INTERNATIONAL STANDARD

ISO/IEC 29341-1-2

First edition
2017-06

Information technology — UPnP Device Architecture —

Part 1-2: UPnP Device Architecture Version 2.0

Technologies de l'information — Architecture de dispositif UPnP — Partie 1-2: Architecture de dispositif UPnP, version 2.0

Reference number
ISO/IEC 29341-1-2:2017(E)

© ISO/IEC 2017



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2017, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

CONTENTS

Introduction 1
0 Addressing 7
0.1 Determining whether to use Auto-IP 7
0.2 Choosing an address 7
0.3 Testing the address 8
0.4 Forwarding rules 8
0.5 Periodic checking for dynamic address availability 9
0.6 Device naming and DNS interaction 9
0.7 Name to IP address resolution 9
0.8 References 9
1 Discovery 10
1.1 SSDP message format 13
1.1.1 SSDP Start-line 13
1.1.2 SSDP message header fields 13
1.1.3 SSDP header field extensions 14
1.1.4 UUID format and recommended generation algorithms 14
1.1.5 SSDP processing rules 14
1.2 Advertisement 15
1.2.1 Advertisement protocols and standards 15
1.2.2 Device available - NOTIFY with ssdp:alive 16
1.2.3 Device unavailable -- NOTIFY with ssdp:byebye 21
1.2.4 Device Update – NOTIFY with ssdp:update 23
1.3 Search 25
1.3.1 Search protocols and standards 25
1.3.2 Search request with M-SEARCH 26
1.3.3 Search response 29
1.4 References 32
2 Description 32
2.1 Generic requirements on HTTP usage 35
2.2 Generic requirements on XML usage 38
2.3 Device description 38
2.4 UPnP Device Template 43
2.5 Service description 44
2.5.1 Defining and processing extended data types 51
2.5.2 String equivalents of extended data types 52
2.5.3 Generic requirements 53
2.5.4 Ordering of Elements 53
2.5.5 Versioning 54
2.6 UPnP Service Template 54
2.7 Non-standard vendor extensions and limitations 54
2.7.1 Placement of Additional Elements and Attributes 56
2.8 UPnP Device Schema 56
2.9 UPnP Service Schema 56
2.10 UPnP Datatype Schema 56
2.11 Retrieving a description using HTTP 57

© ISO/IEC 2017 – All rights reserved

2.12 References 59
3 Control 60
3.1 Control protocols 63
3.1.1 SOAP Profile 63
3.2 Actions 67
3.2.1 Action invocation 67
3.2.2 Action Response 70
3.2.3 UPnP Action Schema 72
3.2.4 Recommendations and additional requirements 72
3.2.5 Action error response 73
3.2.6 UPnP Error Schema 76
3.3 Query for variable 76
3.4 References 77
4 Eventing 77
4.1 Unicast eventing 78
4.1.1 Subscription 79
4.1.2 SUBSCRIBE with NT and CALLBACK 81
4.1.3 Renewing a subscription with SUBSCRIBE with SID 84
4.1.4 Canceling a subscription with UNSUBSCRIBE 85
4.2 Multicast Eventing 86
4.3 Event messages 88
4.3.1 Error Cases 88
4.3.2 Unicast eventing: Event messages: NOTIFY 89
4.3.3 Multicast Eventing: Event messages: NOTIFY 92
4.4 UPnP Event Schema 95
4.5 Augmenting the UPnP Device and Service Schemas 95
4.6 References 95
5 Presentation 96
5.1 References 97
Annex A (normative) IP Version 6 Support 98
A.0 Note (informative) 98
A.1 Introduction 98
A.2 General Principles 98
A.2.1 UPnP Device Architecture V1.0 98
A.2.2 UPnP Device Architecture V2.0 99
A.2.3 IPv6 and Dual Stack 99
A.2.4 Device operation 100
A.2.5 Control point operation 101
A.3 Addressing 101
A.3.1 UPnP Messaging on IPv6 Interfaces 101
A.3.2 Summary of boot/startup process 102
A.3.3 Address Selection and RFC 6724 102
A.4 Discovery 102
A.4.1 OPT and NLS 102
A.4.2 Advertisement 103
A.4.3 Advertisement: Device unavailable 103
A.4.4 Advertisement: Device update 104
A.4.5 Search 104

 © ISO/IEC 2017 – All rights reserved

A.4.6 Search response 104
A.5 Description 104
A.6 Control 104
A.7 Eventing 105
A.8 Presentation 105
A.9 References 105
A.9.1 Normative 105
A.9.2 Informative 106
Annex B Schemas 107
B.1 UPnP Device Schema 107
B.2 UPnP Service Schema 111
B.3 UPnP Control Schema 115
B.4 UPnP Error Schema 116
B.5 UPnP Event Schema 117
B.6 UPnP Cloud Schema 118
B.7 Schema references 119
Annex C Cloud 120
C.1 Introduction 120
C.1.1 What is UPnP™ Cloud Technology (UCA)? 120
C.1.2 Audience 120
C.1.3 In this Annex 120
C.1.4 UDA compared to UCA 122
C.1.5 UCA General Communications Paths 124
C.1.6 UCA Specific Communication Paths 125
C.1.7 UCA Steps as Analogies to UDA 126
C.2 Terms and Definitions 127
C.2.1 Acronyms 127
C.2.2 General Cloud Terms and Definitions 128
C.2.3 Device and Control Point Terms and Definitions 128
C.2.4 Service Terms and Definitions 129
C.2.5 Groups 129
C.3 References 129
C.4 General XMPP Features 130
C.4.1 XMPP Jabber IDs or JIDs 130
C.5 Creating a Device or Control Point Resource 132
C.5.1 Finding a UCS 132
C.5.2 Account Creation 132
C.5.3 Authentication 133
C.5.4 Binding Devices and Control Points as a Resource 135
C.5.5 Embedded Devices 138
C.6 Presence and Discovery 140
C.6.1 Presence (Analog to NOTIFY with ssdp:alive) 140
C.6.2 XMPP disco#items (analog to M-SEARCH for users UCCDs and UCC-CPs) 144
C.6.3 Presence update (analog to NOTIFY with ssdp:update) 145
C.6.4 Presence "unavailable" (Analog to NOTIFY with ssdp:byebye) 145
C.6.5 Service Level Discovery 146
C.6.6 IQ:Query for DDD and SCPD Exchange (analog of HTTP GET for DDD and SCPD) 146

