
**Information technology — UPnP
Device Architecture —**
Part 26-10:
**Telephony device control protocol —
Level 2 — Call management service**

*Technologies de l'information — Architecture de dispositif UPnP —
Partie 26-10: Protocole de contrôle de dispositif de téléphonie —
Niveau 2 — Service de gestion des appels*



STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 29341-26-10:2017



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

1	Scope.....	1
2	Normative References	1
3	Terms, definitions, symbols and abbreviated terms.....	2
4	Notations and conventions	5
4.1	Text conventions	5
4.2	Data Types	5
4.3	Vendor-defined Extensions	6
5	Service Modeling Definitions	6
5.1	Service Type	6
5.2	<u>CallManagement</u> Service Architecture	6
5.2.1	Managing Call With Multiple Telephony Clients	6
5.2.2	User Concept and Multi-User Control.....	10
5.2.3	Call Monopolization.....	11
5.2.4	Selective Information Delivery	16
5.2.5	Handling of Media Streams.....	16
5.2.6	Content Sharing	17
5.3	State Variables	18
5.3.1	State Variable Overview	18
5.3.2	<u>CallInfo</u>	19
5.3.3	<u>TelCPNameList</u>	23
5.3.4	<u>CallBackAvailability</u>	23
5.3.5	<u>PushInfo</u>	23
5.3.6	<u>VoiceMailInfo</u>	24
5.3.7	<u>ParallelCallInfo</u>	25
5.3.8	<u>A ARG TYPE TelephonyServerIdentity</u>	26
5.3.9	<u>A ARG TYPE TelCPName</u>	26
5.3.10	<u>A ARG TYPE TelCPNameList</u>	26
5.3.11	<u>A ARG TYPE Expires</u>	26
5.3.12	<u>A ARG TYPE MediaCapabilityInfo</u>	26
5.3.13	<u>A ARG TYPE CalleeID</u>	28
5.3.14	<u>A ARG TYPE CallPriority</u>	28
5.3.15	<u>A ARG TYPE CallMode</u>	29
5.3.16	<u>A ARG TYPE CallID</u>	29
5.3.17	<u>A ARG TYPE SecretKey</u>	29
5.3.18	<u>A ARG TYPE RejectReason</u>	29
5.3.19	<u>A ARG TYPE TCList</u>	29
5.3.20	<u>A ARG TYPE CallInfoList</u>	30
5.3.21	<u>A ARG TYPE CallLogs</u>	31
5.3.22	<u>A ARG TYPE CallBackID</u>	32
5.3.23	<u>A ARG TYPE CallBackInfoList</u>	32
5.3.24	<u>A ARG TYPE PushInfoList</u>	33
5.3.25	<u>A ARG TYPE VoiceMailInfoList</u>	34
5.3.26	<u>A ARG TYPE VoiceMailID</u>	34

ISO/IEC 29341-26-10:2017(E)

5.3.27	<u>A ARG TYPE CallType</u>	34
5.3.28	<u>A ARG TYPE CallerID</u>	35
5.3.29	<u>A ARG TYPE MaxWaitingTime</u>	36
5.4	Eventing and Moderation	36
5.4.1	Eventing of <u>CallInfo</u>	36
5.4.2	Eventing of <u>TelCPNameList</u>	37
5.4.3	Eventing of <u>CallBackAvailability</u>	37
5.4.4	Eventing of <u>PushInfo</u>	37
5.4.5	Eventing of <u>VoiceMailInfo</u>	37
5.4.6	Eventing of <u>ParallelCallInfo</u>	37
5.5	Actions	37
5.5.1	<u>GetTelephonyIdentity()</u>	38
5.5.2	<u>RegisterTelCPName()</u>	39
5.5.3	<u>UnregisterTelCPName()</u>	41
5.5.4	<u>ChangeTelCPName()</u>	41
5.5.5	<u>GetTelCPNameList()</u>	42
5.5.6	<u>GetMediaCapabilities()</u>	43
5.5.7	<u>StartCall()</u>	44
5.5.8	<u>StopCall()</u>	46
5.5.9	<u>AcceptCall()</u>	48
5.5.10	<u>RejectCall()</u>	50
5.5.11	<u>ModifyCall()</u>	52
5.5.12	<u>AcceptModifyCall()</u>	54
5.5.13	<u>StartMediaTransfer()</u>	56
5.5.14	<u>ChangeMonopolizer()</u>	57
5.5.15	<u>InitiateCall()</u>	59
5.5.16	<u>GetCallInfo()</u>	59
5.5.17	<u>GetCallLogs()</u>	61
5.5.18	<u>ClearCallLogs()</u>	61
5.5.19	<u>RegisterCallBack()</u>	62
5.5.20	<u>ClearCallBack()</u>	63
5.5.21	<u>GetCallBackInfo()</u>	63
5.5.22	<u>ChangeCallMode()</u>	64
5.5.23	<u>GetPushInfo()</u>	65
5.5.24	<u>IgnoreCall()</u>	66
5.5.25	<u>GetVoiceMail()</u>	67
5.5.26	<u>DeleteVoiceMail()</u>	68
5.5.27	<u>EnhancedInitiateCall()</u>	69
5.5.28	<u>WaitingForCall()</u>	70
5.5.29	<u>InitiateParallelCall()</u>	71
5.5.30	<u>AcceptParallelCall()</u>	72
5.5.31	Relationships Between Actions	73
5.5.32	Error Code Summary	73
5.6	Service Behavioral Model	73
5.6.1	State Diagram	73
6	XML Service Description	77
Annex A (normative)	XML complex type <i>peerType</i>	93

Annex B (normative) XML Schema	97
Annex C (informative) Theory of Operation	102
Annex D (informative) Sequence Examples	125
Annex E (informative) How to send DTMF	295
Annex F (informative) Bibliography	296
Figure 1 — Relationship among Media Stream, Media Session and Media Session ID	3
Figure 2 — Architecture of the <u>CallManagement</u> Service	6
Figure 3 — Managing a Telephony Call with Multiple Telephony Clients	7
Figure 4 — Starting a Call with Multiple Telephony Clients Using Media Mixing Capability	8
Figure 5 — Modifying a Call with Multiple Telephony Clients Using Media Mixing Capability	8
Figure 6 — Starting a Call With Multiple Telephony Clients Using Media Sharing Capability	9
Figure 7 — Modifying a Call With Multiple Telephony Clients Using Media Sharing Capability	10
Figure 8 — Example for Multi-User Control Concept	11
Figure 9 — PHONE-based Call Monopolization	13
Figure 10 — Incoming Call Handling	15
Figure 11 — Call State Diagram for a Caller	74
Figure 12 — Call State Diagram for a Callee	75
Figure 13 — Call State Diagram to request the modification of a Call	76
Figure 14 — Call State Diagram to accept the modification of a Call	77
Figure C.1 — Flow basics for creating an outgoing Call (TC-Based Media Handling)	103
Figure C.2 — Flow basics for accepting an incoming Call (TC-Based Media Handling)	105
Figure C.3 — Flow basics for modifying an ongoing Call (TC-Based Media Handling)	107
Figure C.4 — Flow basics for creating an outgoing Call (TS-Based Media Handling)	109
Figure C.5 — Flow basics for accepting an incoming Call (TS-Based Media Handling)	111
Figure C.6 — Flow basics for modifying an ongoing Call (TS-Based Media Handling)	113
Figure C.7 — Flow basics for changing CallMode from Non-Monopolize to Monopolize	114
Figure C.8 — Flow basics for changing CallMode from Monopolize to Non-Monopolize	115
Figure C.9 — Flow basics for Early Media	116
Figure C.10 — Example flow of TS-Based Content Sharing by initiating a Call	118
Figure C.11 — Example flow of TS-Based Content Sharing in an existing Call	119
Figure C.12 — Example flow of TC-Based Content Sharing by Updating a Call	120
Figure C.13 — Example flow of TC-Based Content Sharing without Updating a Call	121
Figure C.14 — Example flow of TC-Based Content Sharing by initiating a Call	122
Figure C.15 — Architecture for Parallel Call	123
Figure C.16 — Flow basics for Establishing a Parallel Call	124
Figure D.1 — Register TelCPName	125
Figure D.2 — Register TelCPName (But the specified TelCPName is already in use.)	126
Figure D.3 — Keep using the same TelCPName	127

ISO/IEC 29341-26-10:2017(E)

Figure D.4 — Keep using the same TelCPName (But the specified Secret Key is invalid.)	128
Figure D.5 — Change TelCPName	129
Figure D.6 — Change TelCPName (But the specified Secret Key is invalid.)	130
Figure D.7 — Change TelCPName (But the specified TelCPName is already in use.)	131
Figure D.8 — Unregister TelCPName	132
Figure D.9 — Unregister TelCPName (But the specified TelCPName does not exist.)	133
Figure D.10 — Get existing TelCPNames	134
Figure D.11 — Get existing TelCPNames (But no TelCP exists)	135
Figure D.12 — Create a Call	137
Figure D.13 — Terminate a Call	153
Figure D.14 — Reject an incoming Call	158
Figure D.15 — Cancel an outgoing Call	161
Figure D.16 — Cancel an outgoing Call (But the specified CallID does not exist.)	164
Figure D.17 — Cancel an outgoing Call (But the StopCall() action is invoked at invalid timing.)	165
Figure D.18 — Get MediaSessionInfo and CallInfo during the Call	166
Figure D.19 — Create an outgoing Call using multiple TCs (Case 1)	169
Figure D.20 — Create an outgoing Call using multiple TCs (Case 2)	183
Figure D.21 — Accept an incoming Call using multiple TCs (Case 1)	188
Figure D.22 — Accept an incoming Call using multiple TCs (Case 2)	201
Figure D.23 — Add TC during a Call (Create a modification request)	205
Figure D.24 — Add TC during a Call (Accept a modification request)	215
Figure D.25 — Remove TC during a Call (Create a modification request)	224
Figure D.26 — Remove TC during a Call (Accept a modification request)	230
Figure D.27 — Change TC during a Call	237
Figure D.28 — Modify Media Session during a Call	246
Figure D.29 — Initiate a Call	254
Figure D.30 — Create an outgoing Call (Monopolization Mode Call(PHONE-TelCP))	257
Figure D.31 — Terminate a Call (But the specified TelCP is not authorized.)	272
Figure D.32 — Cancel an outgoing Call (But the specified TelCP is not authorized.)	273
Figure D.33 — Change TelCP which Monopolize the Call	274
Figure D.34 — Create a Call with TC-Based and TS-Based Media Handlings	277
Table 1 — State Variables	18
Table 2 — Allowed values for type	20
Table 3 — Allowed values for callStatus	20
Table 4 — Allowed values for reason and the corresponding values of the Call Status	21
Table 5 — Allowed values for priority	22
Table 6 — Allowed values for format	22
Table 7 — Allowed values for priority	24
Table 8 — Allowed values for informationType	26

Table 9 — allowedValueList for the <i>A_ARG_TYPE_CallPriority</i> state variable	28
Table 10 — allowedValueList for the <i>A_ARG_TYPE_CallMode</i> state variable	29
Table 11 — Allowed values for videoQuality	35
Table 12 — Allowed values for audioQuality	35
Table 13 — Event Moderation	36
Table 14 — Actions	37
Table 15 — Arguments for <i>GetTelephonyIdentity()</i>	38
Table 16 — Error Codes for <i>GetTelephonyIdentity()</i>	39
Table 17 — Arguments for <i>RegisterTelCPName()</i>	39
Table 18 — Error Codes for <i>RegisterTelCPName()</i>	40
Table 19 — Arguments for <i>UnregisterTelCPName()</i>	41
Table 20 — Error Codes for <i>UnregisterTelCPName()</i>	41
Table 21 — Arguments for <i>ChangeTelCPName()</i>	41
Table 22 — Error Codes for <i>ChangeTelCPName()</i>	42
Table 23 — Arguments for <i>GetTelCPNameList()</i>	43
Table 24 — Error Codes for <i>GetTelCPNameList()</i>	43
Table 25 — Arguments for <i>GetMediaCapabilities()</i>	43
Table 26 — Error Codes for <i>GetMediaCapabilities()</i>	44
Table 27 — Arguments for <i>StartCall()</i>	44
Table 28 — Error Codes for <i>StartCall()</i>	46
Table 29 — Arguments for <i>StopCall()</i>	46
Table 30 — Error Codes for <i>StopCall()</i>	48
Table 31 — Arguments for <i>AcceptCall()</i>	48
Table 32 — Error Codes for <i>AcceptCall()</i>	50
Table 33 — Arguments for <i>RejectCall()</i>	50
Table 34 — Error Codes for <i>RejectCall()</i>	52
Table 35 — Arguments for <i>ModifyCall()</i>	52
Table 36 — Error Codes for <i>ModifyCall()</i>	53
Table 37 — Arguments for <i>AcceptModifyCall()</i>	54
Table 38 — Error Codes for <i>AcceptModifyCall()</i>	55
Table 39 — Arguments for <i>StartMediaTransfer()</i>	56
Table 40 — Error Codes for <i>StartMediaTransfer()</i>	57
Table 41 — Arguments for <i>ChangeMonopolizer()</i>	57
Table 42 — Error Codes for <i>ChangeMonopolizer()</i>	58
Table 43 — Arguments for <i>InitiateCall()</i>	59
Table 44 — Error Codes for <i>InitiateCall()</i>	59
Table 45 — Arguments for <i>GetCallInfo()</i>	59
Table 46 — Error Codes for <i>GetCallInfo()</i>	60
Table 47 — Arguments for <i>GetCallLogs()</i>	61
Table 48 — Error Codes for <i>GetCallLogs()</i>	61
Table 49 — Error Codes for <i>ClearCallLogs()</i>	62
Table 50 — Arguments for <i>RegisterCallBack()</i>	62

ISO/IEC 29341-26-10:2017(E)

Table 51 — Error Codes for <i>RegisterCallBack()</i>	63
Table 52 — Arguments for <i>ClearCallBack()</i>	63
Table 53 — Error Codes for <i>ClearCallBack()</i>	63
Table 54 — Arguments for <i>GetCallBackInfo()</i>	63
Table 55 — Error Codes for <i>GetCallBackInfo()</i>	64
Table 56 — Arguments for <i>ChangeCallMode()</i>	64
Table 57 — Error Codes for <i>ChangeCallMode()</i>	65
Table 58 — Arguments for <i>GetPushInfo()</i>	65
Table 59 — Error Codes for <i>GetPushInfo()</i>	66
Table 60 — Arguments for <i>IgnoreCall()</i>	66
Table 61 — Error Codes for <i>IgnoreCall()</i>	67
Table 62 — Arguments for <i>GetVoiceMail()</i>	67
Table 63 — Error Codes for <i>GetVoiceMail()</i>	68
Table 64 — Arguments for <i>DeleteVoiceMail()</i>	68
Table 65 — Error Codes for <i>DeleteVoiceMail()</i>	69
Table 66 — Arguments for <i>EnhancedInitiateCall()</i>	69
Table 67 — Error Codes for <i>EnhancedInitiateCall()</i>	70
Table 68 — Arguments for <i>WaitingForCall()</i>	70
Table 69 — Error Codes for <i>WaitingForCall()</i>	71
Table 70 — Arguments for <i>InitiateParallelCall()</i>	71
Table 71 — Error Codes for <i>InitiateParallelCall()</i>	72
Table 72 — Arguments for <i>AcceptParallelCall()</i>	72
Table 73 — Error Codes for <i>AcceptParallelCall()</i>	72
Table 74 — Error Code Summary	73

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 Standard, 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 29341-26-10 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](#).

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

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 29341-26-10:2017(E)

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
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

ISO/IEC 29341-26-10:2017(E)

UPnP Low Power Architecture:1	ISO/IEC 29341-16-1:2011
UPnP LowPowerProxy:1 Service	ISO/IEC 29341-16-10:2011
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

ISO/IEC 29341-26-10:2017(E)

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
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

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 29341-26-10:2017

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 29341-26-10:2017

1 Scope

This service definition is compliant with [1]. It defines a service type referred to herein as the CallManagement service.

The CallManagement service is a UPnP service that allows control points to use the telephony features(e.g., voice call, video call, and data transfer etc.) provided by a Telephony Server (TS).

The CallManagement service enables the following features to a Telephony Control Point (TelCP):

- start an outgoing call
- accept an incoming call
- modify the capability of an existing call (e.g., changing from a voice call to a video call)
- terminate a call
- preview call logs
- register a call back and check the availability of the registered call back

To realize these features, the CallManagement service manages the connectivity of calls and media streams.

2 Normative References

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

[1] – UPnP Device Architecture, version 1.0, UPnP Forum, October 15, 2008. Available at: <http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0-20081015.pdf>. Latest version available at: <http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0.pdf>.

[2] – Data elements and interchange formats – Information interchange -- Representation of dates and times, International Standards Organization, December 21, 2000. Available at: [ISO 8601:2000](http://www.iso.org/iso/8601.html).

