transaction pair defines the remote I2C message transaction sequence.

Two types of requests from a WFD Source to a WFD Sink for remote I2C Read/Write message transaction are defined:

- Remote_I2C_Read_Request
- Remote_I2C_Write_Request

Four types of responses from a WFD Sink to a WFD Source for remote I2C read/Write message transaction are defined:

- Remote_I2C_Read_Reply_Ack
- Remote_I2C_Read_Reply_Nak
- Remote_I2C_Write_Reply_Ack
- Remote_I2C_Write_Reply_Nak

Remote I2C Read/Write requests consist of high-level data structures that are described in section 7.1 and 7.2. A Remote I2C Write Request includes one or more Write Transactions. A Remote I2C Read Request includes one Read Transaction and one or more Write Transactions.

One remote I2C Write Transaction in a Remote I2C Read/Write Request may be mapped into one or more actual I2C bus write transactions. One remote I2C Read Transaction in a Remote I2C Read Request may be mapped into one or more actual I2C bus read transactions. Whether a WFD Sink translates the Remote I2C Read/Write Requests into actual I2C bus transactions or not is implementation specific and is out of scope of WFD specification.

The I2C bus writes are required to set up the address for the I2C bus reads.

Figure 28 shows the case of a discrete WFD Sink dongle mapping the Remote I2C data structure into one or more I2C bus transactions.

If a WFD Sink indicates support of Remote I2C Read/Write function at a wfd-I2C parameter in an RTSP M3 Response message, the WFD Sink shall support the reception of a Remote_I2C_Read_Request and a Remote_I2C_Write_Request message and the transmission of a Remote_I2C_Read_Reply_Ack, a Remote_I2C_Read_Reply_Nak, a Remote_I2C_Write_Ack and a Remote_I2C_Write_Nak messages. If a WFD Source indicates support of Remote I2C Read/Write function by including a wfd-I2C parameter in an RTSP M3 Request message, the WFD Source shall support the transmission of a Remote_I2C_Read_Request and a Remote_I2C_Write_Request message and the reception of a Remote_I2C_Read_Reply_Ack, a Remote_I2C_Read_Reply_Nak, a Remote_I2C_Write_Ack and a Remote_I2C_Write_Nak messages.

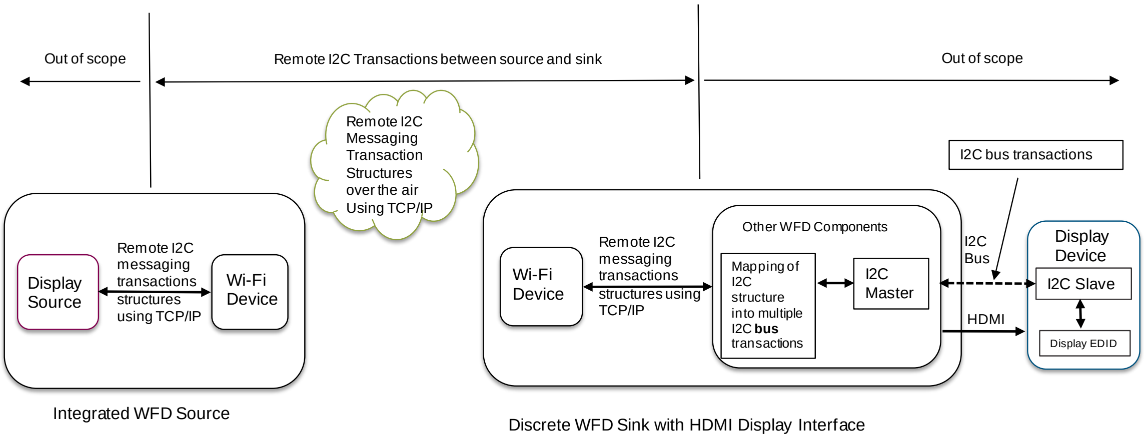


Figure 28. Remote I2C transaction example

7.1 Remote_I2C_Read_Request

A WFD Source sends a Remote_I2C_Read_Request to a WFD Sink using the TCP port number assigned, to initiate I2C read(s) from remote display device(s).

The data structure of Remote_I2C_Read_Request is defined in Table 99

Table 99. Remote_I2C_Read_Request

Field	Size (Octets)	Description
Request identifier	1	This field should be set to 0x00 to indicate Remote_I2C_Read_Request.
Number_Of_I2C_Write_Transactions	1	The total number of I2C write transactions to be sent in this message transaction.
For (i=0; i <Number_Of_I2C_Write_Transactions - 1; i++) {		
Write_I2C_Device_Identifier[i]	1	The I2C device identifier to receive the write request
Number_of_Bytes_To_Write[i]	1	The number of data bytes to write to the I2C device
for (j=0; j < Number_of_Bytes_To_Write; j++) I2C_Data_To_Write[i][j]	Number_of_Bytes_To_Write[i]	The I2C write data for the write device. Total number of bytes to write for each write transaction is equal to Number_of_Bytes_To_Write[i]
No_Stop_Bit[i]	1	When set to a 0x01, a stop bit is not sent at the end of the I2C transaction; otherwise when set to a 0x00, a stop will be generated at the end of the I2C transaction.
I2C_Transaction_Delay[i] }	1	The amount of delay to insert between this and the next I2C transaction. The delay unit is 10ms. The delay range is 0 to 150 ms.
Read_I2C_Device_Identifier	1	The I2C device identifier to receive the read request.
Number_Of_Bytes_To_Read	1	The number of data bytes requested to read from the I2C device.

7.2 Remote_I2C_Write_Request

The WFD Source sends a Remote_I2C_Write_Request to the WFD Sink using the TCP port number assigned, to initiate I2C write(s) to remote display device(s).

The data structure of Remote_I2C_Write_Request is defined in Table 100.

Table 100. Remote_I2C_Write_Request

Field	Size (Octets)	Description
Request Identifier	1	Field set to 0x01 to indicate Remote_I2C_Write_Request.
Write_I2C_Device_Identifier	1	The I2C device identifier to receive the write request.
Number_of_Bytes_To_Write	1	The number of data bytes to write to the I2C device.
For (j=0; j < Number_of_Bytes_To_Write; j++) I2C_Data_To_Write[j]	Number_of_Bytes_To_Write	The I2C write data for the write device Total number of bytes to write is equal to Number_of_Bytes_To_Write.

7.3 Remote_I2C_Read_Reply_Ack

The WFD Sink sends a Remote_I2C_Read_Reply_Ack to the WFD Source in response to a Remote_I2C_Read_Request using the TCP port number assigned, to indicate that successful I2C reads are performed and read data are presented.

The data structure of Remote_I2C_Read_Reply_Ack is defined in Table 101.

Table 101. Remote_I2C_Read_Reply_Ack

Field	Size (Octets)	Description
Reply Identifier	1	Field set to 0x00 to indicate Remote_I2C_Read_Reply_Ack.
Number_of_Bytes_To_Read	1	The number of data bytes read from the I2C device.
For (j=0; j < Number_of_Bytes_Read; j++) I2C_Device_Byte_Read[j]	Number_of_Bytes_To_Read	A data byte read from the I2C device

7.4 Remote_I2C_Read_Reply_Nak

The WFD Sink sends a Remote_I2C_Read_Reply_Nak to the WFD Source in response to a Remote_I2C_Read_Request using the TCP port number assigned, to indicate I2C read failure.

The data structure of Remote_I2C_Read_Reply_Nak is defined in Table 102.

Table 102. Remote_I2C_Read_Reply_Nak

Field	Size (Octets)	Description
Reply Identifier	1	Field set to 0x80 to indicate Remote_I2C_Read_Reply_Nak.

7.5 Remote_I2C_Write_Reply_Ack

The WFD Sink sends a Remote_I2C_Write_Reply_Ack to the WFD Source in response to a Remote_I2C_Write_Request using the TCP port number assigned, to indicate successful I2C Writes are performed.

The Data structure of Remote_I2C_Write_Reply_Ack is defined in Table 103

Table 103. Remote_I2C_Write_Reply_Ack

Field	Size (Octets)	Description
Reply Identifier	1	Field set to 0x01 to indicate Remote_I2C_Write_Reply_Ack.

7.6 Remote_I2C_Write_Reply_Nak

The WFD Sink sends a Remote_I2C_Write_Reply_Nak to the WFD Source in response to a Remote_I2C_Write_Request using the TCP port number assigned, to indicate I2C Write failure.

The data structure of Remote_I2C_Write_Reply_Nak is defined in Table 104.

Table 104. Remote_I2C_Write_Reply_Nak

Field	Size (Octets)	Description
Reply Identifier	1	Field set to 0x81 to indicate Remote_I2C_Write_Reply_Nak.

8 Preferred Display Mode

Because the Preferred Display Mode as defined in the Wi-Fi Display Technical Specification v1.0 has been deprecated, this section is removed. This feature is not applicable for a WFD R2 Device.

Appendix A MPEG System Layer (informative)

This appendix provides an overview of MPEG System Layer specified in ITU-T Rec. H.222.0 [2]. The following basic functions are provided therein:

- synchronization of multiple compressed streams on decoding
- interleaving of multiple compressed streams into a single stream
- initializing of buffering for decoder start up
- continuous buffer management
- time identification
- multiplexing and identification of various components in a system stream

MPEG2-TS multiplexing is used for transport over the RTP/UDP/IP layers for Wi-Fi Display.

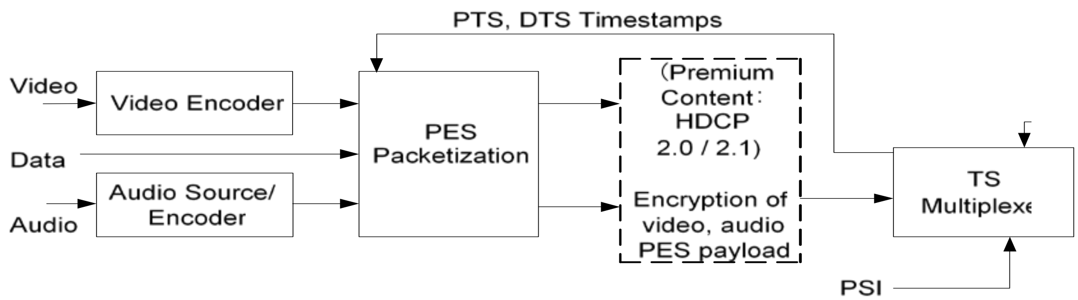


Figure 29. Overview of the H.222.0 Systems Layer – Transmission, Multiplexing and Synchronization

Appendix B MPEG-TS Parameters for Audio, Video and Auxiliary Elementary Streams (normative)

This appendix specifies MPEG2-TS parameters and encapsulation of MPEG2-TS into RTP packets.

The parameters necessary to be specified to include the different supported audio, video and auxiliary stream formats in the MPEG-TS bit stream are described in Table 105, and its amendment [36] or in this specification.

Table 105. MPEG-TS parameters for supported audio, video and image formats

AV Format	Stream_id (PES header)	Stream_type (MPEG-TS PMT section)	Descriptor tag (if present) (MPEG-TS PMT section)	Usage Classifier, Reference
H.264 video	0xE0-0xEF	0x1B	0x28 [AVC descriptor]	Tables 2-22, 2-34, 2-45 of [2]
H.265 video	0xE0-0xEF	0x24, 0x25	0x38 [HEVC video descriptor]	Table 2-22, 2-34, 2-45 as specified in [39]
JPEG	0xBD	0x92	0x92 [JPEG image descriptor]	Table 2-22 of [39] , and as specified in this specification.
PNG	0xBD	0x93	0x93 [PNG image descriptor]	Table 2-22 of [39] , and as specified in this specification.
LPCM	0xBD	0x83	0x83 [LPCM audio stream descriptor]	Table 2-22 of [2], Table 4.6.3-1 of [5], Table 4.6.6.1-1 of [5]
Dolby AC-3	0xBD	0x81	0x81 [AC-3 audio descriptor]	Table 2-22 of [2], Table 4.6.3-1 of [5]
AAC	0xC0-0xDF	0x0F	0x2B [MPEG2 AAC audio descriptor]	Tables 2-22, 2-34, 2-45 of [2]
E-AC-3	0xBD	0x87	0xCC [E-AC-3 audio descriptor]	Table 2-22 of [2], Annex G of [50]
AC-4	0xBD	0x06 (private stream, use descriptor tag in PMT)	0x15 [AC-4 audio extended descriptor, in [65]]	Table D.11 of [54]
DTS-HD	0xBD	0x06 (private stream, use descriptor tag in PMT)	0x7B [DTS-HD Audio Descriptor]	Appendix B.4.1
MPEG-4 AAC and AAC-ELDv2	0xC0-0xDF	0x11	0x1C [MPEG-4 audio descriptor]	Tables 2-22, 2-34, 2-45 of [2]
MPEG-H 3D Audio	0xC0-0xDF	0x2D	0x1C [MPEG-4 audio descriptor]	Tables 2-22, 2-34, 2-45 of [2]

When using the audio formats listed in section 3.4.1, audio format, PES payload usage and H.222 parameter settings (e.g., stream_type signaled in Stream Program map section in Program Map Table within TS packets, and PES header setting such as stream_id, PTS, PES_extension_flag and so on) shall be compliant with the referred document indicated in the table, except with the following additional/modified rules.

- When referring to [5], ISO/IEC 13818-1 (The system part of the MPEG-2 standard) is replaced with [2] (ITU-R Rec. H.222)
- When referring to [5], sections 4.6.1, 4.6.5-4.6.5.4.2, 4.6.6.2-4.6.6.4 and 4.6.6.6 are not referred.
- When referring to [5] for LPCM, following modifications are used.

  Known errata in it are fixed as follows;

  - “1536 bytes” in Table 4.5.2.1-1 is replaced by “1920 bytes”.