© ISO/IEC 2017 – All rights reserved

C.7 PubSub (Analog of Eventing) 155
C.7.1 Creating the UCCD PubSub structure 159
C.7.2 Creating a UCCD PubSub collection 161
C.7.3 Publishing a UCCD PubSub event 166
C.7.4 Subscribing to a UCCD PubSub collection 169
C.7.5 Unsubscribing to a UCCD PubSub collection 171
C.7.6 Permissions model 173
C.8 SOAP over XMPP (Analog of Control) 173
C.9 Support for Binary (Media) Transport 177
C.10 UCA errorCodes 177
C.11 UCA Schemas 178
C.12 Closing a UCA Session 178
C.13 UCA over BOSH and WebSocket 178

Figure 1: — Protocol stack 1
Figure 1-1: — Discovery architecture 11
Figure 1-2: — Advertisement protocol stack 15
Figure 1-3: — Initial and repeat announcements, no announcement spreading 17
Figure 1-4: — Initial and repeat announcements, message spreading of repeat announcements 18
Figure 1-5: — Search protocol stack 25
Figure 2-1: — Description architecture 33
Figure 2-2: — Description retrieval protocol stack 57
Figure 3-1: — Control architecture 61
Figure 3-2: — Control protocol stack 63
Figure 4-1: — Unicast eventing architecture 78
Figure 4-2: — Unicast eventing protocol stack 79
Figure 4-3: — Multicast eventing architecture 86
Figure 4-4: — Mulitcast eventing protocol stack 87
Figure 5-1: — Presentation architecture 96
Figure 5-2: — Presentation protocol stack 96
Figure C-1: — Protocol stacks UDA versus UCA 122
Figure C-2: — Protocol stack UCA UCCD/UCC-CP and UCA Servers (UCS or UCOD) 123
Figure C-3: — General UCA Configuration 124
Figure C-4: — Specific UCA communications 125
Figure C-5: — XMPP Authentication Negotiation 133
Figure C-6: — Stanza routing for applications with UCA and other XMPP functionality 138
Figure C-7: — UDA to UCA Mapping of embedded devices 140
The individual presence exchange between the UCCDs, UCC-CPs, and UCS for an N connected UPnP scenario is illustrated in 143
Figure C-8: — Self <presence> stanza flows 144
Figure C-9: — Combined Connect, Announce and Describe Message Flow 153
Figure C-10: — PubSub Hierarchy Event Structure Creation 158
Figure C-11: — BOSH and WebSocket UCA Stack 178
Figure C-12: — BOSH and WebSocket at UCA component stacks 180

 © ISO/IEC 2017 – All rights reserved

Table 1 — Acronyms 4
Table 1-1 — Root device discovery messages 16
Table 1-2 — Embedded device discovery messages 16
Table 1-3 — Service discovery messages 16
Table 2-1: — Vendor extensions 54
Table 3-1: — SOAP 1.1 UPnP Profile 64
Table 3-2: — mustUnderstand attribute 65
Table 3-3: — UPnP Defined Action error codes 75
Table 4-4: — HTTP Status Codes indicating a Subscription Error 83
Table 4-5: — HTTP Status Codes indicating a Resubscription Error 85
Table 4-6: — HTTP Status Codes indicating a Cancel Subscription Error 86
Table 4-7: — HTTP Status Codes indicating a Notify Error 92
Table 4-8: — Multicast event levels 94
Table A-1: — Matching of Device Address to Multicast Scope 100
Table C-1: — Acronyms 127
Table C-2: — Mapping of DDD `iconList` to [XEP-0084] 150
Table C-3: — Summary of Requirements for DDD elements 154
Table C-4: — `PubSub` `Node` Types 155
Table C-5: — `PubSub` Node Access Models 155
Table C-6: — PubSub Affiliations and their Privileges to "publishing" as defined by [XEP-0060] and further restricted by UCA (see footnotes) 156
Table C-7: — `PubSub` Affiliations and their Privileges to "subscribers" 157

© ISO/IEC 2017 – All rights reserved

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see http://www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of Standards, the meaning of the ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword – Supplementary information .

ISO/IEC 29341-1-2 was prepared by UPnP Forum and adopted, under the PAS procedure, by joint technical committee ISO/IEC JTC 1, *Information technology*, in parallel with its approval by national bodies of ISO and IEC.

The list of all currently available parts of ISO/IEC 29341 series, under the general title *Information technology — UPnP Device Architecture*, can be found on the ISO web site.

© ISO/IEC 2017 – All rights reserved

Introduction

ISO and IEC draw attention to the fact that it is claimed that compliance with this document may involve the use of patents as indicated below.

ISO and IEC take no position concerning the evidence, validity and scope of these patent rights. The holders of -these patent rights have assured ISO and IEC that they are willing to negotiate licenses under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statements of the holders of these patent rights are registered with ISO and IEC.

Intel Corporation has informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

Intel Corporation
Standards Licensing Department
5200 NE Elam Young Parkway
MS: JFS-98
USA – Hillsboro, Oregon 97124

Microsoft Corporation has informed IEC and ISO that it has patent applications or granted patents as listed below:

6101499 / US; 6687755 / US; 6910068 / US; 7130895 / US; 6725281 / US; 7089307 / US; 7069312 / US; 10/783 524 /US

Information may be obtained from:

Microsoft Corporation
One Microsoft Way
USA – Redmond WA 98052

Philips International B.V. has informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

Philips International B.V. – IP&S
High Tech campus, building 44 3A21
NL – 5656 Eindhoven

NXP B.V. (NL) has informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

NXP B.V. (NL)
High Tech campus 60
NL – 5656 AG Eindhoven

Matsushita Electric Industrial Co. Ltd. has informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

Matsushita Electric Industrial Co. Ltd.
1-3-7 Shiromi, Chuoh-ku
JP – Osaka 540-6139

Hewlett Packard Company has informed IEC and ISO that it has patent applications or granted patents as listed below:

5 956 487 / US; 6 170 007 / US; 6 139 177 / US; 6 529 936 / US; 6 470 339 / US; 6 571 388 / US; 6 205 466 / US

© ISO/IEC 2017 – All rights reserved

Information may be obtained from:

Hewlett Packard Company
1501 Page Mill Road
USA – Palo Alto, CA 94304

Samsung Electronics Co. Ltd. has informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

Digital Media Business, Samsung Electronics Co. Ltd.
416 Maetan-3 Dong, Yeongtang-Gu,
KR – Suwon City 443-742

Huawei Technologies Co., Ltd. has informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

Huawei Technologies Co., Ltd.
Administration Building, Bantian Longgang District
Shenzhen – China 518129

Qualcomm Incorporated has informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

Qualcomm Incorporated
5775 Morehouse Drive
San Diego, CA – USA 92121

Telecom Italia S.p.A.has informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

Telecom Italia S.p.A.
Via Reiss Romoli, 274
Turin - Italy 10148

Cisco Systems informed IEC and ISO that it has patent applications or granted patents.