[3] – IETF RFC 2396, Uniform Resource Identifiers (URI): Generic Syntax, T. Berners-Lee, MIT/LCS, R. Fielding, U.C. Irvine, L. Masinter, Xerox Corporation, 1998. Available at: <http://www.ietf.org/rfc/rfc2396.txt>.

[4] – IETF RFC 3261, SIP: Session Initiation Protocol, J. Rosenberg, dynamicsoft, H. Schulzrinne, Columbia U., G. Camarillo, Ericsson, A. Johnston, WorldCom, J. Peterson, Neustar, R. Sparks, dynamicsoft, M. Handley, ICIR, E. Schooler, AT&T, 2002. Available at: <http://www.ietf.org/rfc/rfc3261.txt>.

[5] – IETF RFC 3339, Date and Time on the Internet: Timestamps, G. Klyne, Clearswift Corporation, C. Newman, Sun Microsystems, July 2002. Available at: <http://www.ietf.org/rfc/rfc3339.txt>.

[6] – IETF RFC 3966, The tel URI for Telephone Numbers, H. Schulzrinne, Columbia University, 2004. Available at: <http://www.ietf.org/rfc/rfc3966.txt>.

[7] – IETF RFC 4566, SDP: Session Description Protocol, M. Handley, UCL, V. Jacobson, Packet Design, C. Perkins, University of Glasgow, July 2006. Available at: <http://www.ietf.org/rfc/rfc4566.txt>.

ISO/IEC 29341-26-10:2017(E)

[8] – Extensible Markup Language (XML) 1.0 (Third Edition), François Yergeau, Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, eds., W3C Recommendation, February 4, 2004. Available at: <http://www.w3.org/TR/2004/REC-xml-20040204>.

[9] – XML Schema Part 2: Data Types, Second Edition, Paul V. Biron, Ashok Malhotra, W3C Recommendation, 28 October 2004. Available at: <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028>.

[10] – *MediaManagement:2*, UPnP Forum, December 10, 2012. Available at: <http://www.upnp.org/specs/phone/UPnP-phone-MediaManagement-v2-Service-20121210.pdf>. Latest version available at: <http://www.upnp.org/specs/phone/UPnP-phone-MediaManagement-Service.pdf>.

[11] – IETF RFC 2046, Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types, N. Freed, Innosft, N. Borenstein, First Virtual, November 1996. Available at: <http://www.ietf.org/rfc/rfc2046.txt>.

[12] – *ConnectionManager:1*, UPnP Forum, June 25, 2002. Available at: <http://www.upnp.org/specs/av/UPnP-av-ConnectionManager-v1-Service-20020625.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-av-ConnectionManager-Service.pdf>.

3 Terms, definitions, symbols and abbreviated terms

For the purposes of this document, the terms and definitions given in [1] and the following apply.

3.1 Provisioning terms

3.1.1

conditionally allowed

CA

The definition or behavior depends on a condition. If the specified condition is met, then the definition or behavior is allowed, otherwise it is not allowed.

3.1.2

conditionally required

CR

The definition or behavior depends on a condition. If the specified condition is met, then the definition or behavior is required, otherwise it is not allowed.

3.1.3

not allowed

The definition or behavior is prohibited by this specification. Opposite of required.

3.2 Symbols

3.2.1

::

signifies a hierarchical parent-child (parent::child) relationship between the two objects separated by the double colon. This delimiter is used in multiple contexts, for example: Service::Action(), Action()::Argument, parentProperty::childProperty.

3.3 Terms

3.3.1

Call

connection between parties that are communicating with each other. The connection can be for voice, video, or data transmission.

3.3.2

CallID

unique identifier of a Call.

3.3.3**Call Status**

indicator of the state(e.g., Calling, Ringing, and Talking etc.) of a Call at a particular point in time.

3.3.4**Media Stream**

a flow of media (e.g., audio, video, and data etc.) which is sent and/or received between two parties involved in a Call.

3.3.5**Media Session**

series of interactions for flow of media among the parties involved in the Call.

3.3.6**Media Session ID**

unique identifier of the Media Session for a Call.

Figure 1 shows the relationship among Media Streams, Media Session and Media Session ID. A Media Session can include multiple Media Streams.

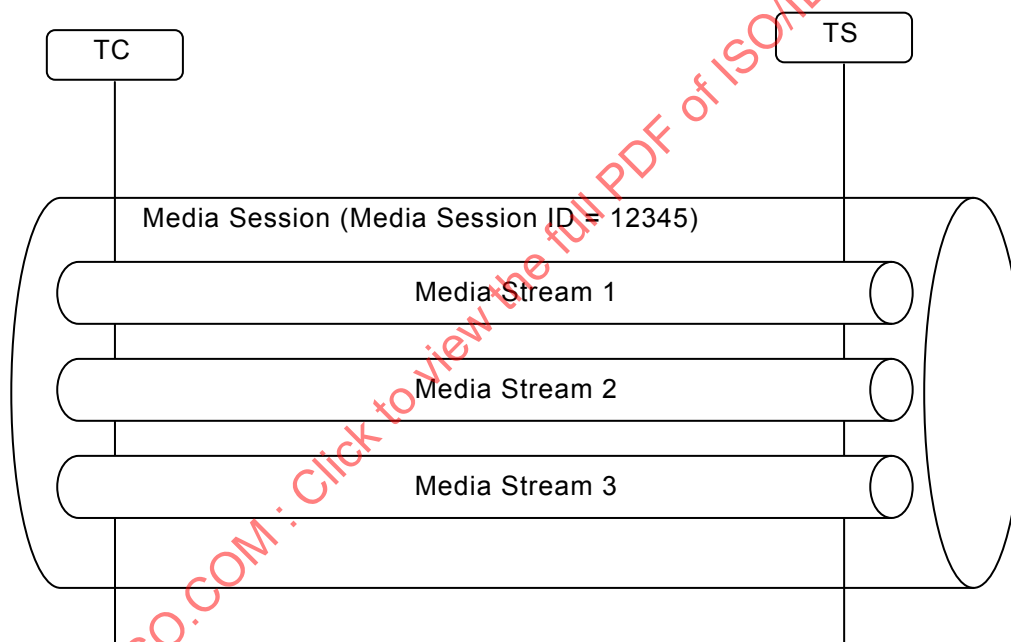


Figure 1 — Relationship among Media Stream, Media Session and Media Session ID

3.3.7**Media Capability**

Media Stream's attributes which are negotiated before sending/receiving a Media Stream in a Media Session. Media Stream's attributes include a codec type, a codec parameter, IP address, and a port number etc. [7] defines one of the methods to describe Media Capability. And the Media Capabilities represent attributes for a set of Media Streams.

3.3.8**Caller**

terminal which initiates a Call.

3.3.9**Callee**

terminal which receives a Call.

3.3.10

Remote Party

Callee, relative to the Caller

Caller, relative to the Callee

3.3.11

Peer

participant in a Call.

3.3.12

Monopolize

to obtain the exclusive rights to control a specific Call managed by the CallManagement service. When a Call is successfully Monopolized by an identity, the CallManagement service only allows this specific identity to manage the Call, until the Call is terminated or this identity hands over the exclusive rights to control the Call to another identity.

3.3.13

Monopolizer

identity (a TelCP or a user) that currently has the exclusive rights to control a specific Call managed by the CallManagement service.

3.3.14

Call Monopolization

mechanisms that allow an identity (a TelCP or a user) to Monopolize a Call managed by the CallManagement service. See 5.2.2 in detail.

3.3.15

Monopolization Mode Call

Call which is Monopolized.

3.3.16

Non-Monopolization Mode Call

Call which is not Monopolized. Therefore, any TelCP can control a Non-Monopolization Mode Call.

3.3.17

TelCPName

unique name for the TelCP

3.3.18

Secret Key

unique value assigned to a TelCP. The CallManagement service uses the Secret Key to verify whether an authorized TelCP is invoking the action.

3.3.19

Push Information

feature that enables a TS to event information received from a service provider on WAN side. This feature also allows to event some status changes that occurs within a TS.

3.3.20

Early Media

media (e.g., audio and video) that is sent and/or received before a Call is accepted or rejected by a Callee. Typical examples of Early Media generated by a Callee are ringing tone and announcements (e.g., queuing status). Early Media generated by a Caller typically consists of voice commands or dual tone multi-frequency (DTMF) tones to drive interactive voice response (IVR) systems.

3.3.21**Content Sharing**

feature that enables a TS or TC to share content (e.g., audio, video or image etc.) to the Remote Party using the Media Session during a Call. See 5.2.6 for more detail.

3.3.22**Parallel Call**

secondary Call established between the devices which are selected by the Peers already participating in a regular telephony Call.

3.3.23**Parallel Call Setup**

process to exchange identities of terminals which will establish Parallel Call.

3.4 Abbreviated terms**3.4.1****GUI**

Graphical User Interface

3.4.2**ID**

Identifier

3.4.3**SDP**

Session Description Protocol

3.4.4**TC**

Telephony Client

3.4.5**TelCP**

Telephony Control Point

3.4.6**TS**

Telephony Server

4 Notations and conventions**4.1 Text conventions**

- Strings that are to be taken literally are enclosed in “double quotes”.
- Words that are emphasized are printed in *italic*.
- Keywords that are defined by the UPnP Working Committee are printed using the forum character style.
- Keywords that are defined in [1] are printed using the arch character style.

4.2 Data Types

This specification uses data type definitions from two different sources. Data types from [1] are used to define state variable and action argument data types. The XML Schema data types [9] are used to define property data types.

For UPnP Device Architecture defined Boolean data types, it is strongly recommended to use the value “0” for false, and the value “1” for true. The values “true”, “yes”, “false”, or “no” may also be used but are not recommended. The values “yes” and “no” are deprecated and shall not be sent out by devices but shall be accepted on input.

For XML Schema defined Boolean data types, it is strongly recommended to use the value “0” for false, and the value “1” for true. The values “true”, “yes”, “false”, or “no” may also be used but are not recommended. The values “yes” and “no” are deprecated and shall not be sent out by devices but shall be accepted on input.

4.3 Vendor-defined Extensions

Whenever vendors create additional vendor-defined state variables, actions or properties, their assigned names and XML representation shall follow the naming conventions and XML rules as specified in [1], 2.5, “Description: Non-standard vendor extensions”.

5 Service Modeling Definitions

5.1 Service Type

The following service type identifies a service that is compliant with this specification.

urn:schemas-upnp-org:service:CallManagement:2

CallManagement service is used herein to refer to this service type.

5.2 CallManagement Service Architecture

This service provides the features for a TelCP to access telephony features that a TS exposes in the UPnP network in order to initiate a Call, accept an incoming Call or receive notifications for incoming Calls, and manage Media Session associated with the Call including starting and stopping the Media Session.

The architecture for the CallManagement service is shown in Figure 2.

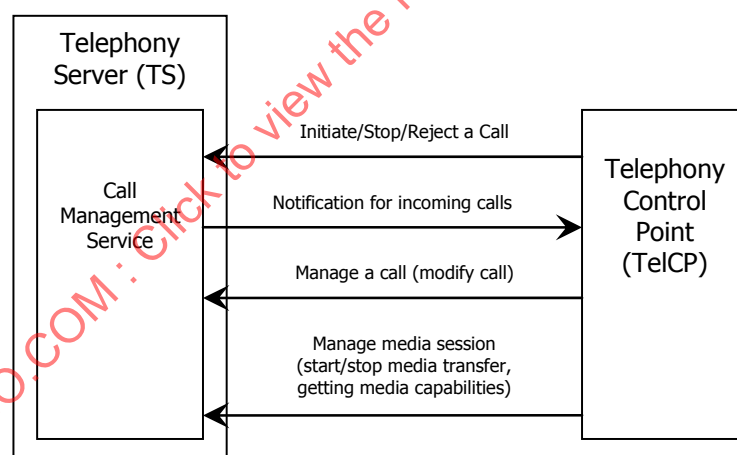


Figure 2 — Architecture of the CallManagement Service

It is assumed that the TelCP features are implemented in a physical device that provides a Graphical User Interface (GUI), for a user to manage locally the telephony functionalities. These are however out of the scope of the CallManagement service design.

The UPnP device that implements the CallManagement service is capable of exposing telephony features to the Peers through some telecommunication means (e.g., PSTN/ISDN, PLMN, and VoIP etc.): The implementation of these telephony features with Peers networks and Peers are out of the scope.

5.2.1 Managing Call With Multiple Telephony Clients

The main components of the telephony architecture are the Telephony Server (TS), Telephony Client (TC), and Telephony Control Point (TelCP). The TS is responsible for interfacing between the Caller or Callee in the wide area network and the TC in the home network. The

multiple Call architecture explained in 5.2.1 considers multiple TCs involve in a telephony Call. When multiple clients in the home network are associated with a Call involving a Callee/Caller in the wide area network, there is a need to share Media Stream among the the clients within the home network so that each client will receive Media Stream from all other clients in addition to the Media Stream from the Callee/Caller in the wide area network. The TS multiplexes Media Streams from multiple TCs in the home network and sends the multiplexed Media Stream to the WAN side.

Figure 3 shows two telephony clients are involved in a Call managed by the TS. The assumption here is that the TCs are in different rooms in the home network and are providing media for the Call with a Caller/Callee in the wide area network which gets media from both TC 1 and TC 2, TC 1 and TC 2 also need to get media from each other in order for the users behind them to listen to each other.

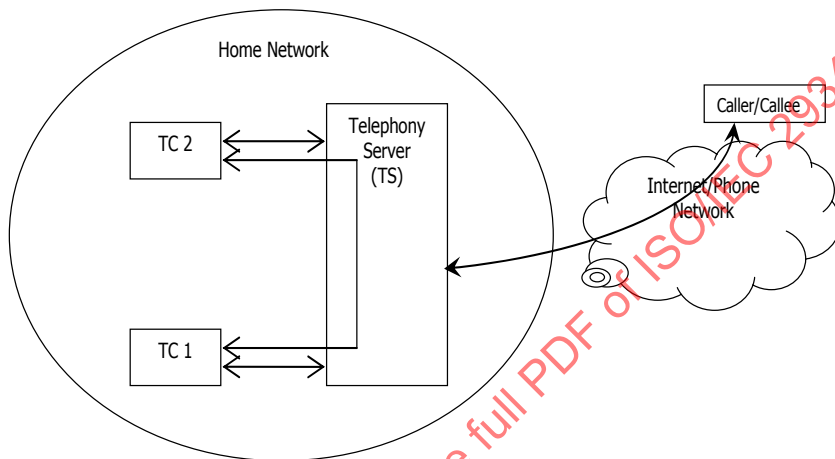


Figure 3 — Managing a Telephony Call with Multiple Telephony Clients

There are two different architectures that can manage Calls with multiple telephony clients. These are:

- a) The TS must make sure that the Media Streams from all the clients including the Callee/Caller in the wide area network are available to each TC. The TS can realize such a feature by performing the mixer functionality where one or more input Media Streams are combined to produce one or more output streams directed specifically to one or more TC. In another words, the TS multiplexes multiple Media Streams from different TCs involved in the Call and also media from the WAN side and sends the multiplexed Media Streams to the other TC involved in the Call. The architecture for sharing Media Stream among multiple telephony clients follows the star architecture where all the TCs are connected to the centralized TS in a star like fashion. The mixer is a software/hardware component resides inside the TS and is not visible in the UPnP network. Its functionality is to gather media inputs from multiple clients and produce output streams directed to each TC. In this approach, while initiating or accepting a Call, the TelCP can decide whether a TC will receive Media Stream from other TCs and whether it is capable of decoding the media from other clients by examining the Media Capability of the clients involved in the Call. Once the TelCP determines that the TC can receive Media Stream from other TCs, the TelCP indicates that option to the TS within the parameter of the StartMediaTransfer (CallID, MediaSessionID, MediaCapabilityInfo) action. The A_ARG_TYPE MediaCapabilityInfo state variable contains a XML tag that indicates whether the TS should mix media from other clients. The tag is: `<enableMediaMixing> 1 </enableMediaMixing>`. The value of this XML element is either 1 or 0. Where 1 indicates that that TS must mix Media Streams from other clients and a value of "0" indicates that the TS must stop mixing media from other TCs. Figure 4 and Figure 5 show how to use media mixing feature in starting a Call and modifying a Call. Figure 5 shows a scenario where the initial Call was established with only one client TC1. The Call is then modified by adding another client TC2 by invoking the ModifyCall() action and soon after the media

mixing capability is enabled on the TS by invoking the *StartMediaTransfer()* action. Once media mixing is enabled, the TS starts sending multiplexed media from to the TCs. The media mixing capability can be disabled or enabled at any time on the TS by invoking the *ModifyCall()* action with the value of the <enableMediaMixing> element set to either “0” or “1” respectively in the *A_ARG_TYPE MediaCapabilityInfo* input argument.