- "0000 0100b" for number_of_frame_headers in Table 4.5.2.1-2 and in its explanation is replaced by "0000 0110b".

for 48ksps 16bits 2ch LPCM, the following parameters are used;

- Number of audio sample data per one Access Unit : 80 samples as specified in [5].
- One audio Access Unit length : 1/600 second as calculated from above.
- Substream_id : 1010 0000b.

for 44.1ksps 16bits 2ch LPCM, the following parameters are used;

- Number of audio sample data per one Access Unit : 80 samples as specified in [5].
- One audio Access Unit length : 4/2205 second as calculated from above.
- Substream_id : 1010 0000b.

PES structure and field values are specified in Table 106 to allow the HDCP system 2.x operation. When the HDCP 2.x encryption is not used, this table is identical to Table 4.5.2.1-2 in [5].

- LPCM audio samples are in network byte order (big-endian) using two's complement integers, as specified in [5]. The order of output of bytes is in network byte order. As a result, the order of the bits in one mono-audio sample is from MSB to LSB, i.e., b15b14b13....b2b1b0.

(note-1) These fields will appear only when PES_extension_flag=1 when using the HDCP 2.x encryption.

Table 106. PES structure and field values for 48ksps/44.1ksps 16bits 2-ch LPCM

Field		Number of bits	Number of bytes	Value
PES Header	packet_start_code_prefix	24	3	0x00 00 01
	stream_id	8	1	0b10111101 [private stream1]
	PES_packet_length	16	2	0x07 8E : if HDCP 2.x is not used 0x07 9F : if HDCP 2.x is used
	'10'	2	1	0b10
	PES_scrambling_control	2		0b00 [not scrambled]
	PES_priority	1		0b0 [no priority]
	data_alignment_indicator	1		0b0 [not defined]
	Copyright	1		0b0 [not defined]
	original_or_copy	1		0b1 [original] or 0b0 [copy]
	PTS_DTS_flag	2	1	10b [present]
	ESCR_flag	1		0b0 [not present]
	ES_rate_flag	1		0b0 [not present]
	DSM_trick_mode_flag	1		0b0 [not present]
	additional_copy_info_flag	1		0b0 [not present]
	PES_CRC_flag	1		0b0 [not present]
	PES_extension_flag	1		0b0 : if HDCP 2.x is not used 0b1 : if HDCP 2.x is used
	PES_header_data_length	8	1	0x07 : if HDCP 2.x is not used 0x18 : if HDCP 2.x is used
	'0010'	4	5	0b0010
	PTS[32…30]	3		shall be set to correct value
	marker_bit	1		0b1

Field		Number of bits	Number of bytes	Value
	PTS[29…15]	15		shall be set to correct value
	marker_bit	1		0b1
	PTS[14…0]	15		shall be set to correct value
	marker_bit	1		0b1
	PES_private_data_flag	1	1	0b1 [present] (note-1)
	pack_header_field_flag	1		0b0 [not present] (note-1)
	program_packet_sequence_counter_flag	1		0b0 [not present] (note-1)
	P-STD_buffer_flag	1		0b0 [not present] (note-1)
	Reserved	3		0b111 (note-1)
	PES_extension_flag_2	1		0b0 [not present] (note-1)
	PES_private_data	128	16	HDCP 2.x counter values and marker_bit. See [18]. (note-1)
	stuffing bytes	16	2	0xFF FF
PES Payload Private Header	sub_stream_id	8	1	0b10100000 [0th sub stream]
	number_of_frame_header	8	1	0b00000110 [6 AUs]
	Reserved	7	1	0b0000000
	audio_emphasis_flag	1		See Table 4.5.2.1-3 of [5]
	quantization_word_length	2	1	See Table 4.5.2.1-4 of [5]
	audio_sampling_frequency	3		See Table 4.5.2.1-5 of [5]
	number_of_audio_channel	3		See Table 4.5.2.1-6 of [5]
PES Payload data	Audio Data (LPCM)	15360	1920	

The PES structure and field values for LPCM audio format applicable for WFD R2 devices is specified in section B.4 with additional modifications to [5] to carry the single channel LPCM and CTA audio for multi-channel LPCM based on [47].

When referring to [5] for Dolby AC-3, the following modifications are used.

- Reference to “DVD-Video specification ([R1])” in [5] is not to be referred.
- “Max. 2-ch/Dolby AC-3” for “Number of channels” at Table 4.5.1-1 in [5] is replaced by “Max. 6-ch/Dolby AC-3”.
- “between 64k and 448 kbps” for “Bit rate” at Table 4.5.3-1 in [5] is replaced by “between 64k and 640 kbps”.
- As Table 4.6.3-1 in [5] specifies that stream_type=0x81, AC-3 here uses System-A defined in reference [R11] within [5].
- PES structure and field values are specified in Table 107. This is consistent with the description in [5].

(note-1) These fields will appear only when PES_extension_flag=1 when using the HDCP 2.x encryption.

Table 107. PES structure and field values for Dolby AC-3

Field		Number of bits	Number of bytes	Value
PES Header	packet_start_code_prefix	24	3	0x00 00 01
	stream_id	8	1	0b10111101 [private stream1]
	PES_packet_length	16	2	Variable

Field		Number of bits	Number of bytes	Value
	'10'	2	1	0b10
	PES_scrambling_control	2		0b00 [not scrambled]
	PES_priority	1		0b0 [no priority]
	data_alignment_indicator	1		0b0 [not defined]
	Copyright	1		0b0 [not defined]
	original_or_copy	1		0b1 (original) or 0b0 (copy)
	PTS_DTS_flag	2	1	10b [present]
	ESCR_flag	1		0b0 [not present]
	ES_rate_flag	1		0b0 [not present]
	DSM_trick_mode_flag	1		0b0 [not present]
	additional_copy_info_flag	1		0b0 [not present]
	PES_CRC_flag	1		0b0 [not present]
	PES_extension_flag	1		0b0 : if HDCP 2.x is not used 0b1 : if HDCP 2.x is used
	PES_header_data_length	8	1	0x07 : if HDCP 2.x is not used 0x18 : if HDCP 2.x is used
	'0010'	4	5	0b0010
	PTS[32…30]	3		shall be set to correct value
	marker_bit	1		0b1
	PTS[29…15]	15		shall be set to correct value
	marker_bit	1		0b1
	PTS[14…0]	15		shall be set to correct value
	marker_bit	1		0b1
	PES_private_data_flag	1	1	0b1 [present] (note-1)
	pack_header_field_flag	1		0b0 [not present] (note-1)
	program_packet_sequence_counter_flag	1		0b0 [not present] (note-1)
	P-STD_buffer_flag	1		0b0 [not present] (note-1)
	Reserved	3		0b111 (note-1)
	PES_extension_flag_2	1		0b0 [not present] (note-1)
	PES_private_data	128	16	HDCP 2.x counter values and marker_bit. See [18]. (note-1)
	stuffing bytes	16	2	0xFF FF
PES Payload	Audio Data (Dolby AC-3)	variable	variable	1 byte or more, and 2560 bytes or less

For E-AC-3, Annex G of [36] is used as the reference for carriage of E-AC_3 audio within MPEG-2 TS.

- PES structure and field values are specified in Table 108. This is consistent with the description in Annex G of [36].

(note-1) These fields will appear only when PES_extension_flag=1 when using the HDCP 2.0/2.1 encryption.

Table 108. PES structure and field values for E-AC-3

Field		Number of bits	Number of bytes	Value
PES Header	packet_start_code_prefix	24	3	0x00 00 01
	stream_id	8	1	0b10111101 [private stream1]
	PES_packet_length	16	2	variable
	'10'	2	1	0b10
	PES_scrambling_control	2		0b00 [not scrambled]
	PES_priority	1		0b0 [no priority]
	data_alignment_indicator	1		0b0 [not defined]
	copyright	1		0b0 [not defined]
	original_or_copy	1		0b1 (original) or 0b0 (copy)
	PTS_DTS_flag	2	1	10b [PTS fields present]
	ESCR_flag	1		0b0 [not present]
	ES_rate_flag	1		0b0 [not present]
	DSM_trick_mode_flag	1		0b0 [not present]
	additional_copy_info_flag	1		0b0 [not present]
	PES_CRC_flag	1		0b0 [not present]
	PES_extension_flag	1		0b0 : if HDCP 2.0/2.1 is not used 0b1 : if HDCP 2.0/2.1 is used
	PES_header_data_length	8	1	0x07 : if HDCP 2.0/2.1 is not used 0x18 : if HDCP 2.0/2.1 is used
	'0010'	4	5	0b0010
	PTS[32…30]	3		shall be set to correct value
	marker_bit	1		0b1
	PTS[29…15]	15		shall be set to correct value
	marker_bit	1		0b1
	PTS[14…0]	15		shall be set to correct value
	marker_bit	1		0b1
	PES_private_data_flag	1	1	0b1 [present] (note-1)
	pack_header_field_flag	1		0b0 [not present] (note-1)
	program_packet_sequence_counter_flag	1		0b0 [not present] (note-1)
	P-STD_buffer_flag	1		0b0 [not present] (note-1)
	Reserved	3		0b111 (note-1)
	PES_extension_flag_2	1		0b0 [not present] (note-1)
	PES_private_data	128	16	HDCP 2.0/2.1 counter values and marker_bit. See [21]. (note-1)
	stuffing bytes	16	2	0xFF FF

Field		Number of bits	Number of bytes	Value
PES Payload	Audio Data (E-AC-3)	variable	variable	1 byte or more, and 12096 bytes or less

The coding of AC-4 frames of an AC-4 elementary stream is based upon TS 103 190-1. AC-4 elementary streams consist of presentations, which define a set of one or more substreams to be presented simultaneously.

AC-4 PES packaging is achieved by packing multiple AC-4 frames into PES packets, to ensure minimal padding is needed. AC-4 frames shall not be split over two or more PES packets.

Additional details on AC-4 PES packaging may be found in section 6.6 of [53].

For the carriage of Dolby TrueHD in MPEG 2 Transport streams for WFD, the transport priority field shall be ignored and all appropriate PES payloads shall only contain Dolby TrueHD (MLP) access units.

At 48 kHz, a Dolby MAT frame contains a duration of 960 samples of audio. There shall be 1 Dolby MAT frame per each PES packet.

- The MPEG-4 AAC MPEG-2 TS parameters apply to the following codecs from the MPEG-4 AAC family:
  - AAC-LC, HE-AAC or HE-AACv2. These codecs are signaled and discovered by means of the “CTA” audio format
  - AAC-ELDv2. This codec is signaled and discovered by means of the “AAC-ELDv2” audio-format.
- When referring to [5] for AAC-LC, following modifications are used.
  - Reference to “DVD-Video specification ([R1])” in [5] is to not to be referred.
  - “04_{16} | MPEG2 audio (ISO/IEC 13818-3)” at Table 4.6.3-1 in [5] is replaced by “$0F_{16}$ | MPEG2 audio (ISO/IEC 13818-7)”.
  - Description in section 4.5.5 is read as “For MPEG2 audio the audio stream shall comply with ISO/IEC 13818-7”.
  - In this specification, only base stream is used and extension stream is not used, and description related to extension stream in [5] is to be neglected.
  - PES structure and field values are specified in Table 109. This is consistent with the description in [5].

(note-1) These fields will appear only when PES_extension_flag=1 when using the HDCP 2.x encryption.

Table 109. PES structure and field values for AAC-LC

Field		Number of bits	Number of bytes	Value
PES Header	packet_start_code_prefix	24	3	0x00 00 01
	stream_id	8	1	0b11000000 to 0b11011111 [MPEG2 AAC]
	PES_packet_length	16	2	Variable
	‘10’	2	1	0b10
	PES_scrambling_control	2		0b00 [not scrambled]
	PES_priority	1		0b0
	data_alignment_indicator	1		0b0
	Copyright	1		0b0
	original_or_copy	1		0b1 (original) or 0b0 (copy)
	PTS_DTS_flag	2	1	10b
	ESCR_flag	1		0b0
	ES_rate_flag	1		0b0
	DSM_trick_mode_flag	1		0b0

Field		Number of bits	Number of bytes	Value
	additional_copy_info_flag	1		0b0
	PES_CRC_flag	1		0b0
	PES_extension_flag	1		0b0 : HDCP 2.x is not used 0b1 : HDCP 2.x is used
	PES_header_data_length	8	1	0x07 : HDCP 2.x is not used 0x18 : HDCP 2.x is used
	'0010'	4	5	0b0010
	PTS[32…30]	3		shall be set to correct value
	marker_bit	1		0b1
	PTS[29…15]	15		shall be set to correct value
	marker_bit	1		0b1
	PTS[14…0]	15		shall be set to correct value
	marker_bit	1		0b1
	PES_private_data_flag	1	1	0b1 (note-1)
	pack_header_field_flag	1		0b0 (note-1)
	program_packet_sequence _counter_flag	1		0b0 (note-1)
	P-STD_buffer_flag	1		0b0 (note-1)
	Reserved	3		0b111 (note-1)
	PES_extension_flag_2	1		0b0 (note-1)
	PES_private_data	128	16	HDCP 2.x counter values and marker_bit. See [18]. (note-1)
	stuffing bytes	16	2	0xFF FF
PES Payload	Audio Data (AAC-LC)	variable	variable	1 byte or more, and 2020 bytes or less

When referring to [2] for AAC-LC, following modification are used.

- MPEG-2_AAC_channel_configuration can have value 7 also, i.e., not limited from 1 to 6, to indicate 8ch (7.1ch) as specified in [17].
- When using AAC-LC for 2ch to 7.1ch, stream shall use 1 ADTS.

PES structure and field values used in the PES header for an auxiliary stream when it carries still image data (JPEG or PNG) for overlays are specified in section B.2.

When the JPEG encoded stream is carried over MPEG2-TS within the scope of this specification, the stream type 0x92 and descriptor tag value of 0x92 shall be used. The carriage of JPEG picture over MPEG2-TS transport stream is specified in section B.2.22.

When the PNG encoded stream is carried over MPEG2-TS within the scope of this specification, the stream type 0x93 and descriptor tag value of 0x93 shall be used. The carriage of PNG picture over MPEG2-TS transport stream is specified in section B.2.3.