Information may be obtained from:

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA – USA 95134

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

 © ISO/IEC 2017 – All rights reserved

Original UPnP Document

Reference may be made in this document to original UPnP documents. These references are retained in order to maintain consistency between the specifications as published by ISO/IEC and by UPnP Implementers Corporation and later by UPnP Forum. The following table indicates the original UPnP document titles and the corresponding part of ISO/IEC 29341:

UPnP Document Title	ISO/IEC 29341 Part
UPnP Device Architecture 1.0	ISO/IEC 29341-1:2008
UPnP Device Architecture Version 1.0	ISO/IEC 29341-1:2011
UPnP Device Architecture 1.1	ISO/IEC 29341-1-1:2011
UPnP Device Architecture 2.0	ISO/IEC 29341-1-2
UPnP Basic:1 Device	ISO/IEC 29341-2
UPnP AV Architecture:1	ISO/IEC 29341-3-1:2008
UPnP AV Architecture:1	ISO/IEC 29341-3-1:2011
UPnP AVTransport:1 Service	ISO/IEC 29341-3-10
UPnP ConnectionManager:1 Service	ISO/IEC 29341-3-11
UPnP ContentDirectory:1 Service	ISO/IEC 29341-3-12
UPnP RenderingControl:1 Service	ISO/IEC 29341-3-13
UPnP MediaRenderer:1 Device	ISO/IEC 29341-3-2
UPnP MediaRenderer:2 Device	ISO/IEC 29341-3-2:2011
UPnP MediaServer:1 Device	ISO/IEC 29341-3-3
UPnP AVTransport:2 Service	ISO/IEC 29341-4-10:2008
UPnP AVTransport:2 Service	ISO/IEC 29341-4-10:2011
UPnP ConnectionManager:2 Service	ISO/IEC 29341-4-11:2008
UPnP ConnectionManager:2 Service	ISO/IEC 29341-4-11:2011
UPnP ContentDirectory:2 Service	ISO/IEC 29341-4-12
UPnP RenderingControl:2 Service	ISO/IEC 29341-4-13:2008
UPnP RenderingControl:2 Service	ISO/IEC 29341-4-13:2011
UPnP ScheduledRecording:1	ISO/IEC 29341-4-14
UPnP ScheduledRecording:2	ISO/IEC 29341-4-14:2011
UPnP MediaRenderer:2 Device	ISO/IEC 29341-4-2
UPnP MediaServer:2 Device	ISO/IEC 29341-4-3
UPnP AV Datastructure Template:1	ISO/IEC 29341-4-4:2008
UPnP AV Datastructure Template:1	ISO/IEC 29341-4-4:2011
UPnP DigitalSecurityCamera:1 Device	ISO/IEC 29341-5-1
UPnP DigitalSecurityCameraMotionImage:1 Service	ISO/IEC 29341-5-10
UPnP DigitalSecurityCameraSettings:1 Service	ISO/IEC 29341-5-11
UPnP DigitalSecurityCameraStillImage:1 Service	ISO/IEC 29341-5-12
UPnP HVAC_System:1 Device	ISO/IEC 29341-6-1
UPnP ControlValve:1 Service	ISO/IEC 29341-6-10
UPnP HVAC_FanOperatingMode:1 Service	ISO/IEC 29341-6-11
UPnP FanSpeed:1 Service	ISO/IEC 29341-6-12
UPnP HouseStatus:1 Service	ISO/IEC 29341-6-13
UPnP HVAC_SetpointSchedule:1 Service	ISO/IEC 29341-6-14
UPnP TemperatureSensor:1 Service	ISO/IEC 29341-6-15
UPnP TemperatureSetpoint:1 Service	ISO/IEC 29341-6-16
UPnP HVAC_UserOperatingMode:1 Service	ISO/IEC 29341-6-17
UPnP HVAC_ZoneThermostat:1 Device	ISO/IEC 29341-6-2

© ISO/IEC 2017 – All rights reserved

UPnP BinaryLight:1 Device	ISO/IEC 29341-7-1
UPnP Dimming:1 Service	ISO/IEC 29341-7-10
UPnP SwitchPower:1 Service	ISO/IEC 29341-7-11
UPnP DimmableLight:1 Device	ISO/IEC 29341-7-2
UPnP InternetGatewayDevice:1 Device	ISO/IEC 29341-8-1
UPnP LANHostConfigManagement:1 Service	ISO/IEC 29341-8-10
UPnP Layer3Forwarding:1 Service	ISO/IEC 29341-8-11
UPnP LinkAuthentication:1 Service	ISO/IEC 29341-8-12
UPnP RadiusClient:1 Service	ISO/IEC 29341-8-13
UPnP WANCableLinkConfig:1 Service	ISO/IEC 29341-8-14
UPnP WANCommonInterfaceConfig:1 Service	ISO/IEC 29341-8-15
UPnP WANDSLLinkConfig:1 Service	ISO/IEC 29341-8-16
UPnP WANEthernetLinkConfig:1 Service	ISO/IEC 29341-8-17
UPnP WANIPConnection:1 Service	ISO/IEC 29341-8-18
UPnP WANPOTSLinkConfig:1 Service	ISO/IEC 29341-8-19
UPnP LANDevice:1 Device	ISO/IEC 29341-8-2
UPnP WANPPPConnection:1 Service	ISO/IEC 29341-8-20
UPnP WLANConfiguration:1 Service	ISO/IEC 29341-8-21
UPnP WANDevice:1 Device	ISO/IEC 29341-8-3
UPnP WANConnectionDevice:1 Device	ISO/IEC 29341-8-4
UPnP WLANAccessPointDevice:1 Device	ISO/IEC 29341-8-5
UPnP Printer:1 Device	ISO/IEC 29341-9-1
UPnP ExternalActivity:1 Service	ISO/IEC 29341-9-10
UPnP Feeder:1.0 Service	ISO/IEC 29341-9-11
UPnP PrintBasic:1 Service	ISO/IEC 29341-9-12
UPnP Scan:1 Service	ISO/IEC 29341-9-13
UPnP Scanner:1.0 Device	ISO/IEC 29341-9-2
UPnP QoS Architecture:1.0	ISO/IEC 29341-10-1
UPnP QosDevice:1 Service	ISO/IEC 29341-10-10
UPnP QosManager:1 Service	ISO/IEC 29341-10-11
UPnP QosPolicyHolder:1 Service	ISO/IEC 29341-10-12
UPnP QoS Architecture:2	ISO/IEC 29341-11-1
UPnP QosDevice:2 Service	ISO/IEC 29341-11-10
UPnP QosManager:2 Service	ISO/IEC 29341-11-11
UPnP QosPolicyHolder:2 Service	ISO/IEC 29341-11-12
UPnP QOS v2 Schema Files	ISO/IEC 29341-11-2
UPnP RemoteUIClientDevice:1 Device	ISO/IEC 29341-12-1
UPnP RemoteUIClient:1 Service	ISO/IEC 29341-12-10
UPnP RemoteUIServer:1 Service	ISO/IEC 29341-12-11
UPnP RemoteUIServerDevice:1 Device	ISO/IEC 29341-12-2
UPnP DeviceSecurity:1 Service	ISO/IEC 29341-13-10
UPnP SecurityConsole:1 Service	ISO/IEC 29341-13-11
UPnP ContentDirectory:3 Service	ISO/IEC 29341-14-12:2011
UPnP MediaServer:3 Device	ISO/IEC 29341-14-3:2011
UPnP ContentSync:1	ISO/IEC 29341-15-10:2011
UPnP Low Power Architecture:1	ISO/IEC 29341-16-1:2011
UPnP LowPowerProxy:1 Service	ISO/IEC 29341-16-10:2011