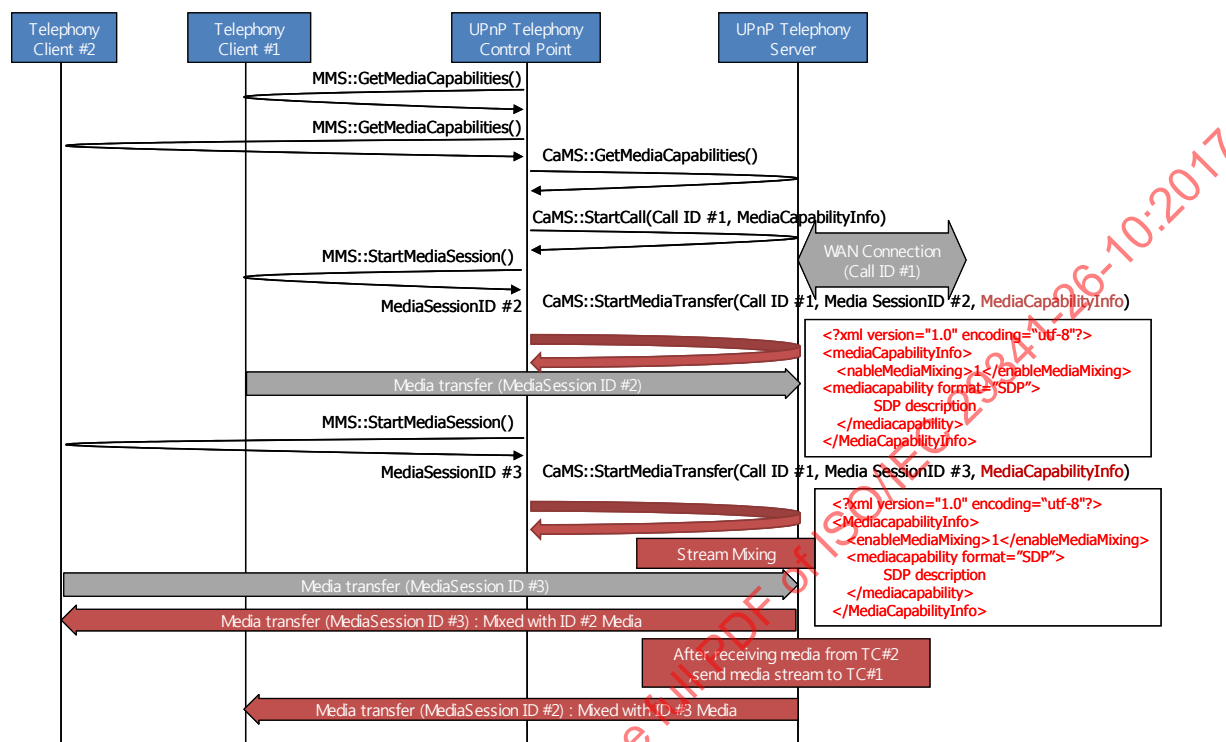


Figure 4 — Starting a Call with Multiple Telephony Clients Using Media Mixing Capability

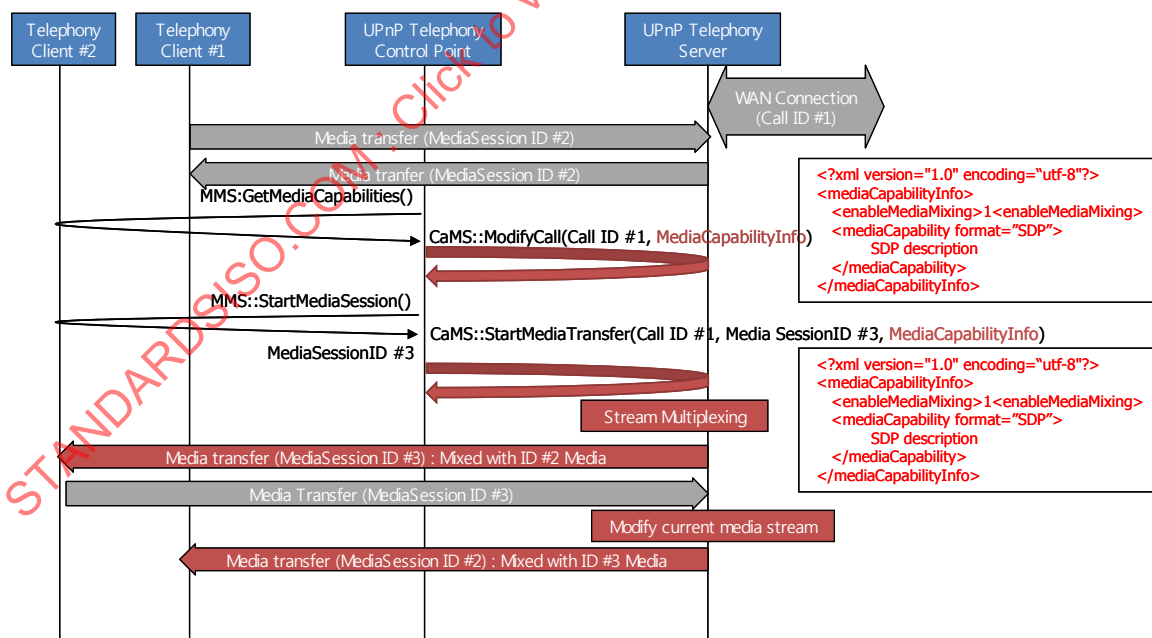


Figure 5 — Modifying a Call with Multiple Telephony Clients Using Media Mixing Capability

- b) An alternative architecture is to create multiple Media Sessions to share Media Streams among the TCs. The TelCP controls the multiple TCs and the TS in order to create

multiple sessions with each TCs. The second architecture requires a TelCP to create multiple Media Sessions. If there are two clients as shown in Figure 6 then the TelCP needs to create 4 different Media Sessions (Session 1 is between TC 1 and TS, Session 2 is between TC 2 and TS, Session 3 is between TS and TC 1 to receive media from TC 2, Session 4 is between TS and TC 2 to receive media from TC 1). The TelCP indicates to the TS whether a session is to share media from another TC by use of a XML tag in the A_ARG_TYPE MediaCapabilityInfo for the action StartMediaTransfer (CallID, MediaSessionID, MediaCapabilityInfo). The tag is: <shareMedia> Media Session ID </shareMedia>. The Media Session ID identifies the Media Stream to share in a Media Session with a TC. For example, if there are 2 telephony Clients TC1 and TC2 and the Media Session ID for TC1 and TS is 2, and Media Session ID for TC2 and TS is 3, then to create a new Media Session with Media Session ID 4 between TC 1 and TS to get media from TC2, the element will have the following value: <shareMedia>3 </shareMedia>. Similarly, to create a new Media Session with Media Session ID 5 between TC2 and TS and to get media from TC1, the element value will be: <shareMedia> 2 </shareMedia>. Figure 6 and Figure 7 show how to use media sharing feature while starting a Call or modifying a Call.

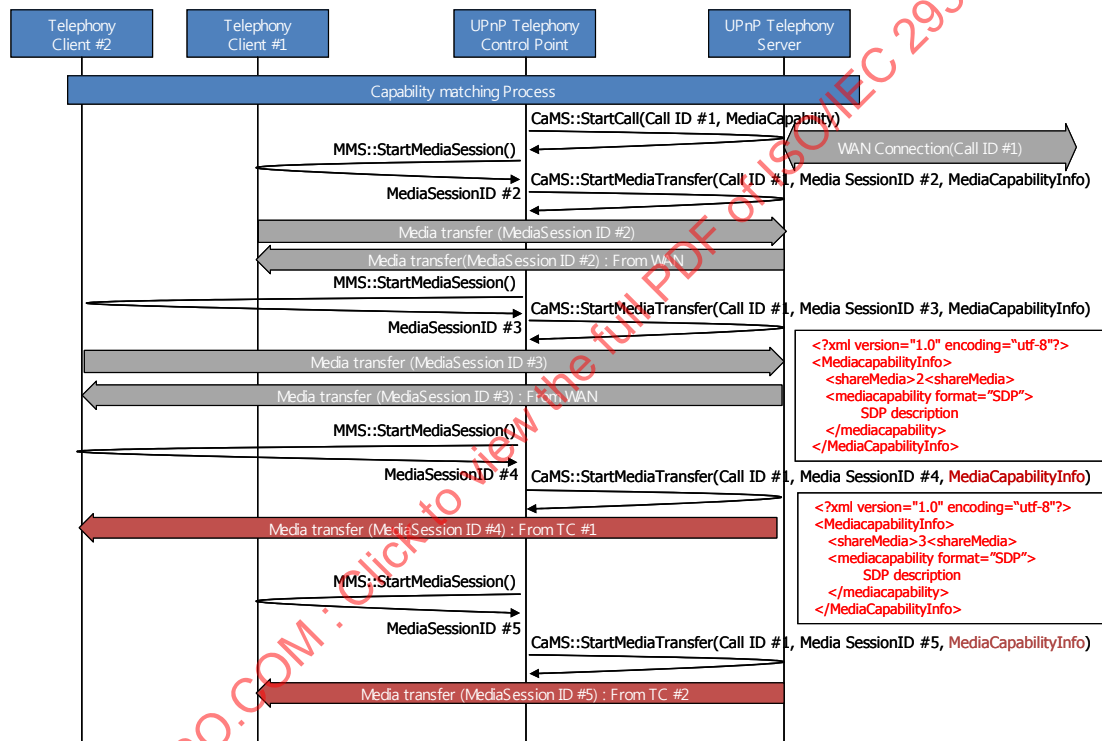


Figure 6 — Starting a Call With Multiple Telephony Clients Using Media Sharing Capability

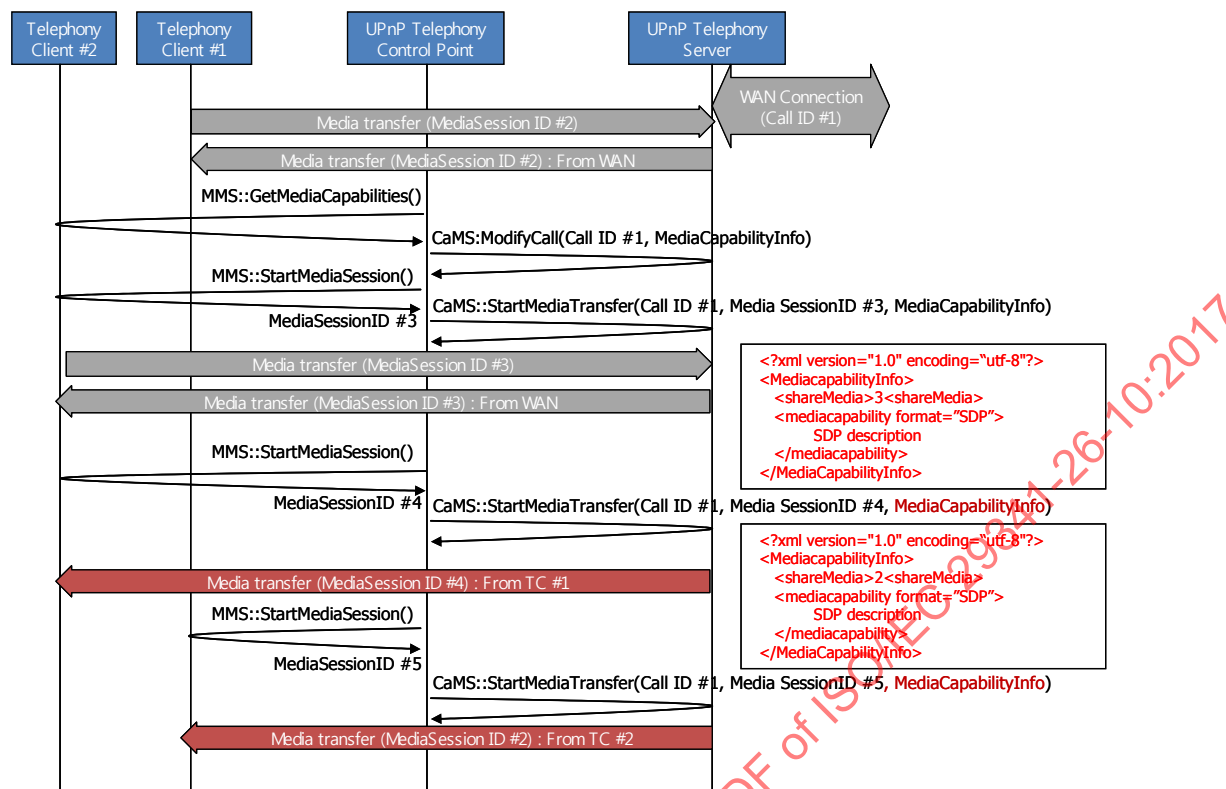


Figure 7 — Modifying a Call With Multiple Telephony Clients Using Media Sharing Capability

5.2.2 User Concept and Multi-User Control

The call management specification provides a mechanism for a Telephony Control Point (TelCP) to register itself with a Telephony Server (TS). The [RegisterTelCPName\(\)](#) action serves this purpose. The action takes the name of the TelCP as an input argument and in response returns a secret key (id for the TelCP). The user concept in this specification can be realized with the same action as used to register a user with the TS. In this instance, the TelCP name will represent the name of the user and the returned secret key which will be the id/secret key for the user. This action will allow a control point to set a unique name for the user. A control point can register multiple users and in each case the TelCP name will represent a different user which will have a different name.

When a TS receives an incoming Call or an SMS or a presence request, the server sends a notification message. The notification message may not include details of the Call and depends on the policy set in the server. The notification message may also contain the name of the user for whom the Call is directed. The notification message can also include the names of TelCP. The “type” attribute of the [targetNames](#) element of the [CallInfo](#) state variable can determine whether the message is directed for a user or for a TelCP.

The TelCP monitors the events from the TS. If a TelCP recognizes a user identifier in the notification message from a previous registration the control point can respond with an action to obtain Call information from the TS. The TelCP where the user is located, invokes the action by providing the key/id of the user. These actions may be optionally sent over a UPnP secure session with the TS to protect inbound Call information or presence information transmitted on the home-network from being intercepted or modified.

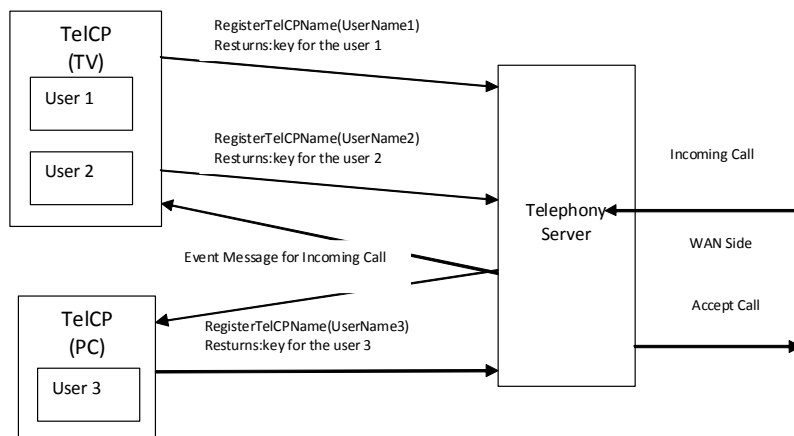


Figure 8 — Example for Multi-User Control Concept

Figure 8 shows a TV with a TelCP. The TelCP is managing two users. The following process explains how the multi-user call management system works:

- The TelCP on the TV registers both users with the Telephony Server. The Telephony Server returns the key for the user for each registration action.
- Each user can log onto the TV using their user name and password. However, the login mechanism is application specific and out of scope of this specification.
- The Telephony Server can be configured locally for different users. The configuration includes setting up which calls will be controlled by which user and by which control point etc.
- When a TelCP receives an event for notification of an incoming Call from the Telephony Server, the TelCP check the notification message. The body of the notification message may contain the name of the users for whom the Call is intended.
- If the Call is for one of the users which the TelCP is managing, the TelCP invokes an action on telephony server to accept or reject the Call.
- When invoking the action to accept/reject, the TelCP also supplies the key associated with the user for whom the Call is intended.
- The Telephony Server will only accept the request from a TelCP that has supplied the valid key and is in the list of authorized TelCP/User to invoke this action.
- The above process is not only valid for managing Call but also for managing presence, voice mail or other telephony services.

5.2.3 Call Monopolization

The telephony architecture enables a Telephony Control Point (TelCP) to manage Calls on the Telephony Server (TS). This includes instructing the TS to start an outgoing Call or answer an incoming Call with a Callee in the wide area network, or instructing the TS to modify or terminate an existing Call with the Callee. In practical deployment scenarios, there can be more than one TelCP in the home network (e.g., multiple TVs and PCs in difference rooms that have TelCP functions). The basic UPnP architecture allows any control point to control UPnP devices. However, in some situations, a user may require important Calls (e.g. private Calls) to be managed only by a limited number of TelCPs. For example:

- a) When a Call on the TS is initiated or accepted by a TelCP which is controlled by a specific user, the user may require this Call to be managed (i.e. modified or terminated) afterwards only by this specific TelCP, or by the user itself by using other TelCPs.
- b) When the TS receives an incoming Call, the user may require that the Call can only be answered by a specific TelCP or a user (i.e., any attempt to answer this Call by unauthorized TelCP, or by an unauthorized users using a TelCP will fail).