A WFD Source shall generate only audio, video and auxiliary stream formats (and associated MPEG2-TS metadata parameters above) that the WFD Sink has already been discovered to support during the discovery phase.

In turn, a WFD Sink shall decode audio, video and auxiliary stream formats as indicated by MPEG2-TS metadata parameters above. If the WFD Source changes audio, video or auxiliary stream formats in the middle of a streaming and this format changes are indicated by transmitted MPEG2-TS metadata parameters, the WFD Sink should continue processing audio, video and auxiliary elementary streams as indicated formats".

B.1 Encapsulation of MPEG2-TS into RTP Packets

A WFD Source shall encapsulate the MPEG2-TS frames in RTP packets following the guidelines on the encapsulation of MPEG systems multiplex streams over RTP [3], [4], and [6]. In particular, MPEG2-TS streams shall be encapsulated in RTP packets as in section 2 of RFC 2250 [4]. The RTP packet length shall be chosen such that the length of the RTP packet plus the length of the UDP and IP headers is less than or equal to the MTU size. As shown in Figure 30 and Figure 31, the RTP payload carries an integral number of MPEG2-TS packets, computed by dividing RTP payload length by the length of an MPEG2-TS packet and rounding the result down. A maximum of 7 MPEG2-TS frames shall be encapsulated in a single RTP packet.

Table 110 lists the RTP encapsulation of the MPEG-TS. Each RTP packet shall contain a timestamp derived from the sender's 90 kHz clock reference synchronized to the system stream PCR (MPEG2-TS), and represents the target transmission time of the first byte of the packet payload. This RTP timestamp is not passed to the decoder, and is solely used to estimate and reduce network induced jitter and synchronize relative time drift between RTP transmitter and receiver.

When the HDCP 2.x encryption is applied, refer to section 3.6.2 of ref [18] for HDCP 2.0, ref [27] for HDCP 2.1 and ref [35] for HDCP 2.2 to carry an encrypted PES packet in TS packets. For higher version than HDCP 2.2, refer to corresponding specification released by DCP LLC.

Table 110. RTP encapsulation of MPEG-TS

Item	Parameter	Usage Classifier, Reference
RTP header PT field	'33'	from [4], [6] for MPEG2-TS
Marker bit (M)	'1' whenever timestamp is discontinuous	from [4]
RTP Timestamp	32-bit timestamp derived from a 90 kHz clock, representing the target transmission time for the first byte of the packet. This clock is synchronized to the system stream PCR (TS) or the SCR (PS), and represents the target transmission time of the first byte of the packet payload.	From [4]

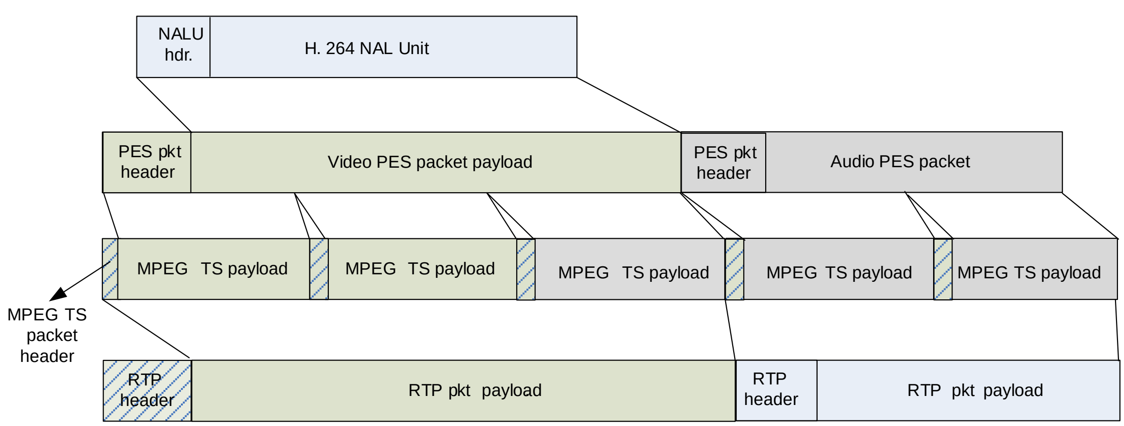


Figure 30. Example of recommended encapsulation of MPEG2-TS packets

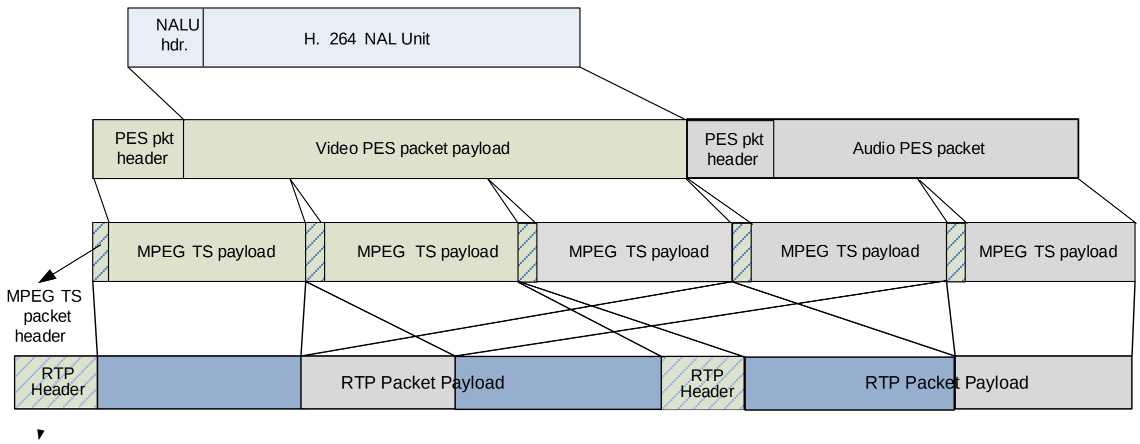


Figure 31. Example of recommended encapsulation of MPEG2_TS packets

B.2 PES Structure and Blending Operation for Auxiliary Streams

PES structure and field values for an auxiliary stream when it carries still image data (JPEG or PNG) for overlays are specified in Table 111.

Table 111. PES structure and field values for Auxiliary Stream

Field		Number of bits	Number of bytes	Value
PES Header	packet_start_code_prefix	24	3	0x00 00 01
	stream_id	8	1	0b10111101 [private stream1]
	PES_packet_length	16	2	Variable
	‘10’	2	1	0b10
	PES_scrambling_control	2		0b00 [not scrambled]
	PES_priority	1		0b0 [no priority]
	data_alignment_indicator	1		0b1
	Copyright	1		0b0 [not defined]
	original_or_copy	1		0b1 (original) or 0b0 (copy)
	PTS_DTS_flag	2	1	10b [present]
	ESCR_flag	1		0b0 [not present]
	ES_rate_flag	1		0b0 [not present]
	DSM_trick_mode_flag	1		0b0 [not present]
	additional_copy_info_flag	1		0b0 [not present]
	PES_CRC_flag	1		0b0 [not present]
	PES_extension_flag	1		0b0 : if HDCP 2.0/2.1 is not used 0b1 : if HDCP 2.0/2.1 is used
	PES_header_data_length	8	1	0x07 : if HDCP 2.0/2.1 is not used 0x18 : if HDCP 2.0/2.1 is used
	‘0010’	4	5	0b0010
	PTS[32…30]	3		shall be set to correct value

Field		Number of bits	Number of bytes	Value
	marker_bit	1		0b1
	PTS[29…15]	15		shall be set to correct value
	marker_bit	1		0b1
	PTS[14…0]	15		shall be set to correct value
	marker_bit	1		0b1
	PES_private_data_flag[1]	1	1	0b1 [present]
	pack_header_field_flag[1]	1		0b0 [not present]
	program_packet_sequence _counter_flag[1]	1		0b0 [not present]
	P-STD_buffer_flag[1]	1		0b0 [not present]
	Reserved[1]	3		0b111
	PES_extension_flag_2[1]	1		0b0 [not present]
	PES_private_data[1]	128	16	HDCP 2.0/2.1 counter values and marker_bit. See [21]
	stuffing bytes	16	2	0xFF FF
PES Payload Private Header	Overlay_ID	8	1	Unique identifier to represent each overlay layer
	presentation_mode[2]	2	1	0b00: deferred presentation mode (suspend presentation if the overlay is currently active) 0b01: active presentation mode (content to be displayed using the PTS) 0b10: deactivate presentation (cancel the presentation resources for this overlay) 0b11: reserved
	data_present_flag[2]	1		0b0: the Graphics data field is empty (Graphics data for the overlay ID was already sent in earlier PES packets) 0b1: Graphics data is included
	Reserved[2]	5		
	overlay_pos_top_left_x[2]	16	2	X coordinate of the Graphic object's top-left corner
	overlay_pos_top_left_y[2]	16	2	Y coordinate of the Graphic object's top-left corner
	overlay_width[2]	16	2	Width of the Graphics image in number of pixels
	overlay_height[2]	16	2	Height of the Graphics image in number of pixels
	overlay_z_layer[2]	8	1	Z-order of the Graphics image for blending. The layer with higher Z-order will be composited over the lower Z-order
	blend_alpha_selection[2]	4	2	The type of alpha to be used for blending the overlay Graphics 0b0000: constant alpha 0b0001: per-pixel source alpha (foreground alpha) 0b0010: per-pixel destination alpha (background alpha) 0b0011: per-pixel source alpha with constant alpha modulation 0b0100: per-pixel destination alpha with constant alpha modulation 0b0101: pre-multiplied per-pixel source alpha 0b0110 – 0b1111: reserved
	blend_mode_selection[2]	4		The blending mode to be used to composite the overlay Graphics

Field		Number of bits	Number of bytes	Value
				0b0000: no blending (Alpha = 0) 0b0001: normal alpha blending (Eq. I) 0b0010: reverse alpha blending (Eq III) 0b0011: normal pre-multiplied alpha blending (Eq. V)) 0b0100: pre-multiplied alpha blending with constant alpha modulation (Eq VI) 0b0101 – 0b1111: reserved
	constant_alpha_flag [2]	1		0b0: constant alpha data is not included in the PES 0b1: constant alpha is included
	Reserved	5		
	per_pixel_alpha_info[2]	2		0b00: per pixel alpha is not present 0b01: straight per pixel alpha (not pre-multiplied) is embedded in the Graphics 0b10: pre-multiplied per pixel alpha is embedded in the Graphics 0b11: reserved
	constant_alpha_value [2]	8	1	Alpha value to be used when blend_alpha_selection includes constant alpha. When the constant_alpha_flag is set to 0b0, this field should be set to 0x00 and the WFD R2 Sink shall ignore this field (note-2)
PES Payload data	Graphics Data	variable	variable	Variable
Notes: 1. These fields will appear only when PES_extension_flag=1 when using the HDCP 2.0/2.1 encryption. 2. These fields will appear only when the PES is carrying auxiliary stream data containing Graphics or sub-titles.				

In order to carry the metadata required for blending and composition of an auxiliary stream containing Graphics, the following additional fields are included in the Private Header of the PES payload as shown in Table 109. The Private header field is 14 octets long and is always present even when the Graphics data in the PES payload may be empty.

overlay_ID to uniquely represent each overlay content so that WFD R2 Sink can allocate a layer for its content and perform blending of the display as necessary

presentation_mode to indicate if this overlay content is for deferred presentation mode (i.e. content to be stored until this flag is set for activating the blending of the content), active presentation mode for displaying at the respective PTS, or to deactivate the overlay content in which case the WFD R2 Sink may cancel the allocated resources

data_present_flag indicates if this PES includes the Graphics data. When sending the PES packet first time for a given overlay ID, this flag shall be set and the PES payload shall include the Graphics data. The WFD R2 Source may include Private Header without containing the Graphics data in the subsequent PES payload associated to an overlay ID when sending of the same Graphics data is not required. The WFD R2 Source may set this field to 0 to send the PES without the Graphics data only when the presentation_mode is set to 0b00 or 0b10.

Note: It is recommended that a WFD R2 Source should use TCP transport when it requires to send auxiliary streams to ensure reliability on sending the overlay Graphics content. When in UDP mode, the WFD R2 Source may repeat the PES for auxiliary stream at each frame interval or any other interval to ensure the WFD R2 Sink correctly receives the PES Private Header fields and the Graphics data (when present) to decode and render the overlay components correctly.

overlay_pos_top_left_x indicates the x coordinate of the starting point of this overlay image in number of pixels relative to the reference resolution used for the overlay layers (specified for the full screen)

overlay_pos_top_left_y indicates the y coordinate of the starting point of this overlay image in number of pixels relative to the reference resolution used for the overlay layers (specified for the full screen) overlay_width indicates the width of the overlay image in number of pixels

overlay_height indicates the height of the overlay image in number of pixels

overlay_z_layer indicates the order in which the overlay image is to be blended to the background or base layer

blend_alpha_selection indicates whether the per pixel alpha or constant plane alpha or both have to be used for blending. The per pixel alpha may be selected from alpha of the source (foreground) or the destination (background) layers for blending this overlay graphics

blend_mode_selection indicates whether the normal blend equation (see Equation I), reverse blend equation (see Equation III), normal pre-multiplied alpha blending (see Equation V), or pre-multiplied alpha blending with constant alpha modulation (see Equation VI) to be used to present the given overlay content

constant_alpha_flag indicates whether the PES packet includes constant alpha for the overlay graphics or not. If this flag is set to 0b1, the field with constant_alpha_value shall be present

perpixel_alpha_info indicates whether the per-pixel alpha is available for the Graphics and how it is sent (embedded in the Graphics data in straight or pre-multiplied form).

constant_alpha_value includes the value of the constant alpha to be used for the blending when the blend_alpha_selection indicates the use of constant alpha during the blending of the overlay graphics. If the constant_alpha_flag is set to 0b0, this field, if present, should be set to 0x00 and the WFD R2 Sink shall ignore this field.