© ISO/IEC 2017 – All rights reserved

UPnP LowPowerDevice:1 Service	ISO/IEC 29341-16-11:2011
UPnP QoS Architecture:3	ISO/IEC 29341-17-1:2011
UPnP QosDevice:3 Service	ISO/IEC 29341-17-10:2011
UPnP QosManager:3 Service	ISO/IEC 29341-17-11:2011
UPnP QosPolicyHolder:3 Service	ISO/IEC 29341-17-12:2011
UPnP QosDevice:3 Addendum	ISO/IEC 29341-17-13:2011
UPnP RemoteAccessArchitecture:1	ISO/IEC 29341-18-1:2011
UPnP InboundConnectionConfig:1 Service	ISO/IEC 29341-18-10:2011
UPnP RADAConfig:1 Service	ISO/IEC 29341-18-11:2011
UPnP RADASync:1 Service	ISO/IEC 29341-18-12:2011
UPnP RATAConfig:1 Service	ISO/IEC 29341-18-13:2011
UPnP RAClient:1 Device	ISO/IEC 29341-18-2:2011
UPnP RAServer:1 Device	ISO/IEC 29341-18-3:2011
UPnP RADiscoveryAgent:1 Device	ISO/IEC 29341-18-4:2011
UPnP SolarProtectionBlind:1 Device	ISO/IEC 29341-19-1:2011
UPnP TwoWayMotionMotor:1 Service	ISO/IEC 29341-19-10:2011
UPnP AV Architecture:2	ISO/IEC 29341-20-1
UPnP AVTransport:3 Service	ISO/IEC 29341-20-10
UPnP ConnectionManager:3 Service	ISO/IEC 29341-20-11
UPnP ContentDirectory:4 Device	ISO/IEC 29341-20-12
UPnP RenderingControl:3 Service	ISO/IEC 29341-20-13
UPnP ScheduledRecording:2 Service	ISO/IEC 29341-20-14
UPnP MediaRenderer:3 Service	ISO/IEC 29341-20-2
UPnP MediaServer:4 Device	ISO/IEC 29341-20-3
UPnP AV Datastructure Template:1	ISO/IEC 29341-20-4
UPnP InternetGatewayDevice:2 Device	ISO/IEC 29341-24-1
UPnP WANIPConnection:2 Service	ISO/IEC 29341-24-10
UPnP WANIPv6FirewallControl:1 Service	ISO/IEC 29341-24-11
UPnP WANConnectionDevice:2 Service	ISO/IEC 29341-24-2
UPnP WANDevice:2 Device	ISO/IEC 29341-24-3
UPnP Telephony Architecture:2	ISO/IEC 29341-26-1
UPnP CallManagement:2 Service	ISO/IEC 29341-26-10
UPnP MediaManagement:2 Service	ISO/IEC 29341-26-11
UPnP Messaging:2 Service	ISO/IEC 29341-26-12
UPnP PhoneManagement:2 Service	ISO/IEC 29341-26-13
UPnP AddressBook:1 Service	ISO/IEC 29341-26-14
UPnP Calendar:1 Service	ISO/IEC 29341-26-15
UPnP Presense:1 Service	ISO/IEC 29341-26-16
UPnP TelephonyClient:2 Device	ISO/IEC 29341-26-2
UPnP TelephonyServer:2 Device	ISO/IEC 29341-26-3
UPnP Friendly Info Update:1 Service	ISO/IEC 29341-27-1
UPnP MultiScreen MultiScreen Architecture:1	ISO/IEC 29341-28-1
UPnP MultiScreen Application Management:1 Service	ISO/IEC 29341-28-10
UPnP MultiScreen Screen:1 Device	ISO/IEC 29341-28-2
UPnP MultiScreen Application Management:2 Service	ISO/IEC 29341-29-10
UPnP MultiScreen Screen:2 Device	ISO/IEC 29341-29-2
UPnP IoT Management and Control Architecture Overview:1	ISO/IEC 29341-30-1

© ISO/IEC 2017 – All rights reserved

UPnP DataStore:1 Service	ISO/IEC 29341-30-10
UPnP IoT Management and Control Data Model:1 Service	ISO/IEC 29341-30-11
UPnP IoT Management and Control Transport Generic:1 Service	ISO/IEC 29341-30-12
UPnP IoT Management and Control:1 Device	ISO/IEC 29341-30-2
UPnP Energy Management:1 Service	ISO/IEC 29341-31-1

© ISO/IEC 2017 – All rights reserved

Introduction

What is UPnP Technology?

UPnP technology defines an architecture for pervasive peer-to-peer network connectivity of intelligent appliances, wireless devices, and PCs of all form factors. It is designed to bring easy-to-use, flexible, standards-based connectivity to ad-hoc or unmanaged networks whether in the home, in a small business, public spaces, or attached to the Internet. UPnP technology provides a distributed, open networking architecture that leverages TCP/IP and Web technologies to enable seamless proximity networking in addition to control and data transfer among networked devices.

The UPnP Device Architecture (UDA) is more than just a simple extension of the plug and play peripheral model. It is designed to support zero-configuration, "invisible" networking, and automatic discovery for a breadth of device categories from a wide range of vendors. This means a device can dynamically join a network, obtain an IP address, convey its capabilities, and learn about the presence and capabilities of other devices. Finally, a device can leave a network smoothly and automatically without leaving any unwanted state behind.