Subclause 5.2.2 addresses the requirement of a). The term Call Monopolization refers to a mechanism that allows an identity (a TelCP or a user) to manage a Call. Depending on the type of identity (i.e. TelCP or user), the Call Monopolization can be performed at the “TelCP-level” allowing the Call to be managed by a specific TelCP; or at the “User-level” allowing the Call to be managed only by a specific user using any TelCP.

The CallManagement service is capable of handling the incoming Calls meeting the requirement of b). When receiving an incoming Call, the CallManagement service determines by its internal policies whether any TelCP or user can answer the Call. If not, the service announces via the evented CallInfo state variable which one or more IDs (TelCPNames or usernames) are allowed to answer (accept or reject) the Call. Depending on the type of identity (i.e. a TelCP or a user), the incoming Call can be handled at the “TelCP-level” allowing the incoming Call to be answered by a specific TelCP(s); or at the “User-level” allowing to be answered by a specific user(s) using any TelCP.

There are two mechanisms for Call Monopolization defined in this version of the specification. Subclause 5.2.3.1 defines the mechanism called “PHONE-based Call Monopolization”, targeting only TelCP-level Call Monopolization. When a TS implements the DeviceProtection service, this version of the specification defines another mechanism for Call Monopolization which utilizes the DeviceProtection service and are defined in [15] and named as “DP-based Call Monopolization”. This mechanism enables both TelCP-level and User-level Call Monopolization.

5.2.3.1 PHONE-based Call Monopolization

The PHONE-based Call Monopolization defines a mechanism for a specific TelCP to Monopolize a Call managed by the CallManagement service. The CallManagement service defines the A_ARG_TYPE TelCPName state variable, which is used by this service to identify a TelCP. This service also defines the A_ARG_TYPE SecretKey state variable, which is used by this service to verify the TelCP’s identity. When a new TelCP starts up, it is required to Call the RegisterTelCPName() action defined in this service to register a unique TelCPName for itself. The action returns a Secret Key which is used subsequently by this service to verify the TelCP’s identity when the TelCP tries to Monopolize a Call or manage a Monopolization Mode Call. Figure 9 and text below illustrates how the PHONE-based Call Monopolization mechanism enables a TelCP to Monopolize a Call.

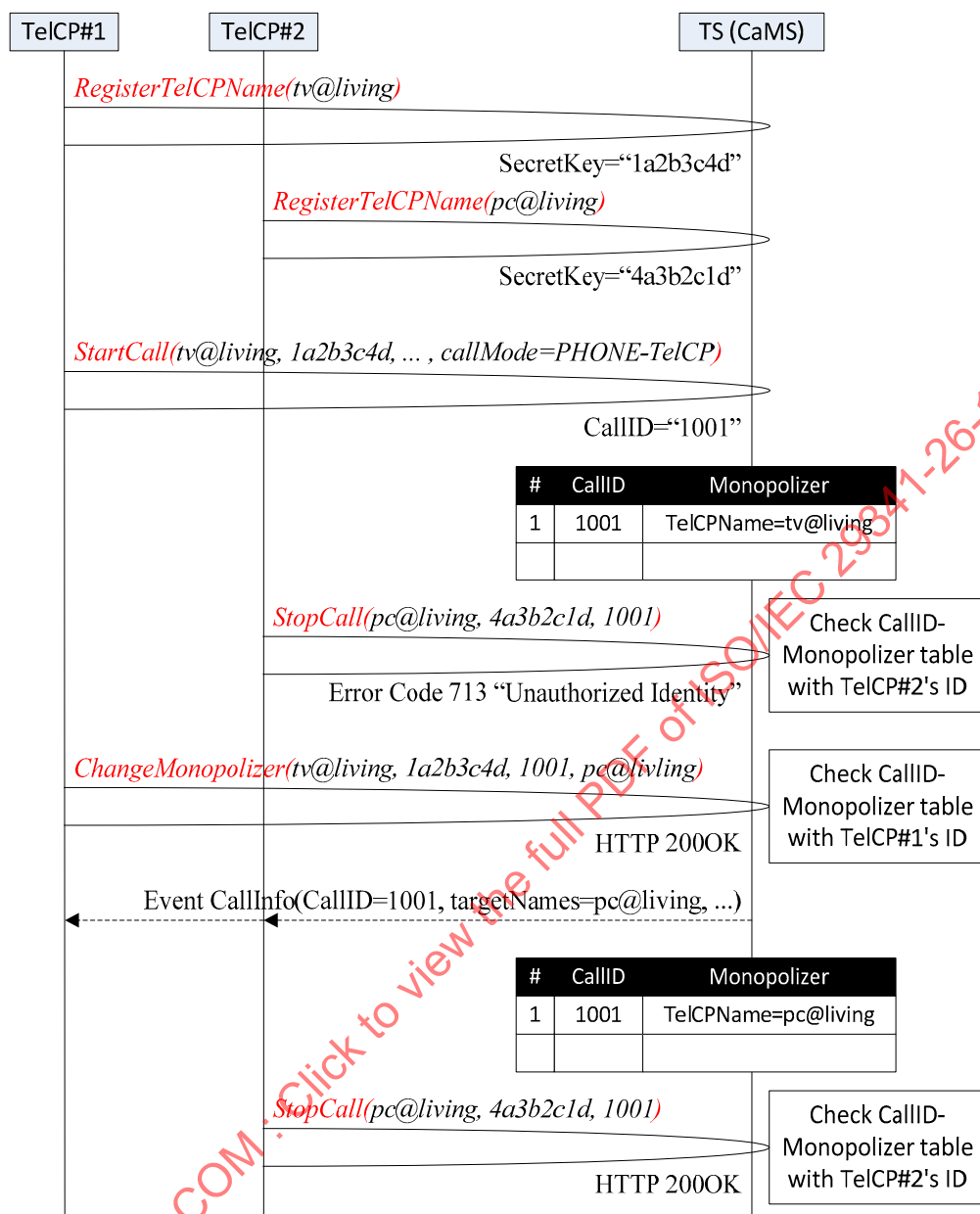


Figure 9 — PHONE-based Call Monopolization

- If a TelCP, e.g., TelCP#1 in Figure 9 wants to Monopolize a Call managed by the CallManagement service, it must first have registered a TelCPName with the service. A new TelCP can invoke the RegisterTelCPName() action to register a unique TelCPName. This action will assign a Secret Key for the TelCP.
- The TelCP#1 can instruct the CallManagement service to initiate a Call by invoking the StartCall() action. In order to Monopolize this specific Call, the TelCP must invoke the action as follows:
 - The input arguments TelCPName and SecretKey of this action must indicate the right values of the TelCP#1's TelCPName and Secret Key, which in this example are "tv@living" and "1a2b3c4d" respectively.
 - The CallMode input argument of this action must also indicate that the Call is to be managed in Monopolized mode by using the "PHONE-TelCP" as its value.
- The CallManagement service verifies the identity of TelCP#1 and if passed, returns the CallID and recognize the TelCP#1 as the current Monopolizer of the Call identified by this

ISO/IEC 29341-26-10:2017(E)

CallID. The CallManagement service internally maps the CallID with Monopolizer's ID (i.e. TelCP#1's TelCPName).

- d) Subsequently, any management action on this Call (e.g., ModifyCall() or StopCall()) must be checked by the CallManagement service to make sure if the action is invoked by the Monopolizer of this Call. For example, in Figure 9,
- TelCP#2 which is not the current Monopolizer of the Call identified by CallID=1001 is not allowed to stop the Call by the CallManagement service (i.e., the StopCall() action will be returned with Error Code 713 "Unauthorized Identity".
- e) The CallManagement service also provides an interface for a Call Monopolizer (i.e. TelCP) to hand over the management rights of the Call to another TelCP.
- By invoking the ChangeMonopolizer() action, the TelCP#1 as the Monopolizer of the Call identified by CallID=1001 hands over the management rights to TelCP#2.
 - The CallManagement service checks TelCP#1's identity and also the CallID by checking internal Monopolizer mapping table, if successful, the table will be updated to indicate TelCP#2 as the current Monopolizer of this Call. The service also events the updated CallInfo state variable indicating the new Monopolizer of this Call.
 - TelCP#2, as the new Monopolizer, is allowed to manage this Call by the CallManagement service, while TelCP#1 is not allowed to manage the Call anymore.

5.2.3.1.1 Incoming Call Handling

When the CallManagement service receives an incoming Call, it can decide by its internal policies whether only a subset of TelCP(s) is allowed to answer the Call. Since the registered TelCPName is used to identify a TelCP, the service announces the allowed identities to answer the incoming Call by listing the relevant TelCPNames as the value of <targetNames> element in the evented CallInfo state variable. Note that the policies itself and the way to configure them are out of the scope of this version of the CallManagement service. When a TelCP invokes the AcceptCall() or RejectCall() action targeting on the Call, the CallManagement service will verify the TelCPName and Secret Key of the TelCP to determine if it is allowed to answer the Call. Figure 10 and text below illustrates how the CallManagement service handles an incoming Call.

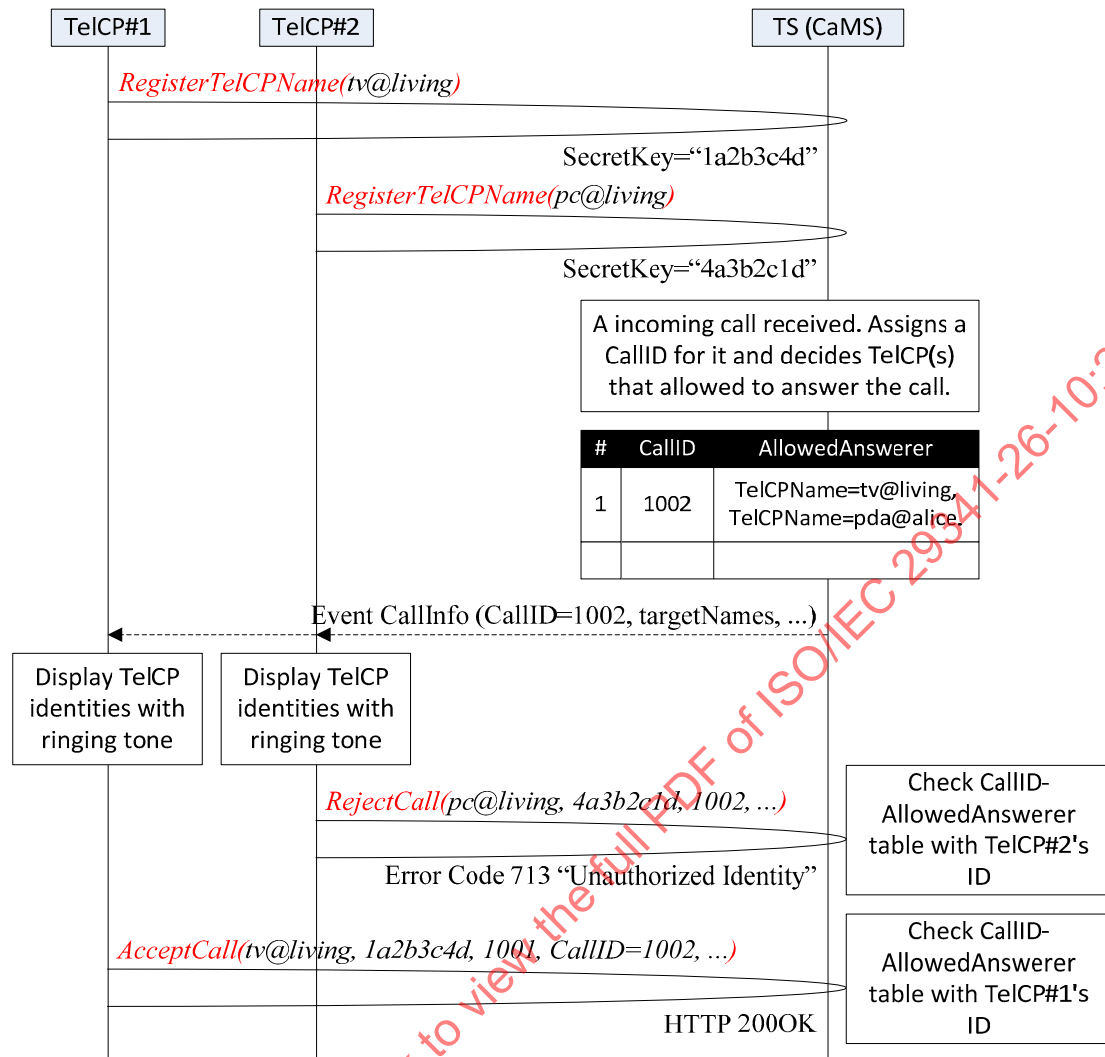


Figure 10 — Incoming Call Handling

- If a TelCP, e.g., TelCP#1 in Figure 10 wants to become a candidate to answer an incoming Call managed by the *CallManagement* service, it must first have a registered TelCPName in the service. A new TelCP can invoke the *RegisterTelCPName()* action to register a unique TelCPName. The action will assign a Secret Key for the TelCP.
- When the *CallManagement* service receives an incoming Call, it can decide whether one or more TelCPs are allowed to answer the Call. The service also assigns a unique CallID for this Call. The service internally maps the CallID with the TelCPName that can manage the Call.
- The *CallManagement* service sends the *CallInfo* state variables of the incoming Call to the TelCPs that had subscribed to the events for this service. The evented *CallInfo* state variable contains the CallID of the Call as well as the TelCPNames of the TelCPs that are allowed to answer the Call.
- When receiving the *AcceptCall()* or *RejectCall()* action from a TelCP the *CallManagement* service verifies the identity of the TelCP by checking its TelCPName and Secret Key from the input arguments of the action. The service also looks up in the "CallID-AllowedAnswerer" table with the TelCPName to decide if the TelCP is allowed to answer/reject the Call. For example, in Figure 10,
 - TelCP#2 with "pc@living" as its TelCPName is not one of the allowed TelCP that can answer the incoming Call, thus the action invoked by this TelCP targeting the Call will fail.

- TelCP#1 has its TelCPName “tv@living” in the evented CallInfo state variable, thus it will succeed in answering the Call by invoking the AcceptCall() action.

5.2.4 Selective Information Delivery

The CallManagement service defined in this specification provides features enabling privacy. For example, when a user starts calling with a Callee in the wide area network, he/she may require some information (e.g. the peer ID) of this Call to be unavailable to the other users in the home network. To meet the above requirement, “Selective Information Delivery” mechanism is used by the CallManagement service.

The Selective Information Delivery defines a mechanism by which the CallManagement service only delivers general information (e.g. Call ID) of a specific Call to the TelCPs, so that privacy related information (e.g. peer ID) are hidden from unauthorized TelCPs or users.

When a Call is managed by the service is Monopolized, or when the service decides that only specific TelCP(s) or User(s) can answer an incoming Call, the Selective Information Delivery mechanism is applied.

- The CallManagement service events the CallInfo state variable to all the subscribed TelCPs containing only the general information of the Call, such as <callID>, <targetNames>, and <callStatus> elements. See 5.3.2 for more detail on which elements of the CallInfo state variable should be and should not be evented.
- When the CallManagement service receives the GetCallInfo() action request targeting the Call, the CallManagement service verifies if the current identity (TelCP or User) is the Monopolizer or allowed to answer the Call. If so, the returned CallInfo state variable contains full information of the Call; otherwise, it contains only general information of the Call, such as <callID>, <TelCPNames>, and <callStatus> elements. See 5.3.2 for more detail on which elements of the CallInfo state variable should be and should not be returned.

5.2.5 Handling of Media Streams

The CallManagement service provides two kinds of mechanisms for handling the Media Streams between the Peers. The first mechanism is “TC-Based Media Handling” and the other mechanism is “TS-Based Media Handling”.

5.2.5.1 TC-Based Media Handling

In the TC-Based Media Handling the TelCP uses one or more TC devices in the ongoing Call. The TelCP creates Media Session(s) between the TS and the TC(s). The Media Streams are sent and/or received between the TC(s) and the Remote Party via the TS using the Media Session(s). For establishing the Media Session for the Call the TelCP negotiates a common set of Media Capabilities between the TS and the TC by matching the Media Capabilities supported by the TS and the TC(s). The TelCP can get the list of supported Media Capabilities of the TS by invoking the CaMS::GetMediaCapabilities() action or by receiving the CallInfo state variable event when the TS receives an incoming Call request or a modify Call request. The Media Capabilities of the TS are described in the <mediaCapability> element of the A_ARG_TYPE MediaCapabilityInfo state variable or the CallInfo state variable which are defined in this document. The TelCP can get the list of supported Media Capabilities of the TC by invoking the MMS::GetMediaCapabilities() action. The Media Capabilities of the TC are described in the <mediaCapability> element of the A_ARG_TYPE MediaCapabilityInfo state variable which is defined in the MediaManagement service [10].