The resolution for generating the pixels for all overlay graphics at the specified overlay_pos_top_left_x, overlay_pos_top_left_y, overlay_width and overlay_height shall correspond to the same reference resolution for the overlay layer. The WFD R2 Source shall select the full-screen reference overlay layer resolution using the following criteria:

> Reference overlay resolution = min {maximum video resolution supported by the Source, maximum video resolution supported by the Sink, Sink's native resolution}

where the maximum resolution supported by the Sink corresponds to the maximum among the supported video resolutions indicated by the WFD R2 Sink during the capability negotiation (in M3 Response) and the Sink's native resolution corresponds to the Native Resolutions/Refresh Rates bitmap parameter indicated by the WFD R2 Sink during capability negotiation (included in wfd2-video-formats parameter in the M3 Response message). The reference overlay resolution parameter is included in the M4 Request message during capability negotiation in the wfd2-aux-stream-formats parameter.

Graphics Data field includes the overlay graphics content. The Graphics data field may be empty when data_present_flag is set to 0, in which case, the information contained in the Private header field will be applied to the graphics data already available from previous PES packets corresponding to the Overlay ID indicated in this PES packet.

B.2.1 Blending Operation

A WFD R2 Source sends the necessary blending information in the PES packet along with the actual overlay pixel content. The Private Header in the PES packet includes the type of alpha and blending mode to be used for blending the given overlay layer with the base or background layer.

A WFD R2 Sink capable to support auxiliary streams shall support the blending modes specified in this specification. The WFD R2 Source may select an appropriate blending mode based on the content being streamed by setting the appropriate values to the blend_alpha_selection and blend_mode_selection fields in the Private Header of the PES packet. The use of blend_alpha_selection and blend_mode_selection fields are summarized in the Table 112.

Table 112. Alpha selection and associated blending modes

blend_alpha_selection Bits	Alpha Selection Type	Applicable blend_mode_selection values
0b0000	Constant Alpha	0b0001: normal alpha blending (Eq. I) 0b0010: reverse alpha blending (Eq. III)
0b0001	Per-Pixel Source Alpha	0b0001: normal alpha blending (Eq. I) 0b0010: reverse alpha blending (Eq. III)

blend_alpha_selection Bits	Alpha Selection Type	Applicable blend_mode_selection values
0b0010	Per-Pixel Destination Alpha	0b0001: normal alpha blending (Eq. I) 0b0010: reverse alpha blending (Eq. III)
0b0011	Per-Pixel Source Alpha with constant alpha modulation	0b0001: normal alpha blending (Eq. I) 0b0010: reverse alpha blending (Eq. III)
0b0100	Per-Pixel Destination Alpha with constant alpha modulation	0b0001: normal alpha blending (Eq. I) 0b0010: reverse alpha blending (Eq. III)
0b0101	Pre-multiplied per-pixel source alpha	0b0011: normal pre-multiplied alpha blending (Eq. V) 0b0100: pre-multiplied alpha blending with constant alpha modulation (Eq. VI)
0b0110 – 0b1111	Reserved	

The following blending operations are specified which are sub-set of possible Porter-Duff blending operations used for alpha compositing, further details are referred to [44].

Normal Alpha Blending Mode:

When using the normal alpha blending in which pixel color data has not been pre-multiplied with alpha, the selected ALPHA (per pixel or constant alpha) is applied to the source or foreground pixel and (1- ALPHA) is applied to the destination or background pixel colors as per the equation I below. This is also referred as (ALPHA, ONE_MINUS_ ALPHA) blending mode.

$$Pixel_{Output} = \alpha_{select} Pixel_{Src} + (1 - \alpha_{select}).Pixel_{Dest} \qquad \text{Equation I}$$

When normal alpha blending mode to be used by the WFD R2 Sink to composite the overlay graphics (as indicated by blend_mode_selection parameter), the following alpha values may be selected for blending of each pixel color (∝_select) as indicated by the bitmap of the blend_alpha_selection field:

$$\propto_{select} = \begin{cases} \propto_{CONSTANT} & \text{if } 0b0000 \\ \propto_{SRC} & \text{if } 0b0001 \\ \propto_{DEST} & \text{if } 0b0010 \\ \propto_{CONSTANT}.\propto_{SRC} & \text{if } 0b0011 \\ \propto_{CONSTANT}.\propto_{DEST} & \text{if } 0b0100 \end{cases} \qquad \text{Equation II}$$

Reverse Alpha Blending Mode:

When using reverse alpha blending (in which pixel color data has not been pre-multiplied with alpha), the (1- selected alpha) is applied to the source or foreground pixel and the selected alpha is applied to the destination or background pixel colors as per the equation III below. The α_{select} parameter may be set as in equation II. This is also referred as (ONE_MINUS_ ALPHA, ALPHA) blending mode.

$$Pixel_{Output} = (1 - \alpha_{select}).Pixel_{Src} + \alpha_{select}.Pixel_{Dest} \qquad \text{Equation III}$$

Pre-multiplied Alpha Blending Mode:

In pre-multiplied alpha blending, the Graphics content contains pixel data that has been pre-multiplied with the alpha beforehand, and thus the alpha blending just requires adding the source pixel colors (foreground pixel) with (1- ALPHA) applied to the destination or background pixel colors.

Pre-multiplied alpha blending mode is used when blend_alpha_selection field is set to 0b0101, with the ∝_select parameter as,

$$\alpha_{select} = \alpha_{SRC.pre} \; if \; 0b0101 \qquad \text{Equation IV}$$

When the blend_mode_selection field is set for normal pre-multiplied alpha blending mode, the following blend equation is used.

$$Pixel_{Output} = Pixel_{Src.Pre} + (1 - \alpha_{select}).Pixel_{Dest} \qquad \text{Equation V}$$

The constant alpha modulation may also be applied to the Graphics with pre-multiplied alpha blending. In this case, the blend_mode_selection field is set to 0b0101 and the blend equation VI is used. When setting this mode, the constant_alpha_value shall contain the constant alpha to be applied in addition to the per-pixel source alpha included in the Graphics data.

$$Pixel_{Output} = \propto_{CONSTANT} Pixel_{Src.Pre} + (1 - \alpha_{CONSTANT}) . (1 - \alpha_{select}). Pixel_{Dest} \qquad \text{Equation VI}$$

When the blend_alpha_selection parameter is set for pre-multiplied alpha, the blend_mode_selection field shall only be set to 0b0011 or 0b0100.

B.2.2 Carriage of JPEG Image over MPEG2-TS Transport Stream

The transport of JPEG is specified for sending the JPEG image data into the transport stream of MPEG2-TS in this specification.

JPEG Stream Descriptor

When the JPEG elementary stream is carried by MPEG2-TS, the JPEG stream descriptor provides the basic information for identifying the JPEG access unit. This descriptor shall be included for each JPEG elementary stream component in the PMT with stream_type equal to 0x92, and has the syntax as shown in Table 113.

Table 113. JPEG stream descriptor

Syntax	No. of bits	Mnemonic
JPEG_descriptor() {		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
image_width	16	uimsbf
image_height	16	uimsbf
For (i = 0; I < N, i++){		
private_data_byte	8	bslf
}		
}		

image_width- This field shall contain the same value as the maximum number of samples per line in the source image parameter in the JPEG frame header, as described in section B.2 of ITU T-81/ISO/IEC 10918-1 [40].

image_height- This field shall contain the same value as the maximum number of lines in the source image parameter in the JPEG frame header, as described in B.2 of ITU T-81/ISO/IEC 10918-1 [40].

JPEG Access Unit and PES Packets

The mapping of the JPEG access unit to the PES packets is shown by Figure 32 and Table 114. The PES payload contains one JPEG Access Unit. The compressed image data encoded using the JPEG interchange format [41] included in each JPEG access unit bit-stream contains all the information required to decode the image data.

The JPEG encoding for sending the image data should be fully compatible with the standard JPEG interchange format as specified in the ITU T-81/ISO/IEC 10918-1 [40]. The WFD R2 Source and WFD R2 Sink capable to support JPEG image in auxiliary stream shall support the JPEG-JFIF encoding format as specified in [41].

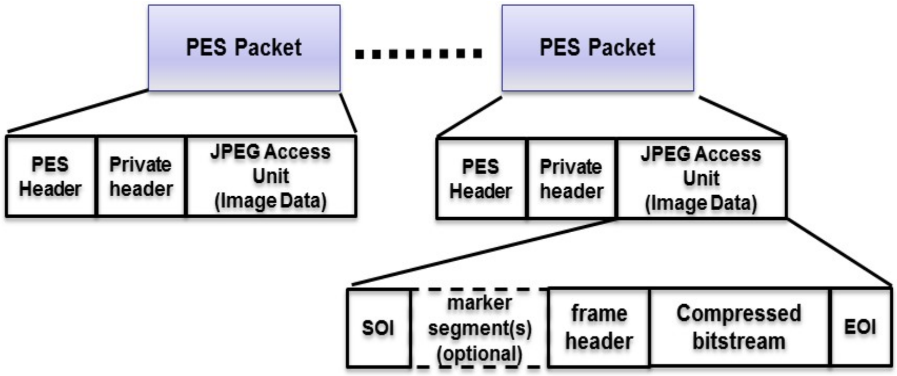


Figure 32. Structure of JPEG Access Units in PES Packets

Constraints for Transport of JPEG

The following constraints apply when transmitting a JPEG elementary stream using MPEG2-TS systems.

- The JPEG image shall always be sent over MPEG2-TS as one JPEG access unit.
- Each JPEG access unit shall include a PES header with PTS and each PES packet shall contain exactly one JPEG access unit.

The following constraints apply to the coding syntax elements in the PES header for transport of the JPEG elementary stream:

- stream_id shall be st to 0xBD (same as Private_stream_1).
- PES_packet_length shall be set to the actual length of the PES packet.
- data_alignment_indicator shall be set to '1', and the PES packet header shall be immediately followed by the first byte of the JPEG access unit. The data_stream_alignment_descriptor is optional and, if included, the alignment_type shall be set to 0x00.
- The PTS_DTS_flags shall be set to '10'.
- The PES for the JPEG image has an associated PTS and the presentation time of succeeding images, if any, should be later than that of the current access unit by at least two frame intervals of any video stream access unit sent through the same MPEG2-TS.
- All other flags should be set appropriately as shown in Table 111.

T-STD Extensions and Buffer Management

To define the decoding of the JPEG image ES carried in a transport stream, the T-STD model and T-STD parameters in [45] are extended based on the model used for the J2K still pictures in [40].

The delay of the JPEG image data through the System Target Decoder buffers TBn and EBn shall be constrained by tdn(j) – t(i) <= 60 seconds for all j, and all bytes I in access unit An(j) as in [40].

B.2.3 Carriage of PNG Image over MPEG2-TS Transport Stream

The transport of PNG is specified for sending it into the transport stream of MPEG2-TS in this specification.

PNG stream descriptor

When the PNG elementary stream is carried by MPEG2-TS, the PNG stream descriptor provides the basic information for identifying the PNG access unit. This descriptor shall be included for each PNG elementary stream component in the PMT with stream_type equal to 0x93, and has the syntax as shown in Table 114.

Table 114. PNG stream descriptor

Syntax	No. of bits	Mnemonic
PNG_descriptor() {		
descriptor_tag	8	uimsbf

Syntax	No. of bits	Mnemonic
descriptor_length	8	Uimsbf
image_width	32	Uimsbf
image_height	32	uimsbf
for (i = 0; I < N, i++){		
private_data_byte }	8	Bslf
}		

image_width- This field shall contain the same value as the width parameter in the PNG IHDR header, as described in [42].

image_height- This field shall contain the same value as the height parameter in the PNG IHDR header, as described in [42].

PNG Access Unit and PES Packets

The mapping of the PNG access unit to the PES packets is shown by Figure 33. The PES payload contains one PNG Access Unit. The PNG encoded data stream included in each PNG access unit contains all the information required to decode the PNG image. The PNG encoding and structure of the data stream are described in [48]. As illustrated in Figure 33 the encoded PNG bitstream shall start with the 8-byte long PNG signature followed by a sequence of chunks. The layout of a chunk is illustrated in Figure 34.

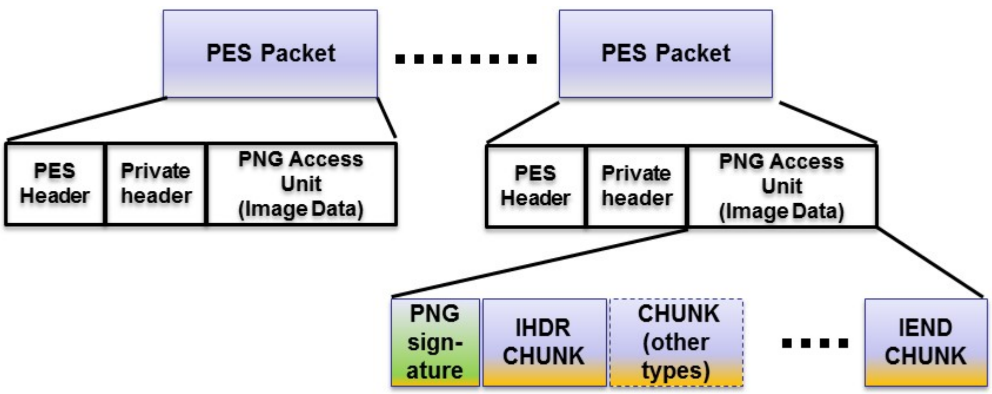


Figure 33. Structure of PNG Access Units in PES Packets

Figure 34. Layout of a Chunk

Encoding the PNG image to the contiguous PNG bit stream within the PES packet is done using the procedure described in section 7 of [42].

Constraints for Transport of PNG

The following constraints apply when transmitting a PNG elementary stream using MPEG2-TS systems

- The PNG image shall always be sent over MPEG2-TS as one PNG access unit.
- Each PNG access unit shall include a PES header with PTS and each PES packet shall contain exactly one PNG access unit.