The technologies leveraged in the UPnP architecture include Internet protocols such as IP, TCP, UDP, HTTP, and XML. Like the Internet, contracts are based on wire protocols that are declarative, expressed in XML, and communicated via HTTP. Using Internet protocols is a strong choice for UDA because of its proven ability to span different physical media, to enable real world multiple-vendor interoperation, and to achieve synergy with the Internet and many home and office intranets. The UPnP architecture has been explicitly designed to accommodate these environments. Further, via bridging, UDA accommodates media running non-IP protocols when cost, technology, or legacy prevents the media or devices attached to it from running IP.

What is "universal" about UPnP technology? No device drivers; common protocols are used instead. UPnP networking is media independent. UPnP devices can be implemented using any programming language, and on any operating system. The UPnP architecture does not specify or constrain the design of an API for applications; OS vendors may create APIs that suit their customers' needs.

UPnP Forum

UPnP Forum is an industry initiative designed to enable easy and robust connectivity among stand-alone devices and PCs from many different vendors. UPnP Forum seeks to develop standards for describing device protocols and XML-based device schemas for the purpose of enabling device-to-device interoperability in a scalable, networked environment. For more information, see http://www.upnp.org.

UPnP Forum has set up working committees in specific areas of domain expertise. These working committees are charged with creating proposed device standards, building sample implementations, and building appropriate test suites. This document indicates specific technical decisions that are the purview of UPnP Forum working committees.

UPnP vendors can build compliant devices with confidence of interoperability and benefits of shared intellectual property and the logo program. Separate from the logo program, vendors may also build devices that adhere to the UPnP Device Architecture defined herein without a formal standards procedure. If vendors build non-standard devices, they determine technical decisions that would otherwise be determined by a UPnP Forum working committee.

In this document

The UPnP Device Architecture (formerly known as the DCP Framework) contained herein defines the protocols for communication between controllers, or *control points*, and devices. For discovery, description, control, eventing, and presentation, the UPnP Device Architecture uses the following protocol stack (the indicated colors and type styles are used throughout this document to indicate where each protocol element is defined):

Figure 1: — Protocol stack

© ISO/IEC 2017 – All rights reserved

<table>
<tr><td colspan="4">UPnP vendor [purple-italic]</td></tr>
<tr><td colspan="4">UPnP Forum [red-italic]</td></tr>
<tr><td colspan="4">UPnP Device Architecture [green-bold]</td></tr>
<tr><td rowspan="2">SSDP [blue]</td><td rowspan="2">Multicast events [navy-bold]</td><td>SOAP [blue]</td><td>GENA [navy-bold]</td></tr>
<tr><td>HTTP [black]</td><td>HTTP [black]</td></tr>
<tr><td colspan="2">UDP [black]</td><td colspan="2">TCP [black]</td></tr>
<tr><td colspan="4">IP [black]</td></tr>
</table>

At the highest layer, messages logically contain only UPnP vendor-specific information about their devices. Moving down the stack, vendor content is supplemented by information defined by UPnP Forum working committees. Messages from the layers above are hosted in UPnP-specific protocols such as the Simple Service Discovery Protocol (SSDP), the General Event Notification Architecture (GENA) and the multicast event protocol defined in this document, and others that are referenced. SSDP is delivered via either multicast or unicast UDP. Multicast events are delivered via multicast UDP. GENA is delivered via HTTP. Ultimately, all messages above are delivered over IP. The remaining clauses of this document describe the content and format for each of these protocol layers in detail. For reference, colors in [square brackets] above indicate which protocol defines specific message components throughout this document.

Two general classifications of devices are defined by the UPnP architecture: controlled devices (or simply “devices”), and control points. A controlled device functions in the role of a server, responding to requests from control points. Both control points and controlled devices can be implemented on a variety of platforms including personal computers and embedded systems. Multiple devices, control points, or both may be operational on the same network endpoint simultaneously.

Note: This document is oriented toward an IPv4 environment. Considerations for an IPv6 environment are expressed in Annex A.

The foundation for UPnP networking is IP addressing. In an IPv4 environment, each device or control point shall have a Dynamic Host Configuration Protocol (DHCP) client and search for a DHCP server when the device or control point is first connected to the network. If a DHCP server is available, i.e., the network is managed; the device or control point shall use the IP address assigned to it. If no DHCP server is available, i.e., the network is unmanaged; the device or control point shall use Auto IP to get an address. In brief, Auto IP defines how a device or control point intelligently chooses an IP address from a set of reserved addresses and is able to move easily between managed and unmanaged networks. If during the DHCP transaction, the device or control point obtains a domain name, e.g., through a DNS server or via DNS forwarding, the device or control point should use that name in subsequent network operations; otherwise, the device or control point should use its IP address.

Certain UPnP networks have more complex configurations such as multiple physical networks and/or multiple logical networks to accommodate multiple non-overlapping addressing schemes. Devices and control points may also have two or more network interfaces, and/or two or more IP addresses assigned to each interface. In such configurations, a single device or control point may be assigned multiple IP addresses from different logical networks in the same UPnP network, resulting in devices appearing to a control point multiple times in the network. Devices and control points that have multiple IP addresses on the same UPnP network are referred to as multi-homed. Throughout this document, the term "UPnP-enabled interface" is used to refer to an interface which is assigned an IP address belonging to the UPnP network. Additional behaviors specific to multi-homed devices and control points will be covered in applicable clauses throughout the document. However, as a general principle, related interactions between control points and devices (e.g. action control request and response messages, event subscription and event messages) shall occur using the same pair of outgoing and incoming UPnP-enabled interfaces.

 © ISO/IEC 2017 – All rights reserved

Given an IP address, Step 1 in UPnP networking is *discovery*. When a device is added to the network, the UPnP discovery protocol allows that device to advertise its services to control points on the network. Similarly, when a control point is added to the network, the UPnP discovery protocol allows that control point to search for devices of interest on the network. The fundamental exchange in both cases is a discovery message containing a few essential specifics about the device or one of its services, e.g., its type, identifier, and a pointer to more detailed information. The clause on Discovery below explains how devices advertise, how control points search, and contains details about the format of discovery messages.

Step 2 in UPnP networking is *description*. After a control point has discovered a device, the control point still knows very little about the device. For the control point to learn more about the device and its capabilities, or to interact with the device, the control point shall retrieve the device's description from the URL provided by the device in the discovery message. Devices may contain other logical devices, as well as functional units, or *services*. The UPnP description for a device is expressed in XML and includes vendor-specific manufacturer information like the model name and number, the serial number, the manufacturer name, URLs to vendor-specific Web sites, etc. The description also includes a list of any embedded devices or services, as well as URLs for control, eventing, and presentation. For each service, the description includes a list of the commands, or *actions*, to which the service responds, and parameters, or *arguments* for each action; the description for a service also includes a list of variables; these variables model the state of the service at run time, and are described in terms of their data type, range, and event characteristics. The clause on Description below explains how devices are described and how control points retrieve those descriptions.