Once the TelCP decides on the Media Capabilities for the Call, the TelCP invokes the StartCall(), AcceptCall(), ModifyCall() or AcceptModifyCall() action to negotiate the Media Capabilities with the Remote Party for establishing the Call or modifying the ongoing Call. When the Call is successfully established or modified, the TelCP invokes the MMS::StartMediaSession() action on the TC and the CaMS::StartMediaTransfer() action on the TS to start the Media Stream transfer between the TS and TC(s). The detail sequence diagrams for the TC-Based Media Stream handling are shown in C.1.

5.2.5.2 TS-Based Media Handling

A TS can be implemented on a physical device like a mobile phone or a HGW. More often physical device itself has the capabilities to handle the Media Streams with certain media codec support. This kind of devices also provide input/output functionality for the media transfer (e.g., a mobile phone has an audio codec, capability to send and/or receive an audio stream, and mic/speaker as the input/output devices to talk with the Remote Party during the Call). The TS-Based Media Handling enables a TelCP to use the TS capabilities of handling the Media Stream in a Call without implementing the TC device functionalities on the physical device which has the TS. For this purpose, the TS exposes the Media Capabilities corresponding to the Media Streams which can be handled by the TS. These Media Capabilities are referred to as the "Native Media Capability" of the TS and are described in the `<nativeMediaCapability>` element of the CallInfo state variable or in the A_ARG_TYPE_MediaCapabilityInfo state variable. A TelCP can get the Native Media Capabilities of the TS by invoking the GetMediaCapabilities() action or receiving the CallInfo state variable when the TS receives an incoming Call or a modify Call request. The TelCP decides on the Native Media Capability for the Call and then the TelCP invokes the StartCall(), AcceptCall(), ModifyCall() or AcceptModifyCall() action to negotiate the Media Capabilities with the Remote Party for establishing the Call or modifying the ongoing Call. When the Call is successfully established or modified, the TelCP invokes CaMS::StartMediaTransfer() action on the TS to start the Media Stream transfer between the TS and the Remote Party. The detail sequence diagrams for the TS-Based Media Stream handling are shown in C.1.

The "TC-Based Media Handling" and "TS-Based Media Handling" can be used simultaneously in a Call. For example, when the TS is implemented on a mobile phone and there is a TC for sending and/or receiving a video stream, the TelCP can create a video Call using the mobile phone as a TS for sending and receiving the audio stream and the TC for sending and/or receiving the video stream. In this case, the TelCP uses "TC-Based Media Handling" for the video capabilities and "TS-Based Media Handling" for the audio capabilities.

5.2.6 Content Sharing

Content Sharing is a feature that enables sharing of content between a TS and the Remote Party during a Call. The users can share audio/video or image content to the Remote Party when talking. The Content Sharing feature utilizes the Media Session between the Peers to transport the content. The content can be sourced from UPnP MediaServer devices, or locally from TC/TS, or from non-UPnP media serving devices. Based on the commands from a TelCP, the selected content is retrieved, processed by the TC or TS, and then shared to the Remote Party by the TS. There are two types of Content Sharing mechanisms that are described in subclauses 5.2.6.1 and 5.2.6.2.

5.2.6.1 TS-Based Content Sharing

The TS-Based Content Sharing mechanism allows a TS to retrieve, process and then send the content to the Remote Party during a Call. The mechanism requires TS to have more media processing capabilities, including retrieving media from media serving devices and transcoding. The TS supporting this mechanism should be able to:

- Retrieve content from the content hosting device, e.g. using HTTP GET method to retrieve the content from a UPnP MediaServer device.
- Process the content before sending to the Remote Party, e.g. transcode to the codec used for the Call, mix the audio content with the speech media.
- Update or use the existing Media Session of the Call to send the content, e.g. renegotiate the Media Capability with the Remote Party to add new channels to transport the A/V content, or send the mixed media in the current channel.

A TelCP supporting this feature shall be able to locate the content (e.g., having AV CP functionalities) and provide the TS with the content information, and then trigger the TS to fetch the content and share to the Remote Party using the actions of this service. The TS supporting this feature shall implement the `<contentSharingCapability>` element defined

ISO/IEC 29341-26-10:2017(E)

in 5.3.12. C.5.1 provides examples on detailed sequence for sharing content during a Call using the TS-Based Content Sharing mechanism.

5.2.6.2 TC-Based Content Sharing

The TC-Based Content Sharing mechanism allows a TC to retrieve, process and then send the content to the Remote Party via a TS during a Call. The mechanism requires a TC to participate in a Call and be responsible for fetching and processing the media content, and sending it to the Remote Party via a TS using the Media Session with the TS. The TC supporting this mechanism should be able to:

- Retrieve content from the content hosting device, e.g. using HTTP GET method to retrieve the content from a UPnP [MediaServer](#) device.
- Process the content before sending to the TS, e.g. transcode to the codec used for the Call, mix the audio content with the speech media.
- Update or use the existing Media Session to send the content to the TS, e.g. add new Media Streams to transport the A/V content, or send the mixed media in the current Media Streams.

A TelCP supporting this feature shall be able to locate the content (e.g., having AV CP functionalities) and provide the TC with the content information, instruct the TS and the TC to update media sessions in the WAN and LAN to add new streams if necessary for sending the content. The TC supporting this feature shall implement the <contentSharingCapability> element defined in the [MediaManagement](#) service in [10]. C.5.2 of this document provides examples on detailed sequence for sharing content during a Call using the TC-Based Content Sharing mechanism.

5.3 State Variables

Note: For first-time reader, it may be more insightful to read the theory of operations first and then the action definitions before reading the state variable definitions.

5.3.1 State Variable Overview

Table 1 — State Variables

Variable Name	R/A ^a	Data Type	Reference
CallInfo	<u>A</u>	string (XML fragment)	See 5.3.2
TelCPNameList	<u>A</u>	string	See 5.3.3
CallBackAvailability	<u>A</u>	string (XML fragment)	See 5.3.4
PushInfo	<u>O</u>	string (XML fragment)	See 5.3.5
VoiceMailInfo	<u>O</u>	string (XML fragment)	See 5.3.6
ParallelCallInfo	<u>O</u>	string (XML fragment)	See 5.3.7
A_ARG_TYPE_TelephonyServerIdentity	<u>R</u>	String	See 5.3.8
A_ARG_TYPE_TelCPName	<u>A</u>	String	See 5.3.9
A_ARG_TYPE_TelCPNameList	<u>A</u>	String	See 5.3.10
A_ARG_TYPE_Expires	<u>A</u>	i4	See 5.3.11
A_ARG_TYPE_MediaCapabilityInfo	<u>A</u>	string (XML fragment)	See 5.3.12
A_ARG_TYPE_CalleeID	<u>R</u>	string	See 5.3.13
A_ARG_TYPE_CallPriority	<u>A</u>	string	See 5.3.14

Variable Name	R/A ^a	Data Type	Reference
<u>A_ARG_TYPE_CallMode</u>	<u>A</u>	<u>string</u>	See 5.3.15
<u>A_ARG_TYPE_CallID</u>	<u>R</u>	<u>string</u>	See 5.3.16
<u>A_ARG_TYPE_SecretKey</u>	<u>A</u>	<u>string</u>	See 5.3.17
<u>A_ARG_TYPE_RejectReason</u>	<u>A</u>	<u>string</u>	See 5.3.18
<u>A_ARG_TYPE_TCList</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.19
<u>A_ARG_TYPE_CallInfoList</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.20
<u>A_ARG_TYPE_CallLogs</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.21
<u>A_ARG_TYPE_CallBackID</u>	<u>A</u>	<u>string</u>	See 5.3.22
<u>A_ARG_TYPE_CallBackInfoList</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.23
<u>A_ARG_TYPE_PushInfoList</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.24
<u>A_ARG_TYPE_VoiceMailInfoList</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.25
<u>A_ARG_TYPE_VoiceMailID</u>	<u>A</u>	<u>string</u>	See 5.3.26
<u>A_ARG_TYPE_CallType</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.27
<u>A_ARG_TYPE_CallerID</u>	<u>A</u>	<u>string</u>	See 5.3.28
<u>A_ARG_TYPE_MaxWaitingTime</u>	<u>A</u>	<u>int</u>	See 5.3.29
^a <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CO</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u>).			

5.3.2 CallInfo

The format of the CallInfo state variable is an XML document. It includes information about a Call. It contains the CallID, the priority of the Call, Remote Party's ID, Call Status, TC's information (i.e., UUID and Media Session ID), and Media Capability etc.

The CallInfo state variable is evented when the CallManagement service receives an incoming Call or the Call Status of an existing Call is changed.

5.3.2.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for CallInfo in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.2.2 Description of fields in the CallInfo structure

```
<?xml version="1.0" encoding="utf-8"?>
<u:cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <callID>unique ID for the Call</callID>
  <targetNames type="type of the targetName">
    unique names or IDs for targets
  </targetNames>
  <callStatus reason="reason why Call Status is changed">
    status of the Call
  </callStatus>
```

```

<priority>priority of the Call</priority>
<remoteParty>
  <peer:id>ID of the Peer</peer:id>
</remoteParty>
<TCList>
  <TC>
    <UDN>uuid:UUID</UDN>
    <mediaSessionID>Unique Media Session ID</mediaSessionID>
  </TC>
  <!-- Any other TC (if any) go here.-->
</TCList>
<enableMediaMixing>1 or 0</enableMediaMixing>
<shareMedia>Media Session ID</shareMedia>
<mediaCapability format="format of the Media Capability">
  Media Capability
</mediaCapability>
<nativeMediaCapability format="format of the Native Media Capability">
  Native Capability
</nativeMediaCapability>
</cams:callInfo>

```

<xml>

Required. Case Sensitive.

<CallInfo>

Required. Shall include the name space declaration for the complex type <peerType> ("urn:schemas-upnp-org:phone:peer") and the namespace declaration for the [CallManagement](#) service Schema ("urn:schemas-upnp-org:phone:cams"). This namespace "urn:schemas-upnp-org:phone:cams" defines the following elements and attributes:

<callID>

Required. xsd:string, shall include the CallID which is unique for a Call.

The [CallManagement](#) service generates the CallID when this service receives an incoming Call or receives [StartCall\(\)](#) or [InitiateCall\(\)](#) action.

<targetNames>

Required. xsd:string, shall include a list of identities (TelCP or user) for Selective Information Delivery (See 5.2.4). The type of identifiers is decided by the attribute "type". The identifiers are listed by comma separated values. When the value of this element is set to "*", all the identifiers are allowed to manage the Call. An identifier shall not include any comma(",") in the string. The allowed values of this attribute are as follows:

Table 2 — Allowed values for type

Value	Description
" TelCPName "	This identity is described by TelCPName which is defined in the CallManagement service.
" TelCPID "	This identity is described by Control Point Identity which is defined in the DeviceProtection service [16].
" Username "	This identity is described by Username which is defined in the DeviceProtection service [16].

<callStatus>

Required. xsd:string, Indicates the status of the Call. The allowed values of this elements are as follows: And this element has an attribute which name is "reason".

Table 3 — Allowed values for callStatus

Value	Description
" Ringing "	The status of the Call when a TS receives an incoming Call. The status of the Call remains as "Ringing" state until a TelCP accepts or rejects the Call.
" Dialing "	The status of the Call when the TS is initiating the Call. The status of the Call remains as "Dialing" state until the Call is received by the Callee in WAN side.

Value	Description
<u>"Calling"</u>	The status of the Call when the Callee receives the Call. The status of the Call remains as "Calling" state until the Call is answered by the Callee.
<u>"Connected"</u>	The status of the Call when the Call is accepted by the Callee. The status of the Call remains as "Connected" state until the Media Streams transfer start.
<u>"Talking"</u>	The status of the Call when the Media Streams transfer starts. The status of the Call remains as "Talking" state until the Call is terminated or modified.
<u>"Disconnected"</u>	The status of the Call when the Call is terminated.
<u>"SendingModifyRequest"</u>	The status of the Call when the TS sends the request to modify the Media Capability of an existing Call (e.g. adding video capability to the audio Call). The status of the Call remains as "SendingModifyRequest" state until the Remote Party answers the modify request.
<u>"ReceivingModifyRequest"</u>	The status of the Call when the TS receives the request to modify the Media Capability of an existing Call from the WAN side (e.g. adding video capability to the audio Call). The status of the Call remains as "ReceivingModifyRequest" state until the TelCP answers the modify request.
<u>"Modified"</u>	The status of the Call when the modify request is accepted by the Caller or the Callee. The status of the Call remains as the "Modified" state until the Media Streams transfer resumes based on the new Media Capabilities.
<u>"ModifyFailed"</u>	The status of the Call when the modify request is failed or rejected. The status of the Call remains as the "ModifyFailed" state until the Media Streams transfer resumes.

reason

Allowed. xsd:string, Indicates the reason why status of a Call is changed. The following table shows the values of reason attribute and relevant Call Status:

Table 4 — Allowed values for reason and the corresponding values of the Call Status

Value	Call Status	Description
<u>"Accepted by TS"</u>	<u>"Talking"</u>	The Callee accepted a Call on the TS.
<u>"Initiated by TS"</u>	<u>"Talking"</u>	The TS itself created an outgoing Call.
<u>"Busy"</u>	<u>"Disconnected"</u>	The Callee is busy.
<u>"Not Acceptable Capability"</u>	<u>"Disconnected"</u>	The Callee does not support the Media Capability.
<u>"Not Found"</u>	<u>"Disconnected"</u>	The Callee does not exist.
<u>"Rejected by TS"</u>	<u>"Disconnected"</u>	The Callee rejects an incoming Call on TS
<u>"Rejected"</u>	<u>"Disconnected"</u>	The Callee rejects an incoming Call.
<u>"Rejected"</u>	<u>"ModifyFailed"</u>	The Caller or the Callee rejects an incoming modify request.
<u>"Cancelled"</u>	<u>"Disconnected"</u>	The Caller cancelled the Call.
<u>"Cancelled"</u>	<u>"ModifyFailed"</u>	The Caller or the Callee cancelled the modify request.
<u>"Receiving Early Media Response"</u>	<u>"Calling"</u>	The TS has received a provisional response for the Early Media session.
<u>"Early Media Started"</u>	<u>"Calling"</u>	The Early Media session is established.
<i>TBD</i>	<i>TBD</i>	<i>(Specified by UPnP vendors.)</i>

<priority>

Allowed. xsd:string, Indicates the priority of the Call. In the case of a Monopolization Mode Call, this element is not included in the CallInfo state variable. The <priority> element can have the following values.

Table 5 — Allowed values for priority

Value	Description
<u>Emergency</u>	Emergency Call.
<u>Normal</u>	Regular Call.
TBD	(Specified by UPnP vendors.)

<remoteParty>

Allowed. peer:peerType, Includes the unique ID of the Remote Party. If the Call is anonymous, the value of the <peer:id> elements set to as the string "anonymous". In the case of a Monopolization Mode Call, this element is not included in the CallInfo state variable.

<TCList>

Allowed. Includes information about one or more TCs. In the case of a Monopolization Mode Call, this element and subelements are not included in the CallInfo state variable. It has following sub elements.

<TC>

Allowed, Includes information about a TC. It has two sub elements: <UDN> and <mediaSessionID>.

<UDN>

Allowed, Indicates the unique device name of a TC.

<mediaSessionID>

Allowed. Indicates the unique ID for a Media Session. See [10] in detail.

<enableMediaMixing>

Allowed. Indicates whether a TS mixes Media Streams from multiple TCs involved in the Call. A value of "1" means that the TS mixes Media Streams from multiple TCs involved in the Call. A value of "0" means that the TS does not mix Media Streams from multiple TCs involved in the Call. In the case of a Non Monopolization Mode Call, if the <enableMediaMixing> element is not included in the CallInfo state variable, then the behavior of this service is the same as the value of the <enableMediaMixing> element is "0". In the case of a Monopolization Mode Call, the <enableMediaMixing> element is not included in the CallInfo state variable. But the output argument of the GetCallInfo() action can include this element if the Monopolizer invokes the GetCallInfo() action.

<shareMedia>

Allowed. Includes the Media Session ID to identify the Media Session to be shared among TCs involved in the Call. In the case of a Non-Monopolization Mode Call, if the <shareMedia> element is not included in the CallInfo state variable, it means that a Media Session is not shared among TCs. In the case of a Monopolization Mode Call, the <shareMedia> element is not included in the CallInfo state variable when the TS events the CallInfo state variable. But if the Monopolizer invokes the GetCallInfo() action, this element is included. The <enableMediaMixing> element and the <shareMedia> element are mutually exclusive and if one element exists in the CallInfo state variable then the other shall not exist.