The following constraints apply to the coding syntax elements in the PES header for transport of the PNG elementary stream:

- stream_id shall be st to 0xBD (same as Private_stream_1).
- PES_packet_length shall be set to the actual length of the PES packet.
- data_alignment_indicator shall be set to '1', and the PES packet header shall be immediately followed by the first byte of the PNG access unit. The data_stream_alignment_descriptor is optional and, if included, the alignment_type shall be set to 0x00.
- The PTS_DTS_flags shall be set to '10'.
- The PES for the PNG image has an associated PTS and the presentation time of succeeding images, if any, should be later than that of the current access unit by at least two frame intervals of any video stream access unit sent through the same MPEG2-TS.
- All other flags should be set appropriately as shown in Table 111.

T-STD Extensions and Buffer Management

To define the decoding of the PNG image ES carried in a transport stream, the T-STD model and T-STD parameters in [2] [39] are extended based on the model used for the J2K still pictures in [42].

The delay of the PNG image data through the System Target Decoder buffers TBn and EBn shall be constrained by tdn(j) – t(i) <= 60 seconds for all j, and all bytes I in access unit An(j) as in [42].

B.3 WFD R2 LPCM Formats and Packet Structures

B.3.1 PES Structure for LPCM Streams

PES structure and field values for carrying 48 ksps 16 bit 2ch and 44.1 ksps 16 bit 2ch LPCM audio data remain as specified in Table 106. The PES header fields in the Private header and audio data inside the PES payload is extended with further modifications to [5] as described in this section. The PES structure and field values for LPCM audio data applicable for WFD R2 devices is as shown in Table 115.

Table 115. PES Structure for WFD R2 LPCM audio

Field		Number of bits	Number of bytes	Value
PES Header	packet_start_code_prefix	24	3	0x00 00 01
	stream_id	8	1	0b10111101 [private stream1]
	PES_packet_length	16	2	variable
	'10'	2	1	0b10
	PES_scrambling_control	2		0b00 [not scrambled]
	PES_priority	1		0b0 [no priority]
	data_alignment_indicator	1		0b0 [not defined]
	copyright	1		0b0 [not defined]
	original_or_copy	1		0b1 [original] or 0b0 [copy]
	PTS_DTS_flag	2	1	10b [present]
	ESCR_flag	1		0b0 [not present]
	ES_rate_flag	1		0b0 [not present]
	DSM_trick_mode_flag	1		0b0 [not present]
	additional_copy_info_flag	1		0b0 [not present]
	PES_CRC_flag	1		0b0 [not present]
	PES_extension_flag	1		0b0 : if HDCP 2.x is not used

Field		Number of bits	Number of bytes	Value
				0b1 : if HDCP 2.x is used
	PES_header_data_length	8	1	0x07 : if HDCP 2.x is not used 0x18 : if HDCP 2.x is used
	'0010'	4	5	0b0010
	PTS[32…30]	3		shall be set to correct value
	marker_bit	1		0b1
	PTS[29…15]	15		shall be set to correct value
	marker_bit	1		0b1
	PTS[14…0]	15		shall be set to correct value
	marker_bit	1		0b1
	PES_private_data_flag	1	1	0b1 [present] (note-1)
	pack_header_field_flag	1		0b0 [not present] (note-1)
	program_packet_sequence _counter_flag	1		0b0 [not present] (note-1)
	P-STD_buffer_flag	1		0b0 [not present] (note-1)
	Reserved	3		0b111 (note-1)
	PES_extension_flag_2	1		0b0 [not present] (note-1)
	PES_private_data	128	16	HDCP 2.x counter values and marker_bit. See [18]. (note-1)
	stuffing bytes	16	2	0xFF FF
PES Payload Private Header	sub_stream_id	8	1	0b10100000 [0th sub stream]
	number_of_frame_header	8	1	0b00000110 [6 AUs]
	Reserved	1	1	0b0
	audio_data_type	1		0b0: if audio data for the number of channels indicated by number_of_audio_channel are included in the PES payload 0b1: if audio data for only one channel indicated by channel_index is included in the PES payload
	channel_index1	5		Channel number for the audio data, which is applicable only when audio data of a single channel is included with the audio_data_type field set to 0b1 (channel index value shall not exceed the number of audio channels indicated in wfd2-audio-codecs or wfd2-cta-audio-infoframe parameter)
	audio_emphasis_flag	1		See Table 4.5.2.1-3 of [5]
	quantization_word_length	2	1	See Table 107 (extension of Table 4.5.2.1-4 of [5]).
	audio_sampling_frequency	3		See Table 108 (extension of Table 4.5.2.1-5 of [5]).
	number_of_audio_channel	3		See Table 109 (extension of Table 4.5.2.1-6 of [5]).
PES Payload data	Audio Data (LPCM)	variable	Variable	Variable

Note:

4. The channel_index field shall be set to channel number as specified in [38] or [47] for the CTA multi-channel audio. When one channel from stereo LPCM is to be included, channel number shall be set as 0 to indicate Left channel and 1 to indicate Right channel.

The quantization_word_length, audio_Sampling_frequency, number_of_audio_channel fields are extended from [5] with the modifications as shown in Table 116, Table 117, Table 118 below. The LPCM_audio_stream_descriptor to be included in the PMT (as per Table 4.6.6.1-1 of [5]) shall also use the values for bits_per_sample based on Table 117, sampling_frequency based on Table 118, and number_of_channels based on Table 118 when transmitting LPCM audio data applicable for WFD R2 devices.

Table 116. Quantization_word_length

Value	Quantization_word_length
00b	16 bits
01b	20 bits
10b	24 bits
11b	Reserved

Table 117. Audio_sampling_frequency

Value	Audio_sampling_frequency
000b	Reserved
001b	44.1 kHz
010b	48 kHz
011b	96 kHz
100b	Reserved
101b	Reserved
110b	Reserved
111b	Reserved

Table 118. Number_of_audio_channels

Value	number_of_audio_channels[1]
000b	1 ch (mono LPCM)[2]
001b	2 channels (stereo)
010b	3 channels
011b	4 channels
100b	5 channels
101b	6 channels
110b	7 channels
111b	8 channels

Notes:

1. The number_of_audio_channels field shall indicate the actual number of audio channels to be included in a PES payload for the WFD R2 Sink, which may be lower or same as the number of LPCM channels indicated in the wfd2-audio-codecs parameter or in the wfd2-cta-audio-infoframe parameter (when CTA audio to be transmitted) sent by the WFD R2 Source in the M4 Request. For example, when streaming of 6 channel CTA LPCM audio format (for 5.1 surround audio) indicated by the wfd2-cta-audio-infoframe parameter, if one of the Sink capable to reproduce 3 channels (for Left, Right and Center speakers), the Source may send PES packet containing these 3 audio channels to that Sink by setting number_of_audio_channels field to 010b in the PES header to indicate that the PES payload will contain the 3 channels of audio data corresponding to its speaker role. Note that the wfd2-cta-audio-infoframe parameter send to any WFD R2 Sink receiving sub-set of the multi-channel audio content

Value	number_of_audio_channels[1]
in the M4 Request may be set with same audio identification and speaker channel allocation information except when byte 4 is set to 0xFF to indicate explicit channel indices when speaker location data block is available (see Table 8 in [51]). 2. In case of mono-LPCM audio data, the number_of_audio_channels field shall be set to 0b000, but audio samples are not repeated as in [5], i.e. the audio data in each AU only includes the samples of single LPCM channel.	

The PES payload carrying the LPCM audio data is included based on the following structure.

- The audio access unit (AU) is one LPCM audio frame that contains audio samples from all channels (N) to be included in the PES payload. The number of channels, N =1 for mono-LPCM or when single channel of multi-channel LPCM is carried in the PES payload, otherwise, N is greater than 1.
- Each AU always contain fixed number of audio samples/channel (M) as shown in Figure 35. The number of samples per channel, M is fixed to 80 samples per channel (M = 80) as in [5], which provides 80xNxB bits to the PES payload where N is the number of channels and B is number of bits per audio sample.
- When including CTA multi-channel LPCM audio data consisting of more than one channel (N > 1), the audio samples in the PES payload shall be in the ascending order of channel numbers corresponding to the speaker location of the WFD R2 Sink as indicated by the Audio InfoFrame included in the wfd2-cta-audio-infoframe parameter sent by the WFD R2 Source in the M4 Request. When including stereo LPCM audio data, order of samples shall be same as specified in Appendix B for 2 ch LPCM.

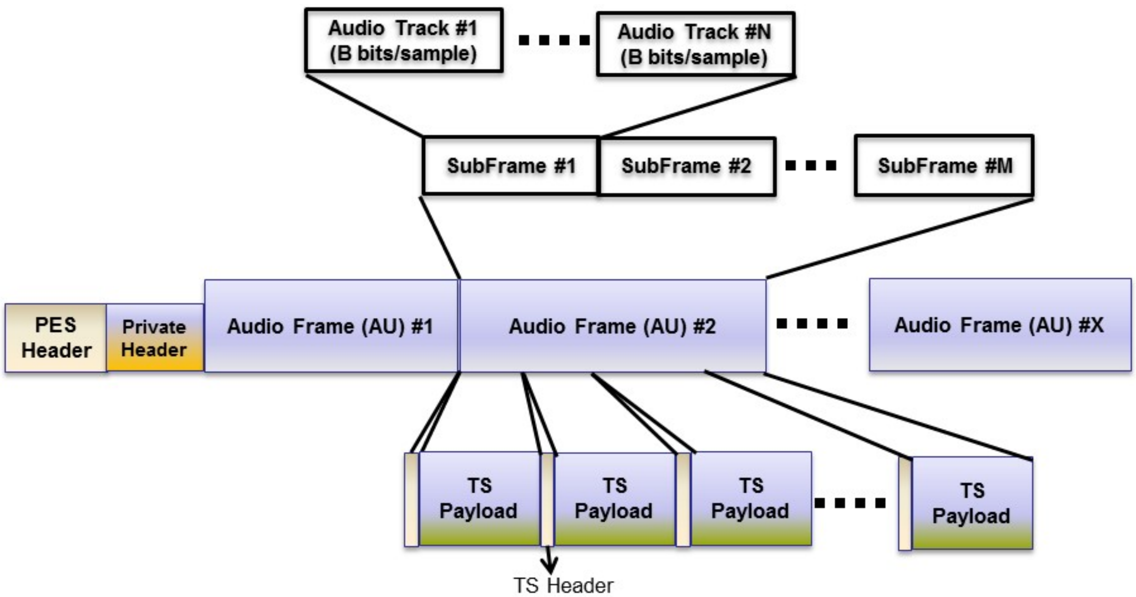


Figure 35. PES structure with X MCA frames in the payload

The LPCM frame includes:

- Scalable number of channels (tracks) N (n = 1 to N) in one frame. One LPCM frame is an Access Unit with 80 samples of audio from each channel. The number of channels included in the PES payload is indicated in the private header. When more than one channels are carried in the PES, the number of channels are indicated by the number_of_audio_channels field and the audio_data_type is set to 0b0. When the audio_data_type is set to 0b1, data from single audio channel indicated by channel index is carried in the PES packet.
- X audio frames/units (AU) are carried in one PES structure which may be determined based on the desired timing unit (e.g. 10 ms) or when audio samples are obtained for a given sampling rate. The number_of_frame_header field indicates how many AU are included.

B.3.2 Speaker Location Names used for Multichannel Audio Streaming

Speaker location names to specify WFD R2 Sinks receiving multichannel audio are specified using the speaker location naming specified in CTA-861.2 specification [47], which are consistent with those in ISO/IEC 62574 [56].

A WFD R2 Sink indicates the speaker role or location applicable to it by setting the corresponding bits of the 32 bit bitmap shown in Table 119. The details of the speaker label names and position descriptions corresponding to each CTA-861-G Code are referred to Table 34 of [47]. All unused speaker positions shall always be set to zero.

Table 119. Bitmap for the Speaker Location Names

Label	Position Description	Code
FL	Front Left	0x00
FR	Front Right	0x01
FC	Front Center	0x02
LFE1	Low Frequency Effects 1	0x03
BL	Back Left	0x04
BR	Back Right	0x05
FLc	Front Left of Center	0x06
FRc	Front Right of Center	0x07
BC	Back Center	0x08
LFE2	Low Frequency Effects 2	0x09
SiL	Side Left	0x0A
SiR	Side Right	0x0B
TpFL	Top Front Left	0x0C
TpFR	Top Front Right	0x0D
TpFC	Top Front Center	0x0E
TpC	Top Center	0x0F
TpBL	Top Back Left	0x10
TpBR	Top Back Right	0x11
TpSiL	Top Side Left	0x12
TpSiR	Top Side Right	0x13
TpBC	Top Back Center	0x14
BtFC	Bottom Front Center	0x15
BtFL	Bottom Front Left	0x16
BtFR	Bottom Front Right	0x17
FLw	Front Left Wide	0x18
FRw	Front Right Wide	0x19
LS	Left Surround	0x1A
RS	Right Surround	0x1B
	reserved	0x1C
	reserved	0x1D
	reserved	0x1E
	reserved	0x1F

Speaker location names may correspond to the Room Configuration Descriptor (RCD), which are conveyed in the wfd2-cta-datablock-collection RTSP parameter. This may allow the conveyance of a specific location in a defined room with specific

B.3.3 Multi-channel Audio Formats Information

The multi-channel audio format to be streamed by a WFD R2 Source is indicated by using the Audio InfoFrame Format as shown in Table 28 or Table 30 of CTA-861-G [47] as applicable.

The mapping of speaker positions to transmission channel allocations are specified by byte 4 of the Audio InfoFrame format when using speaker placement using CTA-861-G for up to 8 LPCM channels as shown in Table 35 of [47]. For higher number of audio channels for 3D LPCM Audio (Audio Coding Extension Type Code 0x0D) or other audio formats applicable in this specification, channel allocation information using the channel assignment indicated as 0xFF shall be used.