Step 3 in UPnP networking is *control*. After a control point has retrieved a description of the device, the control point can send actions to a device's services. To do this, a control point sends a suitable control message to the control URL for the service (provided in the device description). Control messages are also expressed in XML using the Simple Object Access Protocol (SOAP). Like function calls, in response to the control message, the service returns any action-specific values. The effects of the action, if any, are modeled by changes in the variables that describe the run-time state of the service. The clause on Control below explains the description of actions, state variables, and the format of control messages.

Step 4 in UPnP networking is *eventing*. A UPnP description for a service includes a list of actions the service responds to and a list of variables that model the state of the service at run time. The service publishes updates when these variables change, and a control point may subscribe to receive this information. The service publishes updates by sending event messages. Event messages contain the names of one or more state variables and the current value of those variables. These messages are also expressed in XML. A special initial event message is sent when a control point first subscribes; this event message contains the names and values for all evented variables and allows the subscriber to initialize its model of the state of the service. To support scenarios with multiple control points, eventing is designed to keep all control points equally informed about the effects of any action. Therefore, all subscribers are sent all event messages, subscribers receive event messages for all evented variables that have changed, and event messages are sent no matter why the state variable changed (either in response to a requested action or because the state the service is modeling changed). Multicast eventing is a variant of Step 4 in UPnP networking. Through multicast eventing, control points can listen to state changes in services without subscription. This form of eventing is useful first when events which are not relevant to specific UPnP interactions should be delivered to control points to inform users, and second when multiple controlled devices want to inform multiple other control points. Multicast eventing is inherently unreliable since it is based on UDP. To increase the probability of successful transmission, the option to retransmit multicast event notifications is outlined. UPnP Working committees should define whether specific events are multicast events. The clause on Eventing below explains unicast event subscription and the format of both unicast and multicast event messages.

Step 5 in UPnP networking is *presentation*. If a device has a URL for presentation, then the control point can retrieve a page from this URL, load the page into a browser, and depending on the capabilities of the page, allow a user to control the device and/or view device status. The degree to which each of these can be accomplished depends on the specific capabilities

© ISO/IEC 2017 – All rights reserved

of the presentation page and device. The clause on Presentation below explains the protocol for retrieving a presentation page.

Audience

The audience for this document includes UPnP device and control point vendors, members of UPnP Forum working committees, and anyone else who has a need to understanding the technical details of UPnP protocols.

This document assumes the reader is familiar with the HTTP, TCP, UDP, IP family of protocols; this document makes no attempt to explain them. This document also assumes most readers will be new to XML, and while it is not an XML tutorial, XML-related issues are addressed in detail given the centrality of XML to the UPnP Device Architecture. This document makes no assumptions about the reader's understanding of various programming or scripting languages.

Conformance terminology

In this document, features are described as required, recommended, allowed or DEPRECATED as follows:

Required (or shall or mandatory).

> These basic features shall be implemented to comply with UPnP Device Architecture. The phrases “shall not”, and “PROHIBITED” indicate behavior that is prohibited, i.e. that if performed means the implementation is not in compliance.

Recommended (or should).

> These features add functionality supported by UPnP Device Architecture and should be implemented. Recommended features take advantage of the capabilities UPnP Device Architecture, usually without imposing major cost increases. Notice that for compliance testing, if a recommended feature is implemented, it shall meet the specified requirements to be in compliance with these guidelines. Some recommended features could become requirements in the future. The phrase “should not” indicates behavior that is permitted but not recommended.

AllowedAllowed).

> These features are neither required nor recommended by UPnP Device Architecture, but if the feature is implemented, it shall meet the specified requirements to be in compliance with these guidelines. These features are not likely to become requirements in the future.

DEPRECATED.

> Although these features are still described in this specification, they should not be implemented except for backward compatibility. The occurrence of a deprecated feature during operation of an implementation compliant with the current specification has no effect on the implementation’s operation and does not produce any error conditions. Backward compatibility may require that a feature is implemented and functions as specified but it shall never be used by implementations compliant with this specification.

Acronyms

Table 1 — Acronyms

Acronym	Meaning	Acronym	Meaning
ARP	Address Resolution Protocol	SOAP	Simple Object Access Protocol
CP	Control Point	SSDP	Simple Service Discovery Protocol
DCP	Device Control Protocol	UDA	UPnP Device Architecture
DDD	Device Description Document	UPC	Universal Product Code
DHCP	Dynamic Host Configuration Protocol	URI	Uniform Resource Identifier

 © ISO/IEC 2017 – All rights reserved

DNS	Domain Name System	URL	Uniform Resource Locator
GENA	General Event Notification Architecture	URN	Uniform Resource Name
HTML	Hypertext Markup Language	UUID	Universally Unique Identifier
HTTP	Hypertext Trrdansfer Protocol	XML	Extensible Markup Language
SCPD	Service Control Protocol Description		

Glossary

action
Command exposed by a service. Takes one or more input or output arguments. May have a return value. For more information, see clause 2, “Description” and clause 3, “Control”.

argument
Parameter for action exposed by a service. May be in or out. For more information, see clause 2, “Description” and clause 3, “Control”.

control point
Retrieves device and service descriptions, sends actions to services, polls for service state variables, and receives events from services.

device
Logical device. A container. May embed other logical devices. Embeds one or more services. Advertises its presence on network(s). For more information, see clause 1, “Discovery” and clause 2, “Description”.

device description
Formal definition of a logical device, expressed in the UPnP Template Language. Written in XML syntax. Specified by a UPnP vendor by filling in the placeholders in a UPnP Device Template, including, e.g., manufacturer name, model name, model number, serial number, and URLs for control, eventing, and presentation. For more information, see clause 2, “Description”.

device type
Standard device types are denoted by urn:schemas-upnp-org:device: followed by a unique name assigned by a UPnP Forum working committee. One-to-one relationship with UPnP Device Templates. UPnP vendors may specify additional device types; these are denoted by urn:*domain-name*:device: followed by a unique name assigned by the vendor, where *domain-name* is a Vendor Domain Name. For more information, see clause 2, “Description” .

event
Notification of one or more changes in state variables exposed by a service. For more information, see clause 4, “Eventing”.

GENA
General Event Notification Architecture. The event subscription and notification protocol defined in clause 4, “Eventing”.

publisher
Source of event messages. Typically a device's service. For more information, see clause 4, “Eventing”.

root device
A logical device that is not embedded in any other logical device. For more information, see clause 2, “Description”.

service
Logical functional unit. Smallest units of control. Exposes actions and models the state of a physical device with state variables. For more information, see clause 3, “Control”.