<mediaCapability>

Allowed. This element is used to represent the Media Capabilities of an incoming Call or Media Capabilities for the Call which are handled using the TC-Based Media Handling mechanism.

format

Allowed. xsd:string, Indicates the format of Media Capabilities. Allowed values of this attribute are as follows:

Table 6 — Allowed values for format

Value	Description
SDP	Media Capability as defined by [7].
TBD	(Specified by UPnP vendors.)

<nativeMediaCapability>

Allowed. This element is used to represent the TS's Native Media Capabilities or Media Capabilities of the Call which are handled using the TS-Based Media Handling mechanism.

format

Allowed. xsd:string, Indicates the format of Media Capabilities for this element. Allowed values of this attribute are shown in Table 6.

5.3.3 **TelCPNameList**

The **TelCPNameList** state variable includes the names of all the existing TelCPs. The format of the **TelCPNameList** state variable is a comma separated values. It is the same as the format of the **A_ARG_TYPE_TelCPNameList**. When a new TelCP is registered or a TelCP is unregistered or the name of a TelCP is changed, then the **TelCPNameList** state variable is evented.

5.3.4 **CallbackAvailability**

This state variable contains the **A_ARG_TYPE_CallbackID** state variable. This state variable is evented when a Call that is registered for a Call back becomes available. When a Call is registered for a Call back, the TS returns the **CallbackID** ID. Once the registered Call becomes available from the WAN side, the TS notifies the availability of the Call using this state variable.

5.3.4.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for **CallbackAvailability** in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.4.2 Description of fields in the **CallbackAvailability** structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:callbackAvailability
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
  http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <callback>
    <callbackID>ID of the Call back</callbackID>
  </callback>
  <!-- Any other Call back (if any) go here.-->
</cams:callbackAvailability>
```

<xml>

Required for all XML documents. Case Sensitive.

<CallbackAvailability>

Required. Root element.

<Callback>

Required. Contains the list of Call backs that are available and uniquely identified by the CallbackIDs. It includes the CallbackID sub element.

<CallbackID>

Required, Indicates the ID of the Call back.

5.3.5 **PushInfo**

The format of the **PushInfo** state variable is an XML document. It includes information that a TS received from a service provider on WAN side or some status changes that occurs within a TS.

ISO/IEC 29341-26-10:2017(E)

5.3.5.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for PushInfo in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.5.2 Description of fields in the PushInfo structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:pushInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <priority>priority of information</priority>
  <summary>summary of information</summary>
  <url>URL to get detail information</url>
  <import format="format">
    imported information goes here
  </import>
</cams:pushInfo>
```

<xml>

Required for all XML documents. Case Sensitive.

<pushInfo>

Required. Root element.

<priority>

Required. xsd:string, Indicates the priority of the event. Table 7 shows the allowed values of the <priority> element.

Table 7 — Allowed values for priority

Value	Description
" <u>High</u> "	Indicates high priority event.
" <u>Normal</u> "	Indicates normal priority event.
" <u>Low</u> "	Indicates low priority event.
TBD	(Specified by UPnP vendors.)

<summary>

Required. xsd:string, Indicates the summary of the event.

<url>

Required. xsd:url, Indicates the URL to get detail information.

<import>

Allowed. xsd:string, includes information from the other source. The format of this information is represented by the "format" attribute.

5.3.6 VoiceMailInfo

This state variable is evented by the Telephony Server (TS) when a new voice mail becomes available.

5.3.6.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for voiceMailInfo in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.6.2 Description of fields in the **VoiceMailInfo** structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:voiceMailInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <voiceMail>
    <voiceMailID>ID of the voice mail</voiceMailID>
    <callee>the callee who left the voice mail</callee>
    <url>URL to retrieve the voice mail</url>
    <duration> duration of the voice mail</duration>
    <time>date, time of the voicemail when left</time>
    <!-- Any other envelope information (if any) go here.-->
  </voiceMail>
</cams:voiceMailInfo>

<xml>
```

Required for all XML documents. Case Sensitive.

<voicemailInfo>

Required. Root element.

<voiceMail>

Required. Contains the following sub elements.

<voiceMailID>

Required. xsd:string, Indicates the ID of the voice mail.

<callee>

Allowed. xsd:string, Indicates the tel uri of the callee who left the voice mail.

<url>

Allowed. xsd:url, Indicates the url to retrieve the voice mail.

<duration>

Allowed. xsd:string, Indicates the duration of the voice mail.

<time>

Allowed. xsd:string, Indicates the time of the voice mail when left in ISO 8601 format.

5.3.7 **ParallelCallInfo**

The format of the **ParallelCallInfo** state variable is an XML document. It includes configuration information of the remote device.

5.3.7.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for **ParallelCallInfo** in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.7.2 Description of fields in the **ParallelCallInfo** structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:parallelCallInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <informationType>
    status information of remote device
  </informationType>
```

```

<remoteParty>
  <peer:id>ID of the Peer</peer:id>
</remoteParty>
</cams:parallelCallInfo>

```

<xml>

Required. Case Sensitive.

<parallelCallInfo>

Required. Root element.

<informationType>

Required. xsd:string, Indicates the type of configuration information of a remote TS.

Table 8 — Allowed values for informationType

Value	Description
" ParallelCallRequest "	The TS has received a request for Parallel Call from a remote device.
" ParallelCallAccepted "	The TS has received a response that a remote TS accepted a Parallel Call request.
" ParallelCallDenied "	The TS has received a response that a remote TS denied a Parallel Call request.
TBD	(Specified by UPnP vendors.)

<remoteParty>

Allowed. peer:peerType, Includes an identity of a remote device which will receive the Parallel Call. If the value of the <informationType> element is "[ParallelCallAccepted](#)", then this element shall contain the valid value.

5.3.8 [A_ARG_TYPE_TelephonyServerIdentity](#)

This state variable contains the unique identifier of a Telephony Server. The identity of a Telephony Server is expressed using the standard URI (Unified Resource Identifier) scheme as specified in [3]. In case of SIP, the identity of the Telephony Server identity contains the SIP URI. In case of a generic resource identified by a telephone number, the Telephony Server identity contains the TEL URI [6].

5.3.9 [A_ARG_TYPE_TelCPName](#)

The [A_ARG_TYPE_TelCPName](#) state variable describes a unique name for a TelCP. Each TelCP has its own name. A TelCP can register its name to the [CallManagement](#) service by invoking the [RegisterTelCPName\(\)](#) action. And a TelCP also can unregister and change its TelCPName by invoking the [RegisterTelCPName\(\)](#) action. It is recommended that this state variable includes human readable and understandable string. (e.g. TV@Living, Alice's TV etc.)

5.3.10 [A_ARG_TYPE_TelCPNameList](#)

The [A_ARG_TYPE_TelCPNameList](#) state variable includes the names of all the TelCPs. The format of the [A_ARG_TYPE_TelCPNameList](#) is comma separated values of sting. The structure of this state variable is the same as the format of the [TelCPNameList](#) state variable.

5.3.11 [A_ARG_TYPE_Expires](#)

The [A_ARG_TYPE_Expires](#) state variable contains the value of the duration for validity of the TelCPName. The value of this state variable is an integer number in seconds.

5.3.12 [A_ARG_TYPE_MediaCapabilityInfo](#)

This state variable defines an XML document that contains the Media Capability of a Media Session.

5.3.12.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for [MediaCapabilityInfo](#) in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.12.2 Description of fields in the [MediaCapabilityInfo](#) structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <mediaMixingCapability></mediaMixingCapability>
  <mediaSharingCapability></mediaSharingCapability>
  <enableMediaMixing> 1 or 0 </enableMediaMixing>
  <shareMedia>Media Session ID</shareMedia>
  <mediaCapability format="format of the Media Capability">
    Media capability
  </mediaCapability>
  <nativeMediaCapability format="format of the Native Media Capability">
    Media capability
  </nativeMediaCapability>
  <contentSharingCapability>
    <protocolInfo>metadata information of the content</protocolInfo>
    <!-- other protocolInfo goes here. -->
    <URI>URI of the content</URI>
    <sharingType>transfer or streaming</sharingType>
  </contentSharingCapability>
</cams:mediaCapabilityInfo>
```

<xml>

Required. Case Sensitive.

<mediaCapabilityInfo>

Required. Shall include a namespace declaration for the [CallManagement](#) service Schema ("urn:schemas-upnp-org:phone:cams"). This namespace defines the following elements and attributes:

<mediaMixingCapability>

Allowed. The existence of this tag indicates whether a TS supports media mixing ability. This tag can only exist in the output argument [SupportedMediaCapabilityInfo](#) of the [GetMediaCapabilities\(\)](#) action. This tag shall not exist in any other input or output arguments of the type [MediaCapabilityInfo](#) used by any other actions.

<mediaSharingCapability>

Allowed. The existence of this tag indicates whether a telephony client support the media sharing feature. This tag can only exist in the output argument [MediaCapabilityInfo](#) of the [GetMediaCapabilities\(\)](#) action. This tag shall not exist in any other input or output arguments of type [A_ARG_TYPE_MediaCapabilityInfo](#) state variable used by any other actions.

<enableMediaMixing>

Allowed. Indicates whether a TS needs to mix media from multiple TCs involved in the Call. This tag can only exist in the argument of type [MediaCapabilityInfo](#) of the [StartMediaTransfer\(\)](#) and [ModifyCall\(\)](#) actions. This tag shall not exist in any other input or output arguments of type [A_ARG_TYPE_MediaCapabilityInfo](#) used by any other actions. A value of "1" means that the TS needs to mix media from multiple Telephony Clients involved in the Call. A value of "0" means that the Telephony Server shall stop mixing media from multiple Telephony Clients involved in the Call.

<shareMedia>

Allowed. Indicates to the TS that the session is for sharing the media of the Media Session identified by the value of the <shareMedia> element. This tag is only valid when used in the argument of type [MediaCapabilityInfo](#) of the [StartMediaTransfer\(\)](#) or [ModifyCall\(\)](#) actions and when creating Media Sessions for sharing media. Please refer to 5.2.1 for the appropriate usage of this tag.

<mediaCapability>

Allowed. Indicates the Media Capabilities for the Call which are handled using TC-Based Media Handling mechanism.

format

Allowed. xsd:string, Indicates the format of Media Capability. Allowed values of this attribute are shown in Table 6.

<nativeMediaCapability>

Allowed. Indicates the Media Capabilities for the Call which are handled using TS-Based Media Handling mechanism.

format

Allowed. xsd:string, Indicates the format of Media Capabilities for this element. Allowed values of this attribute are shown in Table 6.

<contentSharingCapability>

Allowed. Used by a TS to indicate whether it supports the TS-Based Content Sharing mechanism defined in 5.2.6.1 by the SupportedMediaCapabilityInfo output argument of the GetMediaCapabilities() action. And used by a TelCP to set up a Content Sharing session using the StartCall() or the ModifyCall() action, and trigger the Content Sharing process using the StartMediaTransfer() action.

<protocolInfo>

Allowed. xsd:string. One or more instances of this element are used by the TS to indicate its Media Capabilities (e.g., transport protocols, codecs etc.) for the TS-Based Content Sharing mechanism. A TelCP uses this element to provide the metadata information of the content for the TS to share to the Remote Party. The format of this string element follows the UPnP AV protocolInfo concept as defined in [12].

<URI>

Allowed. xsd:anyURI. used by a TelCP to provide the URI of the content to be shared, used by the TS to locate and retrieve the content.

<sharingType>

Allowed. xsd:string. used by a TelCP to indicate to the TS if the content should be sent as a file or streamed to the Remote Party when invoking the StartCall(), ModifyCall() or StartMediaTransfer() action. The allowed value for this element is "transfer", or "streaming". This element is only a hint to the TS, the TS implementation could anyway choose its supported mechanism.

5.3.13 A ARG TYPE CalleeID

This state variable contains the unique identifier of a Callee. The format of this state variable is URI or A ARG TYPE TelCPName. URI is used when a TelCP creates a Call to a Remote Party in the WAN side. A ARG TYPE TelCPName is used when a TelCP creates a Call to a Remote Party in the same LAN side.

5.3.14 A ARG TYPE CallPriority

This state variable contains the priority of the Call. The allowed values of this state variable are shown in Table 9:

Table 9 — allowedValueList for the A ARG TYPE CallPriority state variable

Value	R/A ^a
" <u>Normal</u> " (DEFAULT)	<u>R</u>
" <u>Emergency</u> "	<u>A</u>
^a <u>R</u> = required, <u>A</u> = allowed, <u>X</u> = Non-standard.	

where:

- "Normal" indicates a regular Call.
- "Emergency" indicates an emergency Call (e.g., Call to the police, Call to the fire station etc.).

5.3.15 A ARG TYPE CallMode

This state variable contains one of the modes for a Call. The allowed values of this state variable are shown in Table 10:

Table 10 — allowedValueList for the A ARG TYPE CallMode state variable

Value	R/A ^a
" <u>PHONE-TelCP</u> "	<u>A</u>
" <u>DP-TelCP</u> "	<u>A</u>
" <u>DP-User</u> "	<u>A</u>
" <u>Non-Monopolize</u> "	<u>R</u>
^a <u>R</u> = required, <u>A</u> = allowed, <u>X</u> = Non-standard.	

where:

- "PHONE-TelCP" indicates PHONE-Based Call Monopolization mechanism defined in 5.2.2.
- "DP-TelCP" indicates TelCP-Level Call Monopolization mechanism using the DeviceProtection service. See [15].
- "DP-User" indicates User-Level Call Monopolization mechanism using the DeviceProtection service. See [15].
- "Non-Monopolize" indicates a Non-Monopolization Mode Call.

5.3.16 A ARG TYPE CallID

This state variable uniquely identifies the Call. The CallManagement service assigns a unique ID for a Call when either the StartCall() or InitiateCall() action is invoked or it receives an incoming Call. It is recommended to follow [4] for the format of the CallID.

5.3.17 A ARG TYPE SecretKey

This state variable contains a unique key that is assigned by a TS to the TelCP. The TelCP uses this key to authenticate itself to the TS in order to manage a Monopolization Mode Call. This key is assigned when the TelCP invokes the RegisterTelCPName() action.

5.3.18 A ARG TYPE RejectReason

This state variable contains the reason for rejecting an incoming Call or a modification request for a Call.

5.3.19 A ARG TYPE TCList

This state variable includes a list of TCs information associated with a Call. The list includes the UDN and the Media Session ID for each TC.

5.3.19.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for TCList in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.19.2 Description of fields in the TCList structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:TCList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <TC>
    <UDN>uuid:UUID</UDN>
    <mediaSessionID>Unique Media Session ID</mediaSessionID>
```

```
</TC>
<!-- Any other TC info (if any) go here.-->
</cams:TCList>
```

<xml>

Required. Case Sensitive.

<TCList>

Required. Shall include the namespace declaration for the [CallManagement](#) service Schema ("urn:schemas-upnp-org:phone:cams"). This namespace "urn:schemas-upnp-org:phone:cams" defines the following elements to includes information about one or more TCs:

<TC>

Allowed. Includes information about a TC. It has two sub elements: <UDN> and <mediaSessionID>.

<UDN>

Allowed. Indicates a unique device name of a TC.

<mediaSessionID>

Required. Indicates a unique ID for a Media Session. See [10] in detail.

5.3.20 **A ARG TYPE CallInfoList**

This state variable contains a list of ongoing Calls and their information as defined in the [CallInfo](#) structure.