B.4 DTS-HD Audio Bitstream Constraints

DTS-HD audio streams are constructed in accordance with ETSI TS 102 114v.1.4.1 [55], and may consist of either:

- a core substream
- one or more extension substreams
- both a core and one or more extension substream

In the case of a core+extension substream, the core component always precedes the first extension component in a given audio frame.

DTS-HD audio streams are constrained according to DECE DMedia specification as described in sections 5.7.2.2 and Table B-5 of [56].

B.4.1 DTS-HD MPEG-TS Parameters and Requirements

Buffering Model

The DTS buffering model is designed in accordance with ISO/IEC 13818-1. Refer to the derivation of BSn for audio elementary streams.

- For DTS core streams, the main audio buffer size (BSn) has a fixed value of 9088 bytes, with a drain rate (Rxn) of 2 Mbps. The fixed value above (9088 bytes) was calculated from a double buffer (2*4096 bytes) plus jitter (384 bytes) + packet bursts (512 bytes).
- For DTS-HD Lossless formats, the value of BSn shall have a fixed value of 66 432 bytes, with an Rxn value of 32 Mbps.
- For all other DTS-HD formats, the value of BSn shall have a fixed value of 17 814 bytes, with an Rxn value of 8 Mbps.

Stream Type

MPEG-2 TS streams containing DTS-HD shall use Stream Type = 0x06.

Registration Descriptor

A Registration Descriptor for DTS-HD is not needed.

DTS-HD Audio Stream Descriptor

The full DTS-HD descriptor described in SCTE 194-2 may be used to signal DTS-HD audio streams, but for the purposes of WFD R2 systems, this is not necessary. A simplified audio stream descriptor may be used to signal DTS-HD, as described below:

```
DTS-HD_audio_stream_descriptor() {
 bit(8)  descriptor_tag;    // 0x7B
 bit(8)  descriptor_length;
 for (i=0; i<N; i++)
```

```
        bit(8)  additional_info_byte[i]; // Additional bytes to pad out to descriptor_length (N
    may be 0)
    }
```

descriptor_tag

The descriptor_tag for DTS-HD shall be set to 0x7B.

B.4.2 DTS-HD Elementary Stream Encapsulation

Stream ID

For DTS-HD streams in WFD R2, stream_id shall be set to 0xBD indicating private stream 1.

Data_Alignment_Indicator

The Data_Alignment_Indicator flag shall be set to 1 to indicate that each PES packet is aligned to a valid sync word.

PES Encapsulation

The PES encapsulation of DTS-HD elementary streams shall be aligned with the DTS-HD audio frames, which always begin with a syncword. To verify that the bitstream to be encapsulated is synchronized, (and that the Syncword is not an aliased sequence of compressed audio data), the frame size parameter of in the audio frame header can be used to verify to number of bytes to the next sync word.

Valid syncwords for the DTS-HD stream type are indicated in Table 120.

Table 120. Syncwords associated with DTS-HD audio

Name	sync word	description
DTS_SYNCWORD_CORE	0x7ffe8001	core substream
DTS_SYNCWORD_SUBSTREAM	0x64582025	extension substream

For the DTS core substream component of an audio frame, the number of bytes in the core is stored in the core frame header in the parameter FSIZE. FSIZE is described in [55].

For the DTS extension substream component of an audio frame, the number of bytes in the extension is stored in the extension frame header in the parameter nuExtSSFsize. nuExtSSFsize is described in [55].

Note that a maximum of 4 extension substream components are possible, and nuExtSSFsize shall be extracted from each substream component comprising an audio frame.

The sum of the bytes of all present core and extension substream components shall be summed to determine the number of bytes in the PES payload.

Figure 36 illustrates the typical composition of a PES payload.

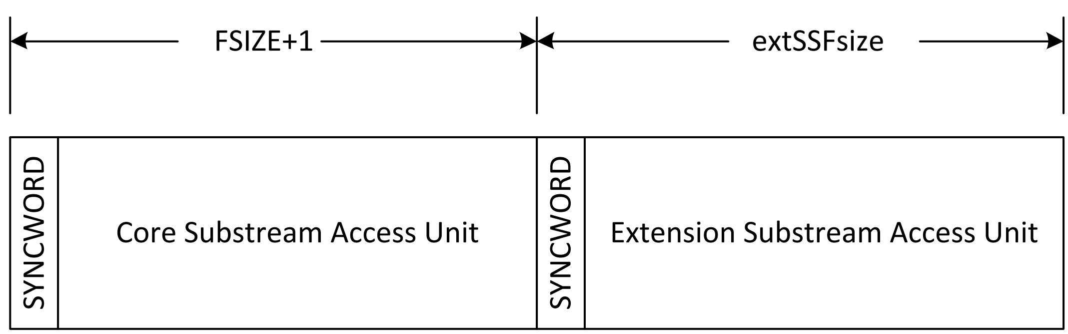


Figure 36. Typical composition of a PES payload

Appendix C Recommendations for Satisfying the HDCP 2.x Locality Check (informative)

This appendix provides a recommended parameter setting at the TCP/IP layer for Locality Check of the HDCP system 2.x (if applicable).

The HDCP system 2.x requires successful completion of Locality Check prior to initiating the Session Key Exchange phase. The Locality Check requires that the round trip time shall be within 7 milliseconds for a WFD Source to transmit a TCP packet containing a 64-bit pseudo random nonce (LC_Init message) to the WFD Sink and to receive a HMAC-SHA256 digest as a corresponding Response from the WFD Sink.

The Locality Check is described in Cl.2.3 of [18] and clarified in the second item of the Errata [19] for HDCP system 2.0, it is described in Cl2.3 of [27] for HDCP system 2.1, and it is described in Cl2.3 of [38] for HDCP system 2.2. For higher version than HDCP 2.2, refer to corresponding specification released by DCP LLC. This section describes a set of recommended optimizations in order to expedite the transit of LC_Init and LC_Send_L_Prime messages between the WFD Source and WFD Sink.

Some of the optimizations are possible in specific implementations of the TCP stack.

A WFD Device should provide expedited processing of the HDCP locality check data [21] for HDCP system 2.0 or [30] for the HDCP system 2.1 or [35] for HDCP system 2.2 at the TCP, IP and MAC layers (for higher version than HDCP 2.2, refer to corresponding specification released by DCP LLC). When the WFD Source transmits the LC_Init message or the WFD Sink transmits the LC_Send_L_prime message without using pre-compute option:

a. at the TCP layer, the URG and PSH flags should be set to “1”,
b. at the IP layer, the TOS field of the IP header should be configured to provide expedited forwarding, low delay and low drop probability,
c. at the MAC layer, WMM access category should be set to AC_VI.
d. at the PHY/MAC layer, reduced MCS should be used to reduce transmission errors and to provide highest reliability to the locality check data.

Appendix D H.264 and H.222 Usage Detail (normative)

This appendix specifies rules for H.264 and H.222 usage at this specification. Restricting rules as specified in this appendix shall be applied to H.264 and H.222 specifications.

D.1 Slice Usage

Except as specified in the following section D.1.1, a slice shall be a picture in this specification.

D.1.1 Multiple Slices in a Picture

If the min-slice-size field in the wfd-video-formats parameter or the wfd-3d-formats parameter transmitted by the targeted WFD Sink has non-zero value, WFD Source may transmit a picture that is constructed by multiple slices. In this case, each slice in a picture shall include the equal number of macroblocks.

The minimum number of macroblocks in a slice shall be greater than or equal to the number of macroblocks indicated in the min-slice-size field of the wfd-video-formats parameter, the wfd-3d-formats or the wfd-preferred-display-mode parameter from the WFD Sink.

The maximum number of slices per a picture shall not exceed the limit as specified in [1].

In the Max SliceNum bits (B9:B0) transmitted using an RTSP M4 (RSTP SET_PARAMETER) Request message, the WFD Source shall not set a value exceeding the value of Max SliceNum bits (B9:B0) that is informed by the WFD Sink to the WFD Source. In the Max SliceNum bits (B9:B0) transmitted using an RTSP M4 (RSTP SET_PARAMETER) Request message, the WFD Source may set a value independent on the value of Max Slice Size Ratio bits (B12:B10) that is informed by the WFD Sink to the WFD Source.

In the Max Slice Size Ratio bits (B12:B10) transmitted using an RTSP M4 (RTSP SET_PARAMETER) Request message, the WFD Source shall not set a value exceeding the value of Max Slice Size Ratio bits (B12:B10) that is informed by the WFD Sink to the WFD Source. In the Max Slice Size Ratio bits (B12:B10) transmitted using an RTSP M4 (RTSP SET_PARAMETER) Request message, the WFD Source may set a value independent on the value of Max SliceNum bits (B9:B0) that is informed by the WFD Sink to the WFD Source.

If a WFD Source tries to change the video resolution using an RTSP M4 (SET_PARAMETER) Request message, the WFD Source should not change the values in Slice encoding parameters from the previously determined values, as far as the new values in Slice encoding parameters are not conflict with the other rules in this specification.

If a WFD Source transmits video elementary stream where one picture is constructed by multiple slices;

- A picture is constructed by I-slice(s) and/or P-slices(s)
- I-slice includes intra-macroblock only
- P-slice includes inter-macroblock and may include intra-macroblock(s)

A WFD Source should determine averaged encoded video data rate not to exceed the value indicated in the WFD Device Maximum throughput field at WFD Device Information subelement transmitted by the targeted WFD Sink, with taking into account an expected end-to-end latency. As a result, the WFD Sink can buffer/decode the encoded video data at reasonable latency.

D.2 Frame Packing Arrangement SEI

When transmitting 3D video, the WFD Source shall include the Frame packing arrangement SEI within a SEI NAL unit at all first Access Units within a GOP.

If the transmitting 3D video uses frame sequential method (frame_packing_arrangement

_type = 5), the WFD Source shall include the Frame packing arrangement SEI within a SEI NAL unit at all Access Units.

This SEI shall not be included in a video elementary stream that contains 2D video, except for the following case;

When switching from 3D video to 2D video happens, this SEI shall exist at all Access Units with setting frame_packing_arrangement_cancel_flag to one, at least 2 seconds after switching from 3D to 2D. It implies that switching from 2D video to 3D video is disallowed at least 2 seconds after switching 3D video to 2D video. When the switching from 3D video to 2D video happens, the frame that starts carrying 2D video slice data and the frame that switches frame_packing_arrangement_cancel_flag value from zero to one in this SEI should be identical.

The syntax element values in this SEI are specified in Table 121 and other values shall not be used.

Table 121. Frame packing arrangement SEI syntax element and values at Wi-Fi Display

Syntax elements	Values
frame_packing_arrangement_id	0 always
frame_packing_arrangement _cancel_flag	0 : frame packing arrangement information follows. When with transmitting 3D video data, it shall be always 0. 1 : cancel the persistence of any previous frame packing arrangement SEI message in output order from decoder. When with transmitting 2D video data and if this SEI exists, this element shall be set to 1.
frame_packing_srrangement _type	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 3 : Side-by-Side 4 : Top & Bottom [1] 5 : Frame Sequential
quincunx_sampling_flag	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 0 always (quincunx sampling is not used)
content_interpretation_type	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 1 always (left half of the frame (frame0) is for left view top half of the picture (frame0) is for left view odd picture (frame0) is for left view)
spatial_flipping_flag	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 0 always (no flipping shall be applied)
frame0_flipping_flag	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 0 always (no flipping shall be applied)
field_views_flag	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 0 always
current_frame_is_frame0_flag	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 0 : if frame_packing_srrangement_type is 3 or 4, or right view if frame_packing_srrangement _type is 5. 1 : left view if frame_packing_srrangement _type is 5
frame0_self_contained_flag	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 0 always (the samples of constituent frame 0 may or may not refer to samples of some constituent the frame 1.)
frame1_self_contained_flag	[This element exists only when frame_packing_ arrangement_cancel_flag = 0] 0 always (the samples of constituent frame 1 may or may not refer to samples of some constituent frame 0.)

Syntax elements	Values
frame0_grid_position_x	[This element exists only when frame_packing_ arrangement_cancel_flag = 0, and frame_packing_srrangement_type =is 3 or 4.] 0 always (horizontal location of the upper left sample of constituent frame 0)
frame0_grid_position_y	[This element exists only when frame_packing_ arrangement_cancel_flag = 0, and frame_packing_srrangement_type =is 3 or 4.] 0 always (vertical location of the upper left sample of constituent frame 0)
frame1_grid_position_x	[This element exists only when frame_packing_ arrangement_cancel_flag = 0, and frame_packing_srrangement_type =is 3 or 4.] 0 always (horizontal location of the upper left sample of constituent frame 1.)
frame1_grid_position_y	[This element exists only when frame_packing_ arrangement_cancel_flag = 0, and frame_packing_srrangement_type =is 3 or 4.] 0 always (vertical location of the upper left sample of constituent frame 1.)
frame_packing_arrangement _reserved_byte	0 : reserved
frame_packing_arrangement _repetition_period	0 : this SEI message shall be applied to the current decoded frame only. In the case of frame_packing_srrangement_type = 5, this value shall be always zero. 1: message persists. See ref [1]
frame_packing_arrangement _extension_flag	0 : reserved
Notes: 1. To distinguish Top and Bottom [half] and Frame Packing, video format confirmed during the Capability Negotiation and/or pic_height_in_map_units_minus1 in SPS are used.	

D.3 SEI and VUI

A WFD Source may include SEIs and VUI that are defined in [1] into Access Unit.

A Primary Sink shall decode video streams with discarding SEIs or VUI that are not supported.

In the case of 3D video, the Frame packing arrangement SEI is used as described in Appendix D.2 and the Stereo video information SEI shall not be used.

D.4 H.222 usage

D.4.1 PTS/DTS for video stream

A WFD Source shall not use AVC 24-hour picture, during a WFD Session.

A WFD Source shall use exactly one PES payload to carry one video Access Unit, during a WFD Session.

As a result, a WFD Sink can derive PTS and DTS (if applicable) information from the PES header for each AU.