© ISO/IEC 2017 – All rights reserved

service description
Formal definition of a logical service, expressed in the UPnP Template language. Written in XML syntax. Specified by a UPnP vendor by filling in any placeholders in a UPnP Service Template. (Was SCPD.) For more information, see clause 2, “Description”.

service type
Standard service types are denoted by urn:schemas-upnp-org:service: followed by a unique name assigned by a UPnP forum working committee, colon, and an integer version number. One-to-one relationship with UPnP Service Templates. UPnP vendors may specify additional services; these are denoted by urn:*domain-name*:service: followed by a unique name assigned by the vendor, colon, and a version number, where *domain-name* is a Vendor Domain Name. For more information, see clause 2, “Description”.

SOAP
Simple Object Access Protocol. A remote-procedure call mechanism based on XML that sends commands and receives values over HTTP. For more information, see clause 3, “Control”.

SSDP
Simple Service Discovery Protocol. A multicast discovery and search mechanism that uses a multicast variant of HTTP over UDP. Defined in clause 1, “Discovery”.

state variable
Single facet of a model of a physical service. Exposed by a service. Has a name, data type, optional default value, optional constraints values, and may trigger events when its value changes. For more information, see clause 2, “Description” and clause 3, “Control”.

subscriber
Recipient of event messages. Typically a control point. For more information, see clause 4, “Eventing”.

UPnP Device Template
Template listing device type, required embedded devices (if any), and required services. Written in XML syntax and derived from the UPnP Device Schema. Defined by a UPnP Forum working committee. One-to-one relationship with standard device types. For more information, see clause 2, “Description”.

UPnP Service Template
Template listing action names, parameters for those actions, state variables, and properties of those state variables. Written in XML syntax and derived from the UPnP Service Schema. Defined by a UPnP Forum working committee. One-to-one relationship with standard service types. For more information, see clause 2, “Description”.

UPnP Device Schema
Defines the elements and attributes used in UPnP Device and Service Templates. Written in XML syntax and derived from XML Schema (Part 1: Structures, Part 2: Datatypes). Defined by the UPnP Device Architecture herein. For more information, see clause 2, “Description”.

Vendor Domain Name
A domain name that is supplied by a vendor. It is owned by the vendor, and shall be registered with an ICANN accredited Registrar, such that it is unique. The vendor shall keep the domain name registration up to date. A Vendor Domain Name length should be chosen to be compatible with the use of the domain name in the UDA.

References and resources

RFC 2710, Multicast Listener Discovery for IPv6. Available at: http://www.ietf.org/rfc/rfc2710.txt.

RFC 2616, HTTP: Hypertext Transfer Protocol 1.1. Available at: http://www.ietf.org/rfc/rfc2616.txt.

 © ISO/IEC 2017 – All rights reserved

RFC 2279, UTF-8, a transformation format of ISO 10646 (character encoding). Available at: http://www.ietf.org/rfc/rfc2279.txt.

XML, Extensible Markup Language. W3C recommendation. Available at: http://www.w3.org/XML/.

DEVICEPROTECTION, UPnP Device Protection specification. Available at http://upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

Each clause in this document contains additional information about resources for specific topics.

0 Addressing

Addressing is Step 0 of UPnP networking. Through addressing, devices and control points get a network address. Addressing enables discovery (Step 1) where control points find interesting device(s), description (Step 2) where control points learn about device capabilities, control (Step 3) where a control point sends commands to device(s), eventing (Step 4) where control points listen to state changes in device(s), and presentation (Step 5) where control points display a user interface for device(s).

The foundation for UPnP networking is IP addressing. A UPnP device or control point is allowed to support IP version 4-only, or both IP version 4 and IP version 6. This clause, and the examples given throughout clauses 1 through 5 of this document, assumes an IPv4 implementation. Annex A of this document describes IPv6 operation. Each UPnP device or control point which does not itself implement a DHCP server shall have a Dynamic Host Configuration Protocol (DHCP) client and search for a DHCP server when the device or control point is first connected to the network (if the device or control point itself implements a DHCP server, it allowed to allocate itself an address from the pool that it controls). If a DHCP server is available, i.e., the network is managed; the device or control point shall use the IP address assigned to it. If no DHCP server is available, i.e., the network is unmanaged; the device or control point shall use automatic IP addressing (Auto-IP) to obtain an address.

Auto-IP (defined in RFC 3927) defines how a device or control point: (a) determines if DHCP is unavailable, and (b) intelligently chooses an IP address from a set of link-local IP addresses. This method of address assignment enables a device or control point to easily move between managed and unmanaged networks.

This clause provides an overview of the basic operation of Auto-IP. The operations described in this clause are detailed and clarified in the reference documents listed below. Where conflicts between this document and the reference documents exist, the reference document always takes precedence.

0.1 Determining whether to use Auto-IP

A device or control point that supports Auto-IP and is configured for dynamic address assignment begins by requesting an IP address via DHCP by sending out a DHCPDISCOVER message. The amount of time this DHCP Client listens for DHCPOFFERs is implementation dependent. If a DHCPOFFER is received during this time, the device or control point shall continue the process of dynamic address assignment. If no valid DHCPOFFERs are received, the device or control point shall then auto-configure an IP address using Auto-IP.

0.2 Choosing an address

To auto-configure an IP address using Auto-IP, the device or control point uses an implementation dependent algorithm for choosing an address in the 169.254/16 range. The first and last 256 addresses in this range are reserved and shall NOT be used.

The selected address shall then be tested to determine if the address is already in use. If the address is in use by another device or control point, another address shall be chosen and tested, up to an implementation dependent number of retries. The address selection shall be randomized to avoid collision when multiple devices or control points are attempting to allocate addresses. The device or control point chooses an address using a pseudo-random

© ISO/IEC 2017 – All rights reserved

algorithm (distributed over the entire address range from 169.254.1.0 to 169.254.254.255) to minimize the likelihood that devices or control points that join the network at the same time will choose the same address and subsequently choose alternative addresses in the same sequence when collisions are detected. This pseudo-random algorithm should be seeded using the device's or control point's Ethernet hardware MAC address.

0.3 Testing the address

To test the chosen address, the device or control point shall use an Address Resolution Protocol (ARP) probe. An ARP probe is an ARP request with the device or control point hardware address used as the sender's hardware address and the sender's IP address set to 0s. The device or control point shall then listen for responses to the ARP probe, or other ARP probes for the same IP address. If either of these ARP packets is seen, the device or control point shall consider the address in use and try a different address. The ARP probe is allowed to be repeated for greater certainty that the address is not already in use; it is recommended that the probe be sent four times at two-second intervals.