D.4.2 PAT/PMT

During a WFD Session and if a WFD Source is not in WFD Standby mode, the WFD Source shall transmit PAT/PMT repeatedly with a maximum time interval of 100 msec between repetition.

The following PID values shall be used for Wi-Fi Display:

- program_map_PID in PAT to indicate PID for PMT : 0x01 00
- PCR_PID in PMT to indicate PID for PCR : 0x10 00
- elementary_PID in PMT to indicate PID for video stream : 0x10 11
- elementary_PID in PMT to indicate PID for audio stream : 0x11 00 to 0x11 1F
    - If the transport stream contains one or more audio streams, the PID values of the audio streams shall be numbered consecutively, starting from 0x11 00.
- elementary_PID in PMT to indicate PID for auxiliary stream : 0x11 20 to 0x11 2F
    - If the transport stream contains one or more auxiliary streams, the PID values of the auxiliary streams shall be numbered consecutively, starting from 0x11 20.

D.4.3 Program and program element descriptors in PMT

AVC timing and HRD descriptor

The AVC timing and HRD descriptor provides timing and HRD parameters of the associated AVC video stream. The descriptor_tag for the AVC timing and HRD descriptor is “42” (0x2A) as specified in section 2.6.1 of [2].

A WFD Source shall include the AVC timing and HRD descriptor in PMT when transmitting video stream to the Primary Sink, independent on the inclusion of VUI in SPS.

If a WFD Source sets the HRD_management_valid_flag, it shall set the “initial_cpb_removal_delay” and “cpb_removal_delay” in Buffer period SEI and Picture Timing SEI respectively, as specified in section 2.6.67 of [2].

The WFD Sink may use the information in “initial_cpb_removal_delay” and “cpb_removal_delay” for encoded video removal.

Other descriptors

Settings of the AVC video descriptor, the AC-3 Audio descriptor and the AAC audio descriptor are described in Table 105.

If a WFD Source includes a video elementary stream in MPEG2-TS, the WFD Source should include descriptors listed below in the PMT. All of these descriptors are intended to be used together. So, if one of these three descriptors is present, it is recommended that other two descriptors are also present.

- video stream descriptor

    This descriptor is defined in section 2.6.2 and 2.6.3 of [2] and its descriptor_tag is “2” (0x02) as specified in section 2.6.1 of [2]. To allow early display initialization, the values in this descriptor should always match negotiated frame rate indicated in a wfd-video-formats parameter as defined in section 6.1.3 (or a wfd-3d-formats parameter as defined in section 6.1.4, or a wfd-preferred-display-mode parameter as defined in section 6.1.14).

    multiple_frame_rate_flag: This 1-bit field is set to '1' to indicate that multiple frame rates may be present in the video stream. This 1-bit field is set to '0' to indicate that only a single frame rate is present.

    frame_rate_code: This is a 4-bit field as defined in section 6.3.3 of [9], except that when the multiple_frame_rate_flag is set to a value of '1' the indication of a particular frame rate also permits certain other frame rates to be present in the video stream, as specified in Table 2-47 in [2]. In referred table, the frame_rate_code defines possible frame rates for each frame_rate_code.

    MPEG_1_only_flag: A WFD Sink shall ignore the value of this flag.

    constrained_parameter_flag: A WFD Sink shall ignore the value of this flag.

    still_picture_flag: A WFD Sink shall ignore the value of this flag.

    profile_and_level_indication: A WFD Sink shall ignore the value of this flag.

- target background grid descriptor

    This descriptor is defined in section 2.6.12 and 2.6.13 of [2] and its descriptor_tag is “7” (0x07) as specified in section 2.6.1 of [2]. Inclusion of this descriptor allows early display initialization. This descriptor is for the "virtual screen" and the offset where the image should be placed (used for PiP or letterboxing). Basically where to decode the video window in terms of the screen (refer to Figure 2-3 in [2]).

Note: Usually the offset is zero because the virtual screen is as big as the picture itself

horizontal_size: The horizontal size of the target background grid in pixels.

vertical_size: The vertical size of the target background grid in pixels.

aspect_ratio_information: Specifies the sample aspect ratio or display aspect ratio of the target background grid.

- video window descriptor

  This descriptor is defined in section 2.6.14 and 2.6.15 of [2] and its descriptor_tag is “8” (0x08) as specified in section 2.6.1 of [2]. Inclusion of this descriptor allows early H.264 decoder initialization. The video window descriptor is used to describe the window characteristics of the associated video elementary stream. Values in this descriptor reference the target background grid descriptor for the same stream. Refer to Figure 2-3 in [1] for the horizontal_offset and vertical_offset fields. These fields allow for determining the width x height of the H.264 layer defined in PPS.

Appendix E RTSP message examples (informative)

This appendix gives examples of RTSP messages.

E.1 From RTSP M1 to M7 without errors

Table 122 gives example of RTSP messages from M1 to M7 without errors and M16 message exchange, between the WFD Source and the Primary Sink. Please note that there is no Secondary Sink in this example.

Also, this is an example for multiplexed video and audio streaming.

<example>

1. i: Initial value of CSeq from a WFD Source to a WFD Sink
2. j: Initial value of CSeq from a WFD Sink to a WFD Source (j may or may not be equal to i)
3. Bold letter in an RTSP M3 Request message indicates the mandatory parameter in this message.
4. "Date" line in the header is optional
5. In this example, time-out value of keep-alive function is set to 30 seconds at an RTSP M6 Response.

Table 122. RTSP message example-1

Message ID	req/res (direction)	RTSP message

M1	Request
	(src →snk)
		`OPTIONS * RTSP/1.0`
		`CSeq: i`
		`Require: org.wfa.wfd1.0`

	Response
	(snk→src)
		`RTSP/1.0 200 OK`
		`CSeq: i`
		`Date: Sun, Aug 21 2011 04:20:53 GMT`
		`Public: org.wfa.wfd1.0, GET_PARAMETER, SET_PARAMETER`

M2	Request
	(snk →src)
		`OPTIONS * RTSP/1.0`
		`CSeq: j`
		`Require: org.wfa.wfd1.0`

	Response
	(src→snk)
		`RTSP/1.0 200 OK`
		`CSeq: j`
		`Date: Sun, Aug 21 2011 04:20:53 GMT`
		`Public: org.wfa.wfd1.0, SETUP, TEARDOWN, PLAY, PAUSE,`
		`GET_PARAMETER, SET_PARAMETER`

M3	Request
	(src →snk)
		`GET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0`
		`CSeq: i+1`
		`Content-Type: text/parameters`
		`Content-Length: 141`
		
		`wfd_video_formats`
		`wfd_audio_codecs`
		`wfd_3d_video_formats`
		`wfd_content_protection`
		`wfd_display_edid`
		`wfd_coupled_sink`
		`wfd_client_rtp_ports`

	Response
	(snk→src)
		`RTSP/1.0 200 OK`
		`CSeq: i+1`
		`Content-Length: 290`
		`Content-Type: text/parameters`
		
		`wfd_video_formats: 00 00 01 01 00000001 00000000`
		`00000000 00 0000 0000 00 none none`

Message ID	req/res (direction)	RTSP message
		wfd_audio_codecs: LPCM 00000003 00 wfd_3d_video_formats: none wfd_content_protection: none wfd_display_edid: none wfd_coupled_sink: none wfd_client_rtp_ports: RTP/AVP/UDP;unicast 1028 0 mode=play
M4	Request (src →snk)	SET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: i+2 Content-Type: text/parameters Content-Length: 302 wfd_video_formats: 00 00 01 01 00000001 00000000 00000000 00 0000 0000 00 none none wfd_audio_codecs: LPCM 00000002 00 wfd_presentation_URL: rtsp://10.82.24.140/wfd1.0/streamid=0 none wfd_client_rtp_ports: RTP/AVP/UDP;unicast 1028 0 mode=play
	Response (snk→src)	RTSP/1.0 200 OK CSeq: i+2
M5	Request (src →snk)	SET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: i+3 Content-Type: text/parameters Content-Length: 27 wfd_trigger_method: SETUP
	Response (snk→src)	RTSP/1.0 200 OK CSeq: i+3
M6	Request (snk →src)	SETUP rtsp://10.82.24.140/wfd1.0/streamid=0 RTSP/1.0 CSeq: j+1 Transport: RTP/AVP/UDP;unicast;client_port=1028
	Response (src→snk)	RTSP/1.0 200 OK CSeq: j+1 Session: 6B8B4567;timeout=30 Transport: RTP/AVP/UDP;unicast;client_port=1028;server_port=5000
M7	Request (snk →src)	PLAY rtsp://10.82.24.140/wfd1.0/streamid=0 RTSP/1.0 CSeq: j+2 Session: 6B8B4567
	Response (src→snk)	RTSP/1.0 200 OK Date: Sun, Aug 21 2011 04:20:53 GMT CSeq: j+2
(there may be other message exchanges)		
M16	Request (src →snk)	GET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: i+4 Session: 6B8B4567
	Response (snk→src)	RTSP/1.0 200 OK Cseq: i+4

Note that session ID line is not mandatory in the header of RTSP Request messages transmitted by the RTSP server (in this case, WFD Source). The example shown above for an RTSP M16 Request massage includes session ID line but it is optional.

E.2 RTSP M4 with error case

Table 123 gives an example of RTSP M4 message error case when two parameters (in this case, unsupported audio format and unsupported level for video are requested to set).

The WFD Sink responds in an RTSP M3 Response message indicating only formats and settings that it can accept in following RTSP M4 message exchange at that time, as specified in section 6.4.3. While another RTSP M4 attempt is acceptable according to section 6.4.3 that the RTSP M4 Request message can be sent at any time, this should generally be treated as a non-recoverable failure to connect and followed up with a teardown trigger from the WFD Source, if the WFD Sink returns an RTSP M4 Response message indicating an error even though the selected format and setting in the RTSP M4 Request message has been chosen to be consistent with the RTSP M3 Response message.

Table 123. RTSP message example-2

Message ID	req/res (direction)	RTSP message
M4	Request (src →snk)	SET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: i Content-Type: text/parameters Content-Length: 302 wfd_video_formats: 00 00 01 11 00000001 00000000 00000000 00 0000 0000 00 none none wfd_audio_codecs: LPCM 00000000 00 wfd_presentation_URL: rtsp://10.82.24.140/wfd1.0/streamid=0 none wfd_client_rtp_ports: RTP/AVP/UDP;unicast 1028 0 mode=play
	Response (snk→src)	RTSP/1.0 303 See Other CSeq: i Content-Type: text/parameters Content-Length: 47 wfd_video_formats: 457 wfd_audio_codecs: 415

E.3 Example RTSP Messages for TCP Mode

Table 124 though Table 128 give an example of RTSP messages for handling TCP mode and transport switch between TCP and UDP transport.

Table 124. RTSP M3 message to query available buffer length

Message ID	req/res (direction)	RTSP message
M3	Request (src →snk)	GET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: 8 Content-Type: text/parameters Content-Length: x wfd_video_formats wfd_audio_codecs wfd_3d_video_formats wfd_content_protection wfd_display_edid wfd2_buffer_length
	Response (snk→src)	RTSP/1.0 200 OK CSeq: 8 Content-Length: y Content-Type: text/parameters

Message ID	req/res (direction)	RTSP message
		wfd_video_formats: 00 00 01 01 00000001 00000000 00000000 00 0000 0000 00 none none wfd_audio_codecs: LPCM 00000003 00 wfd_3d_video_formats: none wfd_content_protection: none wfd_display_edid: none wfd_coupled_sink: none wfd2_buffer_length: 4000

Table 125. RTSP M4 and M6 messages for switching to TCP

Message ID	req/res (direction)	RTSP message
M4	Request (src →snk)	SET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: i Session: 6B8B4567 Content-Type: text/parameters Content-Length: x wfd2_transport_switch: RTP/AVP/TCP;unicast mode=play wfd2_buffer_length: 3000
	Response (snk→src)	RTSP/1.0 200 OK CSeq: i Session: 6B8B4567 Content-Length: y Content-Type: text/parameters wfd2_buffer_length: 2000 Note: If the Sink cannot support the requested amount of buffer, it returns a different value in the wfd2-buffer-length parameter in the Response message with the status code 200.
M6	Request (snk →src)	SETUP rtsp://10.82.24.140/wfd1.0/streamid=0 RTSP/1.0 CSeq: j Transport: RTP/AVP/TCP;unicast;client_port=19006-19007
	Response (src→snk)	RTSP/1.0 200 OK CSeq: j Session: 6B8B4567;timeout=30 Transport: RTP/AVP/TCP;unicast;client_port=19006-19007;server_port=23004-23005
Notes: 1. The wfd2-buffer-length parameter is optional when requesting a transport switch.		

Table 126. RTSP M4 and M6 message for switching to UDP

Message ID	req/res (direction)	RTSP message
M4	Request (src →snk)	SET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: i+1 Session: 6B8B4567 Content-Type: text/parameters

Message ID	req/res (direction)	RTSP message
		Content-Length: x Wfd2_transport_switch: RTP/AVP/UDP;unicast mode=play wfd2_buffer_length: 150
	Response (snk→src)	RTSP/1.0 200 OK CSeq: i+1
M6	Request (snk →src)	SETUP rtsp://10.82.24.140/wfd1.0/streamid=0 RTSP/1.0 CSeq: j+1 Transport: RTP/AVP/UDP;unicast;client_port=1028-1029
	Response (src→snk)	RTSP/1.0 200 OK CSeq: j+1 Session: 6B8B4567;timeout=30 Transport: RTP/AVP/UDP;unicast;client_port=1028-1029;server_port=5000-5001
Notes: 1. The wfd2-buffer-length parameter is optional when requesting a transport switch.		