After successfully configuring a link-local address, the device or control point shall send two gratuitous ARPs, spaced two seconds apart, this time filling in the sender IP address. The purpose of these gratuitous ARPs is to make sure that other hosts on the net do not have stale ARP cache entries left over from some other host that may previously have been using the same address.

Devices and control points that are equipped with persistent storage are allowed to record the IP address they have selected and on the next boot use that address as their first candidate when probing, in order to increase the stability of addresses and reduce the need to resolve address conflicts.

Address collision detection is not limited to the address testing phase, when the device or control point is sending ARP probes and listening for replies. Address collision detection is an ongoing process that is in effect for as long as the device or control point is using a link-local address. At any time, if a device or control point receives an ARP packet with its own IP address given as the sender IP address, but a sender hardware address that does not match its own hardware address, then the device or control point shall treat this as an address collision and shall respond as described in either a) or b) below:

a) Immediately configure a new link-local IP address as described above; or,

b) If the device or control point currently has active TCP connections or other reasons to prefer to keep the same IP address, and has not seen any other conflicting ARP packets recently (e.g., within the last ten seconds) then it is allowed to elect to attempt to defend its address once, by recording the time that the conflicting ARP packet was received, and then broadcasting one single gratuitous ARP, giving its own IP and hardware addresses as the source addresses of the ARP. However, if another conflicting ARP packet is received within a short time after that (e.g., within ten seconds) then the device or control point shall immediately configure a new Auto-IP address as described above.

The device or control point shall respond to conflicting ARP packets as described in either a) or b) above; it shall NOT ignore conflicting ARP packets. If a new address is selected, the device or control point shall cancel previous advertisements and re-advertise with the new address.

After successfully configuring an Auto-IP address, all subsequent ARP packets (replies as well as requests) containing an Auto-IP source address shall be sent using link-level *broadcast* instead of link-level *unicast*, in order to facilitate timely detection of duplicate addresses.

0.4 Forwarding rules

IP packets whose source or destination addresses are in the 169.254/16 range shall NOT be sent to any router for forwarding. Instead, the senders shall ARP for the destination address and then send the packets directly to the destination on the same link. IP datagrams with a multicast destination address and an Auto-IP source address shall NOT be forwarded off the local link. Devices and control points are allowed to assume that all 169.254/16 destination

 © ISO/IEC 2017 – All rights reserved

addresses are on-link and directly reachable. The 169.254/16 address range shall not be subnetted.

0.5 Periodic checking for dynamic address availability

A device or control point that has auto-configured an IP address shall periodically check for the existence of a DHCP server. This is accomplished by sending DHCPDISCOVER messages. How often this check is made is implementation dependent, but checking every 5 minutes would maintain a balance between network bandwidth required and connectivity maintenance. If a DHCPOFFER is received, the device or control point shall proceed with dynamic address allocation. Once a DHCP assigned address is in place, the device or control point is allowed to release the auto-configured address, but is also allowed to choose to maintain this address for a period of time (or indefinitely) to maintain connectivity.

To switch over from one IP address to a new one, the device should, if possible, cancel any outstanding advertisements made on the previous address and shall issue new advertisements on the new address. The clause on Discovery explains advertisements and their cancellations. In addition, any event subscriptions are deleted by the device (see clause on Eventing).

For a multi-homed device with multiple IP addresses, to switch one of the IP addresses to a new one, the device should cancel any outstanding advertisements made on the previous IP address, and shall issue new advertisements on the new IP addresses. Furthermore, it shall also issue appropriate update advertisements on all unaffected IP addresses. The clause on Discovery explains advertisements, their cancellations and updates. The clause on Eventing explains the effect on event subscriptions.

0.6 Device naming and DNS interaction

Once a device has a valid IP address for the network, it can be located and referenced on that network through that address. There may be situations where the end user needs to locate and identify a device. In these situations, a friendly name for the device is much easier for a human to use than an IP address. If a device chooses to provide a host name to a DHCP server and register with a DNS server, the device should either ensure the requested host name is unique or provide a means for the user to change the requested host name. Most often, devices do not provide a host name, but provide URLs using literal (numeric) IP addresses.

Moreover, names are much more static than IP addresses. Clients referring a device by name don't require any modification when the IP address of a device changes. Mapping of the device's DNS name to its IP address could be entered into the DNS database manually or dynamically according to RFC 2136. While devices supporting dynamic DNS updates can register their DNS records directly in the DNS, it is also possible to configure a DHCP server to register DNS records on behalf of these DHCP clients.

0.7 Name to IP address resolution

A device that needs to contact another device identified by a DNS name needs to discover its IP address. The device submits a DNS query according to RFC1034 and 1035 to the pre-configured DNS server(s) and receives a response from a DNS server containing the IP address of the target device. A device can be statically pre-configured with the list of DNS servers. Alternatively a device could be configured with the list of DNS server through DHCP, or after the address assignment through a DHCPINFORM message.

0.8 References

RFC1034, Domain Names - Concepts and Facilities. Available at: http://www.ietf.org/rfc/rfc1034.txt.

RFC1035, Domain Names - Implementation and Specification. Available at: http://www.ietf.org/rfc/rfc1035.txt.

© ISO/IEC 2017 – All rights reserved

RFC 2131, Dynamic Host Configuration Protocol. Available at: http://www.ietf.org/rfc/rfc2131.txt.

RFC 2136, Dynamic Updates in the Domain Name System. Available at: http://www.ietf.org/rfc/rfc2136.txt.

RFC 3927, Dynamic Configuration of IPv4 Link-Local Addresses. Available at: http://www.ietf.org/rfc/rfc3927.txt.

1 Discovery

Discovery is Step 1 in UPnP™ networking. Discovery comes after addressing (Step 0) where devices get a network address. Through discovery, control points find interesting device(s). Discovery enables description (Step 2) where control points learn about device capabilities, control (Step 3) where a control point sends commands to device(s), eventing (Step 4) where control points listen to state changes in device(s), and presentation (Step 5) where control points display a user interface for device(s).

Discovery is the first step in UPnP networking. When a device is added to the network, the UPnP discovery protocol allows that device to advertise its services to control points on the network. Similarly, when a control point is added to the network, the UPnP discovery protocol allows that control point to search for devices of interest on the network. The fundamental exchange in both cases is a discovery message containing a few, essential specifics about the device or one of its services, e.g., its type, universally unique identifier, a pointer to more detailed information and optionally parameters that identify the current state of the device.

 © ISO/IEC 2017 – All rights reserved