Table 127. RTSP M3 message for requesting audio buffer information

req/res (direction)	RTSP message

Request (src →snk)	`GET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0`
	`CSeq: 13`
	`Session: 6B8B4567`
	`Content-Type: text/parameters`
	`Content-Length: x`
	
	`wfd2_audio_playback_status`

Response (snk→src)	`RTSP/1.0 200 OK`
	`CSeq: 13`
	`Session: 6B8B4567`
	`Content-Length: y`
	`Content-Type: text/parameters`
	
	`wfd2_audio_playback_status:  1200 468000`

Table 128. RTSP M3 message for requesting video buffer information

req/res (direction)	RTSP message

Request (src →snk)	`GET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0`
	`CSeq: 13`
	`Session: 6B8B4567`
	`Content-Type: text/parameters`
	`Content-Length: x`
	
	`wfd2_video_playback_status`

Response (snk→src)	`RTSP/1.0 200 OK`
	`CSeq: 13`
	`Session: 6B8B4567`
	`Content-Length: y`

req/res (direction)	RTSP message

	`Content-Type: text/parameters`
	
	`wfd2_video_playback_status: 2400 468600`

Appendix F HDCP 2.x Notation Mapping to WFD Device HDCP Implementation Revision

Table 129 provides necessary mapping of the HDCP2.x notation to the appropriate HDCP implementation revision number, allowing the correct specifications release version to be identified. Future releases of HDCP specifications can be added to this table to facilitate access to relevant specifications.

Table 129. HDCP 2.x Notation Mapping

WFD Device Implementation Revision	HDCP System 2.x mapping	HDCP 2.x mapping	Specifications References
V2.0	HDCP System 2.0	HDCP2.0	[18], [19]
V2.1	HDCP System 2.1	HDCP2.1	[27]
V2.2	HDCP System 2.2	HDCP 2.2	[34]
Future Revisions			

Appendix G CTA Audio Example (informative)

An example of the RTSP messages to exchange CTA parameters between a WFD R2 Source and a WFD R2 Sink is shown in Table 130.

Table 130. Example of RTSP messages to exchange CTA Audio parameters between WFD R2 Source and WFD R2 Sink

Message ID	req/res (direction)	RTSP message
M3	Request (src →snk)	GET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: i+1 Content-Type: text/parameters Content-Length: 141 wfd2-audio-codecs wfd2-cta-datablock-collection
	Response (snk→src)	RTSP/1.0 200 OK CSeq: i+1 Content-Length: 290 Content-Type: text/parameters wfd_audio_codecs: LPCM 00000003 00 CTA 00000001 00 wfd2-cta-datablock-collection : 00422C1504505E04005F56017B040D7B13630500000F0F0A000000E714200021012202
M4	Request (src →snk)	SET_PARAMETER rtsp://localhost/wfd1.0 RTSP/1.0 CSeq: i+2 Content-Type: text/parameters Content-Length: 302 wfd_audio_codecs: CTA wfd2-cta-audio-infoframe: 001A 04010AF20D0DFF000700000000
	Response (snk→src)	RTSP/1.0 200 OK CSeq: i+2

The WFD R2 Sink's M3 Response contains the wfd2-cta-datablock-collection payload. The following tables describe the CTA Audio data blocks that make up the payload.

The example wfd2-cta-datablock-collection payload, "2C1504505704005F56017B040DEB13630500000F0F0A000000E714200021012202", is a concatenation of an Audio Data Block, a Room Configuration Data Block and a Speaker Location Data Block as shown in Table 131 through Table 139.

The example wfd2-cta-audio-infoframe payload "04010AF20D0DFF000700000000" in the WFD R2 Source's M4 Request is shown in Table 138.

Audio Data Block

This Audio Data Block example shows that the example WFD R2 Sink supports 48 kHz L-PCM, AC-3, Enhanced AC-3 and DTS-HD.

Data Block Header

Hex Equivalent: 0x2C

Table 131. Data Block Header Example

Bits:	7	6	5	4	3	2	1	0
Byte 1	Audio Data Block Tag = 1 (0b001)			Length of following in bytes = 12				

AC-3 Short Audio Descriptor

Hex Equivalent: 0x150450

Table 132. AC-3 Short Audio Descriptor Example

Bits:	7	6	5	4	3	2	1	0
Byte 1	F17=0	AC-3 Audio Format Code = 0b0010				Max Number Channels -1 = 5		
Byte 2	F27=0	192kHz=0	176.4kHz=0	96kHz=0	88.2kHz=0	48kHz=1	44.1kHz=0	32kHz=0
Byte 3	Maximum bit rate divided by 8 kHz = 80							

Enhanced AC-3 Short Audio Descriptor

Hex Equivalent: 0x570400

Table 133. Enhanced AC-3 Short Audio Descriptor Example

Bits:	7	6	5	4	3	2	1	0
Byte 1	F17=0	Enhanced AC-3 Audio Format Code = 0b1010				Max Number Channels -1 = 7		
Byte 2	F27=0	192kHz=0	176.4kHz=0	96kHz=0	88.2kHz=0	48kHz=1	44.1kHz=0	32kHz=0
Byte 3	Reserved = 0							

DTS-HD Short Audio Descriptor

Hex Equivalent: 0x5F5601

Table 134. DTS-HD Short Audio Descriptor Example

Bits:	7	6	5	4	3	2	1	0
Byte 1	F17=0	DTS-HD Audio Format Code = 0b1011				Max Number Channels -1 = 7		
Byte 2	F27=0	192kHz=1	176.4kHz=0	96kHz=1	88.2kHz=0	48kHz=1	44.1kHz=1	32kHz=0
Byte 3	Codec Dependent (Typically = 0x01 for DTS-HD)							

L-PCM 3D Short Audio Descriptor (Arbitrarily set for maximum of 3 channels, e.g. a Soundbar)

Hex Equivalent: 0x7B040D

Audio Data Block, Hex Equivalent: 0x2C 150450 570400 5F5601 7B040D

Table 135. L-PCM 3D Short Audio Descriptor Example

Bits:	7	6	5	4	3	2	1	0
Byte 1	MC3=0	Extended Audio Format Code = 0b1111				MC2=0	MC1=1	MC0=1
Byte 2	MC4=0	192kHz=0	176.4kHz=0	96kHz=0	88.2kHz=0	48kHz=1	44.1kHz=0	32kHz=0
Byte 3	Reserved = 0			CXT0-CXT4 = 0b01101				

Room Configuration Data Block

This Room Configuration Data Block example shows how to define the room characteristics.

Room Configuration Data Bloc, Hex Equivalent: 0xEB13630500000F0F0A000000

Table 136. Room Configuration Data Block Example

Bits:	7	6	5	4	3	2	1	0
Byte 1	Extended Tag = 7 (0b111)			Length of following in bytes = 11				
Byte 2	Room Configuration Data Block Extended Tag Code = 19							
Byte 3	Display = 0	Speaker = 1	SLD = 1	Speaker Count = 3				
Byte 4 (SPM1)	FLW/FRW = 0	F46=0	FLC/FRC = 0	BC = 0	BL/BR = 0	FC = 1	LFE1 = 0	FL/FR = 1
Byte 5 (SPM2)	TpSiL/TpSiR = 0	SiL/SiR = 0	TpBC = 0	LFE2 = 0	LS/RS = 0	TpFC = 0	TpC = 0	TpFL/TpFR = 0
Byte 6 (SPM3)	F67=0	F66=0	F65=0	F64=0	TpLS/TpRS = 0	BtFL/BtFR = 0	BtFC = 0	TpBL/TpBR = 0
Byte 7 (MAX1)	Xmax = 15							
Byte 8 (MAX2)	Ymax = 15							
Byte 9 (MAX3)	Zmax = 10							
Byte 10 (DISP1)	DisplayX = 0							
Byte 11 (DISP2)	DisplayY = 0							
Byte 12 (DISP3)	DisplayZ = 0							

Speaker Location Data Block

This Speaker Location Data Block example shows how to define the speakers attached to the WFD R2 Sink.

Speaker Location Data Block, Hex Equivalent: 0xE714200021012202

Data Block Header

Table 137. Speaker Location Data Block Example

Bits:	7	6	5	4	3	2	1	0
Byte 1	Extended Tag = 7			Length of following in bytes = 7				
Byte 2	Speaker Location Data Block Extended Tag Code = 20							

Speaker Location Descriptor

Bits:	7	6	5	4	3	2	1	0
Byte 3	F17=0	COORD = 0	Active = 1	Channel Index = 1				
Byte 4	F27=0	F26=0	F35=0	Speaker ID = 0x00 (Front Left)				
Byte 5	F17=0	COORD = 0	Active = 1	Channel Index = 1				
Byte 6	F27=0	F26=0	F35=0	Speaker ID = 0x01 (Front Right)				

Byte 7	F17=0	COORD = 0	Active = 1	Channel Index = 2
Byte 8	F27=0	F26=0	F35=0	Speaker ID = 0x02 (Front Center)

Audio InfoFrame

This Audio InfoFrame example indicates a transmission of the CTA-861-G defined L-PCM 3D audio type, using “Delivery by Channel Index”.

Audio InfoFrame Hex Equivalent: 0x04010AF20D0DFF000700000000

Table 138. Audio InfoFrame Example

	7	6	5	4	3	2	1	0
Type	InfoFrame Type = 4							
Version	Version = 1							
Length	Length of Audio InfoFrame = 10							
DB1	Coding Type = 15 (Use Audio Coding Ext., CTX)				F13=0	Number Channels -1 = 2 (3 channels)		
DB2	F27=0	F26=0	F25=0	Sampling Frequency = 3 (48 kHz)			Sample Size = 1 (16 bit)	
DB3	F37=0	F36=0	F35=0	CXT = 13 (L-PCM 3D Audio)				
DB4	Channel Allocation = 0xFF (Delivery by Channel Index)							
DB5	Downmix = 0	Left Shift Value = 0 (0 dB)				F52=0	LFE Playback Level = 0 (unknown)	
DB6	CID07=0	CID06=0	CID05=0	CID04=0	CID03=0	CID02=1	CID01=1	CID00=1
DB7	CID15=0	CID14=0	CID13=0	CID12=0	CID11=0	CID10=0	CID09=0	CID08=0
DB8	CID23=0	CID22=0	CID21=0	CID20=0	CID19=0	CID18=0	CID17=0	CID16=0
DB9	CID31=0	CID30=0	CID29=0	CID28=0	CID27=0	CID26=0	CID25=0	CID24=0
DB10	F107=0	F106=0	F105=0	F104=0	F103=0	F102=0	F101=0	F100=0

The following is an example of a Room Configuration Data Block (RCDB) in a system that consists of a WFD R2 Source and two WFD R2 Secondary Sinks.

In this example WFD R2 Secondary Sink #1 contains 1 Channel assigned to the Left Channel (FL) and WFD R2 Secondary Sink #2 contains 1 Channel assigned to the Left Channel (FR).

The RCDB for WFD R2 Secondary Sink #1 is configured as shown in Table 139.

Table 139. Room Configuration Data Block Example 2

Bits:	7	6	5	4	3	2	1	0
Byte 1	Tag Code = 7			Length of following block payload (bytes) = 11				
Byte 2	Extended Tag Code (0x13)							
Byte 3	Display = 0	Speaker = 1	SLD = 1	Speaker Count = 1				
SPM1	FLW/FRW = 0	F46=0	FLC/FRC = 0	BC = 0	BL/BR = 0	FC = 0	LFE1 = 0	FL/FR = 0
SPM2	TpSiL/TpSiR = 0	SiL/SiR = 0	TpBC = 0	LFE2 = 0	LS/RS = 0	TpFC = 0	TpC = 0	TpFL/TpFR = 0
SPM3	F67=0	F66=0	F65=0	F64=0	TpLS/TpRS = 0	BtFL/BtFR = 0	BtFC = 0	TpBL/TpBR = 0
MAX1	Xmax = 15							

MAX2	Ymax = 15
MAX3	Zmax = 10
DISP1	DisplayX = 0
DISP2	DisplayY = 0
DISP3	DisplayZ = 0

Note that since a complete L/R pair is not represented in this WFD R2 Sink, no flags in the SPM registers are set.

The Speaker Location Data Block (SLDB) for WFD R2 Secondary Sink #1 is configured as shown in Table 140.

Table 140. Speaker Location Data Block Example 2

Bits:	7	6	5	4	3	2	1	0
Byte 1	Tag Code = 7			Length of following block payload (bytes) = 3				
Byte 2	Extended Tag Code (0x14)							
Channel	F17=0	COORD = 0	Active = 1	Channel Index (0 to 31) = 0				
ID	F27=0	F26=0	F35=0	Speaker ID (0 to 31) (from Table 1 in CTA-861-F.2) = 0x00				

Note the following:

- No flags are set in the SPM block because a complete pair is not present.
- The values in MAX field are arbitrary since spatial coordinates are not defined in the SLD that follows.
- Speaker ID = 0 indicating Left Front channel

The RCDB for WFD R2 Sink #2 is identical to that of WFD R2 Sink #1. The SLDB for WFD R2 Secondary Sink #2 is configured as shown in Table 141.

Table 141. Speaker Location Data Block Example 3

Bits:	7	6	5	4	3	2	1	0
Byte 1	Tag Code = 7			Length of following block payload (bytes) = 3				
Byte 2	Extended Tag Code (0x14)							
Channel	F17=0	COORD = 0	Active = 1	Channel Index (0 to 31) = 0				
ID	F27=0	F26=0	F35=0	Speaker ID (0 to 31) (from Table 1 in CTA-861-F.2) = 0x01				

The only difference for WFD R2 Sink #2 is the Speaker ID = 0x01, indicating Right Front channel.

If the SLDB includes coordinates (X,Y,Z) of the speaker, these are interpreted relative to the Xmax, Ymax & Zmax in the RCDB.

Note that the value of X,Y & Z range from -2 to 1.984375. This value is multiplied by the corresponding Xmax, Ymax & Zmax in the RCDB and divided by 10 to give absolute position in meters relative to the Primary Listening Position (PLP).

For example, if X = -1 and Xmax = 15, then the position of X in meters is (-1 * 15) / 10 = -1.5m (this is 1.5 meters left of the PLP).