5.3.20.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for [CallInfoList](#) in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.20.2 Description of fields in the [CallInfoList](#) structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:callInfoList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <callInfo>
    <callID>unique ID for the Call</callID>
    <targetNames type="type of the targetName">
      unique names or IDs for targets
    </targetNames>
    <callStatus reason="reason why Call Status is changed">
      status of the Call
    </callStatus>
    <priority>priority of the Call</priority>
    <remoteParty>
      <peer:id>ID of the Peer<peer:id>
    </remoteParty>
    <TCList>
      <TC>
        <UDN>uuid:UUID</UDN>
        <mediaSessionID>Unique Media Session ID</mediaSessionID>
      </TC>
      <!-- Any other TC (if any) go here.-->
    </TCList>
    <mediaCapability format="format of Media Capability">
      Media capability
    </mediaCapability>
  </callInfo>
</callInfoList>
```

```

    <TelCPNames>unique names for the TelCPs</TelCPNames>
    <callStatus reason="reason why status is changed">
        status of the Call
    </callStatus>
</callInfo>
<!-- Any other callInfo (if any) go here.-->
</cams:callInfoList>

```

<xml>

Required. Case Sensitive.

<callInfoList>

Required. Shall include the name space declaration for the complex type <peerType> ("urn:schemas-upnp-org:phone:peer") and the namespace declaration for the [CallManagement](#) service Schema ("urn:schemas-upnp-org:phone:cams"). This element includes one or more <callInfo> elements as described in the [CallInfo](#) structure.

5.3.21 [A ARG TYPE CallLogs](#)

This state variable contains the history of the Calls managed by a TS.

5.3.21.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for [CallLogs](#) in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.21.2 Description of fields in the [CallLogs](#) structure

```

<?xml version="1.0" encoding="utf-8"?>
<cams:callLogs
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <call direction=incoming|outgoing success=true|false>
    <type> type of the Call whether audio, video or data</type>
    <caller>
      <peer:id>id of the Caller</peer:id>
      <peer:name>user friendly name associated with id</peer:name>
    </caller>
    <callee>
      <peer:id>id of the Caller</peer:id>
      <peer:name>user friendly name associated with id</peer:name>
    </callee>
    <startDateTime>time in ISO 8601 format</startDateTime>
    <duration>Call duration in ms</duration>
  </call>
  <!-- Any other Call History (if any) go here.-->
</cams:callLogs>

```

<xml>

Required. Case Sensitive.

<callLogs>

Required. Shall include the name space declaration for the complex type <peerType> ("urn:schemas-upnp-org:phone:peer") and the namespace declaration for the [CallManagement](#) service Schema ("urn:schemas-upnp-org:phone:cams"). This namespace "urn:schemas-upnp-org:phone:cams" defines the following elements and attributes:

<call>

Allowed. Contains information relevant to a Call managed by a TS. The [direction](#) attribute identifies whether it is an incoming or outgoing Call and the [success](#) attribute identifies whether the Call was successful or failed. It contains the following sub elements:

<type>

Required. Identifies the type of the Call such as audio video or data Call.

<caller>

Required. peer:peerType, Contains information related to a Caller. It contains the following sub elements:

<peer:id>

Required. Uniquely identifies a Caller (i.e., telephone number or SIP address etc.)

<peer:name>

Allowed. Contains a user friendly name for the Caller.

<callee>

Required. peer:peerType, Contains information related to a Callee. Contains the following sub elements:

<peer:id>

Required. Uniquely identifies a Callee (i.e., telephone number or SIP address etc.)

<peer:name>

Allowed. Contains a user friendly name for the Callee.

<startTime>

Required. Indicates the starting time of the Call. The format of the element is in ISO 8601 format.

<duration>

Allowed. Indicates the entire duration of the Call in milliseconds.

5.3.22 A ARG TYPE CallBackID

This state variable contains the unique identifier of a Call back registration. A TS generates this unique identifier as a result of the Call back registration request from a TelCP. The TS uses this unique identifier in the event notification message to indicate the status of the Call back registration which includes the availability of the Callee.

5.3.23 A ARG TYPE CallBackInfoList

This state variable contains the list of Call backs that are currently registered with the TS. The structure of the A ARG TYPE CallBackInfoList state variable is a XML Document:

- <callbackinfo> is the root element.

Note that since the value of A ARG TYPE CallBackInfoList is XML, it needs to be escaped (using the normal XML rules: [8] 2.4) before embedding in a SOAP response message.

5.3.23.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for CallBackInfoList in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.23.2 Description of fields in the CallBackInfoList structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:callbackInfoList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <callbackInfo>
    <callbackID>identifies a Call back registration</callbackID>
    <callee>
```

```

    <peer:id>id of the Callee</peer:id>
  </callee>
</callBackInfo>
</cams:callBackInfoList>

```

<xml>

Required. Case Sensitive.

<callBackInfoList>

Required. Shall include the name space declaration for the complex type <peerType> ("urn:schemas-upnp-org:phone:peer") and the namespace declaration for the [CallManagement](#) service Schema ("urn:schemas-upnp-org:phone:cams"). This namespace "urn:schemas-upnp-org:phone:cams" defines the following elements and attributes:

<callBackInfo>

Allowed. Includes a specific entry for the Call back registration including the ID of the Call back that uniquely identifies a specific Call back registration and the ID of the Callee.

<callBackID>

Required. Uniquely identifies a Call back registration within a TS.

<callee>

Required. peer:peerType, Includes a unique identification of the Callee such as phone number or SIP address etc. Contains the following sub elements:

<peer:id>

Required. Uniquely identifies a Callee (i.e., telephone number or SIP address etc.).

5.3.24 [A ARG TYPE PushInfoList](#)

The format of the [PushInfoList](#) state variable is an XML document. It includes a list of [PushInfo](#).

5.3.24.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for [PushInfoList](#) in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.24.2 Description of fields in the [PushInfoList](#) structure

```

<?xml version="1.0" encoding="utf-8"?>
<cams:pushInfoList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <pushInfo>
    <priority>priority of information</priority>
    <summary>summary of information</summary>
    <url>URL to get detail information</url>
    <import format="format">
      imported information goes here
    </import>
  </pushInfo>
  <!-- Any other pushInfo (if any) go here.-->
</cams:pushInfoList>

```

<xml>

Required for all XML documents. Case Sensitive.

<pushInfoList>

Required. Shall include the namespace declaration for the [CallManagement](#) service Schema ("urn:schemas-upnp-org:phone:cams"). This element includes zero or more <pushInfo> elements as described in the [pushInfo](#) structure.

5.3.25 A ARG TYPE VoiceMailInfoList

This state variable contains the A ARG TYPE VoiceMailInfoList state variable. This state contains a list of voicemails available in the Telephony Server (TS).

5.3.25.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for VoiceMailInfoList in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.25.2 Description of fields in the VoiceMailInfoList structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:voiceMailInfoList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <voiceMail>
    <voiceMailID>ID of the voice mail</voiceMailID>
    <callee>the callee who left the voice mail</callee>
    <url>URL to retrieve the voice mail</url>
    <duration> duration of the voice mail</duration>
    <time>date, time of the voicemail when left</time>
    <!-- Any other envelope information (if any) go here.-->
  </voiceMail>
  <!-- Any other voice mail (if any) go here.-->
</cams:voiceMailInfoList>
```

<xml>

Required for all XML documents. Case Sensitive.

<voiceMailInfoList>

Required. Root element.

<voiceMail>

Required. Contains the list of voice mails that are available. Each of the voice mail is uniquely identified by the VoiceMailID.

<voiceMailID>

Required. xsd:string, Indicates the ID of the voice mail.

<callee>

Allowed. xsd:string, Indicates the tel uri of the callee who left the voice mail.

<url>

Allowed. xsd:string, Indicates the url to retrieve the voice mail.

<duration>

Allowed. xsd:string, Indicates the duration of the voice mail.

<time>

Allowed. xsd:string, Indicates the time of the voice mail when left in ISO 8601 format.

5.3.26 A ARG TYPE VoiceMailID

This state variable contains the unique identifier of a voice mail which is unique within a TS.

5.3.27 A ARG TYPE CallType

The A ARG TYPE CallType state variable includes the type of a Call.

5.3.27.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for CallType in the XML namespace "urn:schemas-upnp-org:phone:cams" which is located at "http://www.upnp.org/schemas/phone/cams-v2.xsd".

5.3.27.2 Description of fields in the CallType structure

```
<?xml version="1.0" encoding="utf-8"?>
<cams:callType
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
  http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <video>
    <videoQuality>Quality of video</videoQuality>
  </video>
  <audio>
    <audioQuality>Quality of audio</audioQuality>
  </audio>
</cams:callType>
```

<xml>

Required for all XML documents. Case Sensitive.

<callType>

Required. Root element.

<video>

Allowed. Indicates whether a Call initiated by aTelCP includes a video stream.

<videoQuality>

Allowed. Indicates quality of a video stream of a Call.

Table 11 — Allowed values for videoQuality

Value	Description
" <u>HD</u> "	High definition video quality
" <u>SD</u> "	Standard definition video quality
TBD	(Specified by UPnP vendors.)

<audio>

Allowed. Indicates whether a Call initiated by aTelCP includes an audio stream.

<audioQuality>

Allowed. Indicates quality of a audio stream of a Call.

Table 12 — Allowed values for audioQuality

Value	Description
" <u>SWB</u> "	Super wide-band audio quality
" <u>WB</u> "	Wide-band audio quality
" <u>NB</u> "	Narrow-band audio quality
TBD	(Specified by UPnP vendors.)

5.3.28 A ARG TYPE CallerID

The A ARG TYPE CallerID state variable represents the identity of a Caller.

5.3.29 A_ARG_TYPE_MaxWaitingTime

This A_ARG_TYPE_MaxWaitingTime state variable includes maximum duration in seconds for waiting time.

5.4 Eventing and Moderation**Table 13 — Event Moderation**

Variable Name	Evented	Moderated Event	Max Event Rate ^a (seconds)	Logical Combination	Min Delta per Event ^b
<u>CallInfo</u>	<u>YES</u>	<u>NO</u>			
<u>TelCPNameList</u>	<u>YES</u>	<u>NO</u>			
<u>CallBackAvailability</u>	<u>YES</u>	<u>NO</u>			
<u>PushInfo</u>	<u>YES</u>	<u>NO</u>			
<u>VoiceMailInfo</u>	<u>YES</u>	<u>NO</u>			
<u>ParallelCallInfo</u>	<u>YES</u>	<u>NO</u>			
<u>A_ARG_TYPE_TelephonyServerIdentity</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_TelCPName</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_TelCPNameList</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_Expires</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_MediaCapabilityInfo</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CalleeID</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallPriority</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallMode</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallID</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_SecretKey</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_RejectReason</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_TCLList</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallInfoList</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallLogs</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallBackID</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallBackInfoList</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_PushInfoList</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_VoiceMailInfoList</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_VoiceMailID</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallType</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_CallerID</u>	<u>NO</u>	<u>NO</u>			
<u>A_ARG_TYPE_MaxWaitingTime</u>	<u>NO</u>	<u>NO</u>			
Non-standard state variables implemented by a UPnP vendor go here	<u>NO</u>	<u>NO</u>			
^a Determined by N, where Rate = (Event)/(N secs). ^b (N) * (allowedValueRange Step).					

5.4.1 Eventing of CallInfo

This state variable is evented when the status of the Call changes or the CallManagement service receives an incoming Call.

When this state variable is evented the <callID> element shall include the CallID for the Call whose status has changed or in the case of an incoming Call, this element includes the CallID of the incoming Call.

5.4.2 Eventing of TelCPNameList

This state variable is evented when a new TelCPName is registered with the TS, or the TelCPName is unregistered or the registered TelCPName is changed.

When a TelCP receives an event message with this state variable, the TelCP shall update the list of TelCPNames which is being managed by the TelCP.

5.4.3 Eventing of CallBackAvailability

This state variable is evented when the Callee of a registered Call back becomes available to receive the Call.

5.4.4 Eventing of PushInfo

This state variable is evented when a TS receives information from a service provider on the WAN side. This state variable is also evented when there is a change in status in the TS.

5.4.5 Eventing of VoiceMailInfo

This state variable is evented when a TS receives a new voice mail.

5.4.6 Eventing of ParallelCallInfo

This state variable is evented when a TS receives a request and/or response for a Parallel Call.

5.5 Actions

Table 14 – Actions

Name	Device R/A ^a	Control Point R/A ^b
<u>GetTelephonyIdentity()</u>	R	R
<u>RegisterTelCPName()</u>	A	A
<u>UnregisterTelCPName()</u>	A	A
<u>ChangeTelCPName()</u>	A	A
<u>GetTelCPNameList()</u>	A	A
<u>GetMediaCapabilities()</u>	A	A
<u>StartCall()</u>	A	A
<u>StopCall()</u>	A	A
<u>AcceptCall()</u>	A	A
<u>RejectCall()</u>	A	A
<u>ModifyCall()</u>	A	A
<u>AcceptModifyCall()</u>	A	A
<u>StartMediaTransfer()</u>	A	A
<u>ChangeMonopolizer()</u>	A	A
<u>InitiateCall()</u>	R	R
<u>GetCallInfo()</u>	A	A
<u>GetCallLogs()</u>	A	A
<u>ClearCallLogs()</u>	A	A
<u>RegisterCallBack()</u>	A	A

Name	Device R/A ^a	Control Point R/A ^b
<u>ClearCallBack()</u>	A	A
<u>GetCallBackInfo()</u>	A	A
<u>ChangeCallMode()</u>	A	A
<u>GetPushInfo()</u>	A	A
<u>IgnoreCall()</u>	A	A
<u>GetVoiceMail()</u>	A	A
<u>DeleteVoiceMail()</u>	A	A
<u>EnhancedInitiateCall()</u>	A	A
<u>WaitingForCall()</u>	A	A
<u>InitiateParallelCall()</u>	A	A
<u>AcceptParallelCall()</u>	A	A
^a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CO</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u>, <u>A-D</u>). ^b For a control point this column indicates whether a control point shall be capable of invoking this action, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CO</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u>, <u>A-D</u>).		

5.5.1 GetTelephonyIdentity()

This action returns the identity of the Telephony Server. If the action succeeds then the output argument TelephonyIdentity contains the unique identity of the Telephony Server. If the identity of the Telephony Server has not been assigned by the time this action is invoked then this action will fail with an error code.

5.5.1.1 Arguments

Table 15 — Arguments for GetTelephonyIdentity()

Argument	Direction	relatedStateVariable
<u>TelephonyIdentity</u>	<u>OUT</u>	<u>A_ARG_TYPE_TelephonyServerIdentity</u>

5.5.1.2 Argument Descriptions

The value of the output argument TelephonyIdentity shall contain a valid URI as specified in [3]. The value follows the standard URI definitions. In case of SIP, the SIP URI [4] is used. In case of a generic resource identified by a telephone number, the TEL URI [6] is used.

The examples of SIP URI as the value of the output argument TelephonyIdentity are as follows:

```

sip:alice@atlanta.com
sip:alice:secretword@atlanta.com;transport=tcp
sips:alice@atlanta.com?subject=project%20x&priority=urgent
sip:+1-212-555-1212:1234@gateway.com;user=phone
sips:1212@gateway.com
sip:alice@192.0.2.4

```

[4] defines the sip: (sips:, for resources to be contacted securely) URI scheme whereas the general form, is: sip:user:password@host:port;uri-parameters?headers

The examples of TEL URI as the value of the output argument TelephonyIdentity are as follows:

Tel:+1-201-555-0123 (This URI points to a phone number in the United States. The hyphens are included to make the number more human readable; they separate country, area code and subscriber number.)

tel:7042;phone-context=example.com (This URI describes a local phone number valid within the context "example.com".)

tel:863-1234;phone-context=+1-914-555 (This URI describes a local phone number that is valid within a particular phone prefix.)

[6] defines the URI scheme tel, which describes any resources identified by telephone numbers. A telephone number is a string of decimal digits that uniquely indicates the network termination point. The number contains the information necessary to route the Call to this point.

[4] 19.1.6 describes how to deal with conversion and compatibility rules among SIP/SIPS URI and TEL URI. Implementers shall be aware of these recommendations.

5.5.1.3 Service Requirements

None.

5.5.1.4 Control Point Requirements When Calling The Action

None.

5.5.1.5 Dependency on Device State

None.

5.5.1.6 Effect on Device State

None.

5.5.1.7 Errors

Table 16 — Error Codes for GetTelephonyIdentity()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
714	Identity does not exist	The identity of the TS has not been assigned yet by the telephony service in the WAN side.

5.5.2 RegisterTelCPName()

This action allows a TelCP to register its TelCPName with the CallManagement service. The registered TelCPName is only valid for the duration of the output argument Expires. Therefore, if the TelCP wants to use the same TCName beyond the duration specified in the output argument Expires, the TelCP shall re-register the TelCPName by invoking this action before the duration expires.

5.5.2.1 Arguments

Table 17 — Arguments for RegisterTelCPName()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>CurrentSecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>NewSecretKey</u>	<u>OUT</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>Expires</u>	<u>OUT</u>	<u>A_ARG_TYPE_Expires</u>

5.5.2.2 Argument Descriptions

The input argument TelCPName contains the TelCPName to be registered with the CallManagement service. When a TelCP invokes this action for the first time, the input

argument TelCPName includes the TelCPName which the TelCP wants to use. When a TelCP invokes this action to continue using the same registered TelCPName beyond its expiration, the input argument TelCPName includes the previously registered TelCPName.

The input argument CurrentSecretKey contains the Secret Key of the TelCP which was assigned to the TelCP from a previous invocation of the action. When a TelCP invokes this action to register a TelCPName for the first time, the value of the input argument CurrentSecretKey is the empty string(""). When a TelCP invokes this action to continue using the registered TelCPName, the CurrentSecretKey shall include the unique Secret Key which was assigned to the TelCP from a previous invocation of the action.

The output argument NewSecretKey contains the newly assigned Secret Key of the TelCP. When a TelCP invokes this action for the first time, the output argument NewSecretKey includes a new Secret Key for the TelCP. When a TelCP invokes this action to continue using the registered TelCPName, the NewSecretKey include the same or a different value that the CallManagement service previously assigned to the TelCP.

The output argument Expires contains the duration for which the TelCPName is valid.

5.5.2.3 Service Requirements

This service shall check the uniqueness of the TelCPName when a TelCP invokes this action to register a TelCPName.

This service shall keep track of all registered TelCPNames and the corresponding Secret Key associated with each TelCPName.

When a TelCP invokes this action to register a TelCPName for the first time and this action succeeds, this service shall event the TelCPNameList state variable to notify the current set of existing TelCPNames.

When a TelCP invokes this action to keep using the registered TelCPName, the CallManagement service shall check whether an authorized TelCP is invoking this action by checking the input arguments TelCPName and CurrentSecretKey.

5.5.2.4 Control Point Requirements When Calling The Action

The TelCP invoke this action when it starts up.

The TelCP shall invoke this action periodically to keep using the same TelCPName before the duration Expires expires.

5.5.2.5 Dependency on Device State

None.

5.5.2.6 Effect on Device State

None.

5.5.2.7 Errors

Table 18 — Error Codes for RegisterTelCPName()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
702	TelCPName already exists	The specified <u>NewTelCPName</u> is already in use.

5.5.3 UnregisterTelCPName()

This action allows a TelCP to unregister the TelCPName. If this action is invoked without registering the TelCPName then the action fails with an error code.

5.5.3.1 Arguments**Table 19 — Arguments for UnregisterTelCPName()**

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>

5.5.3.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action.

The input argument SecretKey contains the Secret Key of the TelCP.

5.5.3.3 Service Requirements

The CallManagement service shall check whether an authorized TelCP is invoking this action by checking the input arguments TelCPName and SecretKey.

If an authorized TelCP invokes this action, the CallManagement service deletes the TelCPName and the Secret Key from the table which manages all the registered TelCPNames and the corresponding Secret Keys. A TelCP without valid TelCPName and Secret Key will not be able to manage a Call.

5.5.3.4 Control Point Requirements When Calling The Action

None.

5.5.3.5 Dependency on Device State

None.

5.5.3.6 Effect on Device State

None.

5.5.3.7 Errors**Table 20 — Error Codes for UnregisterTelCPName()**

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.

5.5.4 ChangeTelCPName()

This action allows a TelCP to change the registered TelCPName.

5.5.4.1 Arguments**Table 21 — Arguments for ChangeTelCPName()**

Argument	Direction	relatedStateVariable
<u>CurrentTelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>CurrentSecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>

Argument	Direction	relatedStateVariable
<u>NewTelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>NewSecretKey</u>	<u>OUT</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>Expires</u>	<u>OUT</u>	<u>A_ARG_TYPE_Expires</u>

5.5.4.2 Argument Descriptions

The input argument CurrentTelCPName contains the current name of the TelCP which invokes this action.

The input argument CurrentSecretKey contains the Secret Key of the TelCP.

The input argument NewTelCPName contains the new TelCPName which the TelCP wants to register.

The output argument NewSecretKey contains the newly assigned Secret Key of the TelCP. It may have the same value as the previously assigned Secret Key when the TelCP registered its TelCPName. The value of the output argument NewSecretKey may be differ from the value which was previously assigned. This is implementation specific.

The output argument Expires contains the duration for which the TelCPName is valid.

5.5.4.3 Service Requirements

The CallManagement service shall check whether an authorized TelCP is invoking this action by checking the input arguments TelCPName and SecretKey.

If an authorized TelCP invokes this action, the CallManagement service updates the TelCPName and the Secret Key in the table which manages all the registered TelCPNames and the corresponding Secret Keys.

5.5.4.4 Control Point Requirements When Calling The Action

None.

5.5.4.5 Dependency on Device State

None.

5.5.4.6 Effect on Device State

None.

5.5.4.7 Errors

Table 22 — Error Codes for ChangeTelCPName()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
702	TelCPName already exists.	The specified <u>NewTelCPName</u> is already in use.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.

5.5.5 GetTelCPNameList()

This action allows a TelCP to get the TelCPNames of all TelCPs registered with the CallManagement service .

5.5.5.1 Arguments

Table 23 — Arguments for GetTelCPNameList()

Argument	Direction	relatedStateVariable
<u>TelCPNameList</u>	<u>OUT</u>	<u>A_ARG_TYPE_TelCPNameList</u>

5.5.5.2 Argument Descriptions

The output argument TelCPNameList includes the list of TelCPNames registered with the CallManagement service.

5.5.5.3 Service Requirements

This service shall manage the TelCPNames which are registered by the TelCPs.

5.5.5.4 Control Point Requirements When Calling The Action

None.

5.5.5.5 Dependency on Device State

None.

5.5.5.6 Effect on Device State

None.

5.5.5.7 Errors

Table 24 — Error Codes for GetTelCPNameList()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
600-699	TBD	See Control clause in [1].

5.5.6 GetMediaCapabilities()

This action returns a set of Media Capabilities supported by the CallManagement service. If the input argument TCMediaCapabilityInfo contains the empty string(""), this action returns all the Media Capabilities which the CallManagement service supports. If the input argument TCMediaCapabilityInfo contains a valid value, this action returns the Media Capabilities supported by the CallManagement service, which will be a subset of TCMediaCapabilityInfo.

5.5.6.1 Arguments

Table 25 — Arguments for GetMediaCapabilities()

Argument	Direction	relatedStateVariable
<u>TCMediaCapabilityInfo</u>	<u>IN</u>	<u>A_ARG_TYPE_MediaCapabilityInfo</u>
<u>SupportedMediaCapabilityInfo</u>	<u>OUT</u>	<u>A_ARG_TYPE_MediaCapabilityInfo</u>

5.5.6.2 Argument Descriptions

The input argument TCMediaCapabilityInfo contains either the empty string("") or the Media Capabilities that a TelCP wants to use for a Media Session. For instance, the input argument TCMediaCapabilityInfo can contain the output argument of the MMS::GetMediaCapabilities() action.

The output argument SupportedMediaCapabilityInfo contains the Media Capabilities of the CallManagement service. If the input argument TCMediaCapabilityInfo contains the empty string(""), this action returns all the Media Capabilities which the CallManagement service supports. If the input argument TCMediaCapabilityInfo contains a valid value, this action

returns the Media Capabilities supported by the CallManagement service, which will be a subset of TCMediaCapabilityInfo.

5.5.6.3 Service Requirements

This service shall return the Media Capabilities of the CallManagement service regardless of its status. If this service supports the TS-Based Content Sharing mechanism, the SupportedMediaCapabilityInfo output argument MUST contain the <contentSharingCapability> element.

5.5.6.4 Control Point Requirements When Calling The Action

It is recommended to invoke this action before the TelCP invokes the StartCall(), AcceptCall() or ModifyCall() to get the latest Media Capabilities of this service.

5.5.6.5 Dependency on Device State

None.

5.5.6.6 Effect on Device State

None.

5.5.6.7 Errors

Table 26 — Error Codes for GetMediaCapabilities()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
600-699	TBD	See Control clause in [1].

5.5.7 StartCall()

This action starts an outgoing Call. When the CallManagement service receives this action, the telephone terminal (e.g. mobile phone, HGW and so on) which has this service starts the new outgoing Call to the Callee. The priority and Media Capabilities of the Call are set by the TelCP and the CallManagement service starts the Call using the specified priority and Media Capabilities set by the TelCP. When the CallMode is set to "PHONE-TelCP", "DP-TelCP" or "DP-User", this Call is created as a Monopolization Mode Call. When the CallMode is set to "Non-Monopolize", this Call is created as a Non-Monopolization Mode Call and can be controlled by any other TelCPs other than the TelCP that is starting the Call.

5.5.7.1 Arguments

Table 27 — Arguments for StartCall()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>CalleeID</u>	<u>IN</u>	<u>A_ARG_TYPE_CalleeID</u>
<u>CallPriority</u>	<u>IN</u>	<u>A_ARG_TYPE_CallPriority</u>
<u>MediaCapabilityInfo</u>	<u>IN</u>	<u>A_ARG_TYPE_MediaCapabilityInfo</u>
<u>CallMode</u>	<u>IN</u>	<u>A_ARG_TYPE_CallMode</u>
<u>CallID</u>	<u>OUT</u>	<u>A_ARG_TYPE_CallID</u>

5.5.7.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

If the Call is a Non-Monopolization Mode Call or "DP-TelCP" or "DP-User", the CallManagement service may ignore the arguments TelCPName and SecretKey.

The input argument CalleeID is the unique ID of the Callee.

The input argument CallPriority identifies the priority of the Call.

The input argument MediaCapabilityInfo contains the Media Capability information that the TelCP wants to use for starting the Call by this action. The Media Capability in this argument shall be a subset of the return value of the GetMediaCapabilities() action.

The input argument CallMode describes the mode of the Call.

The output argument CallID is the return value of this action and uniquely identifies the Call. The CallManagement service assigns an unique CallID for every Call made through this service.

5.5.7.3 Service Requirements

In the case of a PHONE-Based Monopolized Call, when the CallManagement service receives this action, the service shall check whether an authorized TelCP is invoking this action by checking the TelCPName and the SecretKey, and if authorized associate the TelCPName as the Monopolizer for this Call.

In the case of a DP-Based Monopolized Call, when the CallManagement service receives this action, the service shall check whether an authorized identity (i.e. TelCP or User) is invoking this action by checking the Control Point Identity (TelCPID) or the Username associated with the TLS connection as defined in DeviceProtection, and if authorized associate the TelCPID or the Username as the Monopolizer for this Call. See [15] for more detail.

This service shall manage the Call Status and Media Capabilities of the Call created by this action. If this action succeeds, the status of this Call is set to "Dialing" and the CallInfo state variable is evented with the value of the Call Status set to as "Dialing". If the Call is a Monopolization Mode Call, the CallInfo state variable contains the <callID>, <TelCPNames>, and <callStatus> elements. If the Call is a Non-Monopolization Mode Call, the CallInfo state variable contains all the elements.

When the CallManagement service receives response from the Callee that the Callee has received the request of creating a new Call but not accepted yet, the Call Status is set to "Calling" and the CallInfo state variable is evented.

When the CallManagement service receives Early Media response from the WAN side, the Call Status is set to "Calling" and the value of the reason attribute is set to "Receiving Early Media Response". The CallInfo state variable is then evented.

5.5.7.4 Control Point Requirements When Calling The Action

When a TelCP makes a Call using TC(s), the TelCP shall decide the Media Capability for the Call by comparing the Media Capabilities of the CallManagement service and the TC(s). That means the input argument MediaCapabilityInfo for the action shall be the subset of the Media Capabilities of the CallManagement service and that of the TC(s).

When a TelCP receives the CallInfo state variable with the value of the Call Status as "Calling" and the value of the reason attribute as "Receiving Early Media Response", the TelCP sets up Media Session(s) for Early Media between the TS and the TC(s) by invoking the StartMediaSession() action to the TC(s) and the StartMediaTransfer() action to the TS.

A TelCP shall include its TelCPName and SecretKey as input arguments to initiate a PHONE-Based Monopolized Call.

ISO/IEC 29341-26-10:2017(E)

A TelCP shall invoke the action via TLS tunnel as defined in [DeviceProtection](#) to initiate a DP-Based Monopolized Call. See [15] for more detail.

A TelCP shall check whether a User has been logged in or not as defined in [DeviceProtection](#) before invoking this action to initiate a "[DP-User](#)" based Monopolized Call. See [15] for more detail.

When setting up a Call for sharing content to the Callee using the TS-Based Content Sharing mechanism, a TelCP shall provide the content metadata information in the <contentSharingCapability> element of the [MediaCapabilityInfo](#) input argument.

5.5.7.5 Dependency on Device State

NONE

5.5.7.6 Effect on Device State

NONE

5.5.7.7 Errors

Table 28 — Error Codes for [StartCall\(\)](#)

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
705	Invalid priority	The specified priority is invalid.
706	Invalid CallMode	The specified CallMode is invalid.
707	Invalid Media Capability	The specified Media Capability is invalid.
708	Busy	No resource to create a Call is left.
709	Invalid CalleeID	The specified CalleeID is invalid.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.8 [StopCall\(\)](#)

This action allows a TelCP to terminate an existing Call. When this action is invoked, the [CallManagement](#) service terminates the Call identified by the input argument [CallID](#). If succeeds, this service stops sending and/or receiving the Media Streams and releases the relevant resources (e.g. CallID, priority, Media Capability etc.).

If the Call is a Monopolization Mode Call, the [CallManagement](#) service shall check whether the TelCP is authorized to stop the Call. In the case of "[PHONE-TelCP](#)", the [CallManagement](#) service checks the value of the input arguments [TelCPName](#) and [SecretKey](#) supplied by the TelCP and if they match with the values stored for the TelCP in the [CallManagement](#) service then the action will succeed.

5.5.8.1 Arguments

Table 29 — Arguments for [StopCall\(\)](#)

Argument	Direction	relatedStateVariable
TelCPName	IN	A_ARG_TYPE_TelCPName
SecretKey	IN	A_ARG_TYPE_SecretKey
CallID	IN	A_ARG_TYPE_CallID

5.5.8.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

If the Call is a "PHONE-TelCP", these two arguments shall be equal to the TelCPName and Secret Key of the TelCP which created this Call.

If the Call is a Non-Monopolization Mode Call or "DP-TelCP" or "DP-User", the CallManagement service may ignore the arguments TelCPName and SecretKey.

The argument CallID contains the CallID of the Call that the TelCP wants to terminate.

5.5.8.3 Service Requirements

When the CallManagement service receives this action for a "PHONE-TelCP" based Monopolized Call, the service shall check whether the correct TelCP is invoking this action by checking the input arguments TelCPName and SecretKey. This action will succeed if the TelCPName and SecretKey provided in the input arguments correspond to the Monopolizer of this Call.

When the CallManagement service receives this action for a "DP-TelCP" based Monopolized Call, the service shall check whether the correct TelCP is invoking this action by checking the Control Point Identity (TelCPID) associated with the TLS connection as defined in DeviceProtection. This action will succeed if the TelCPID corresponds to the Monopolizer of this Call. See [15] for more detail.

When the CallManagement service receives this action for a "DP-User" based Monopolized Call, the service shall check whether the correct User controlling the TelCP is invoking this action by checking the Username associated with the TLS connection as defined in DeviceProtection. This action will succeed if the Username corresponds to the Monopolizer of this Call. See [15] for more detail.

If the action succeeds then the status of the Call changes to "Disconnected" and the CallInfo state variable gets evented. Once a Call is stopped, the CallManagement service shall release all the resources related to the Call (e.g CallID, Media Capabilities etc.)

5.5.8.4 Control Point Requirements When Calling The Action

Once a Call is stopped, the TelCP shall also stop the Media Sessions on the TC side for the Call. The TelCP then shall release all the resources related with the Call. (e.g CallID, Media Capabilities and so on.)

A TelCP shall include its TelCPName and SecretKey as input arguments to terminate a PHONE-Based Monopolized Call.

A TelCP shall invoke the action via TLS tunnel as defined in DeviceProtection to terminate a DP-Based Monopolized Call. See [15] for more detail.

A TelCP shall check whether a User has been logged in or not as defined in DeviceProtection before invoking this action to terminate a "DP-User" based Monopolized Call. See [15] for more detail.

5.5.8.5 Dependency on Device State

None

5.5.8.6 Effect on Device State

None

5.5.8.7 Errors

Table 30 — Error Codes for StopCall()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.9 AcceptCall()

This action allows a TelCP to accept an incoming Call. The input argument TargetCallID identifies the incoming Call. And the input argument MediaCapabilityInfo specifies the Media Capabilities of the Call. If the action succeeds, the CallManagement service establishes the Call between the Caller and the Callee using the Media Capabilities specified by the input argument MediaCapabilityInfo.

If CallMode is set to as "PHONE-TelCP", "DP-TelCP" or "DP-User", then this Call is created as a Monopolization Mode Call. If CallMode is set to as "Non-Monopolize", then this Call is created as a Non-Monopolization Mode Call.

5.5.9.1 Arguments

Table 31 — Arguments for AcceptCall()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>TargetCallID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallID</u>
<u>MediaCapabilityInfo</u>	<u>IN</u>	<u>A_ARG_TYPE_MediaCapabilityInfo</u>
<u>CallMode</u>	<u>IN</u>	<u>A_ARG_TYPE_CallMode</u>

5.5.9.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

If the Call is a Non-Monopolization Mode Call or "DP-TelCP" or "DP-User", the CallManagement service may ignore the arguments TelCPName and SecretKey.

The input argument TargetCallID identifies the Call to be accepted.

The input argument MediaCapabilityInfo contains the Media Capabilities to be used for the Call. The Media Capabilities in this argument shall be a subset of the values of the <mediaCapability> element in the CallInfo state variable.

The input argument CallMode describes the mode of the Call. If the TelCP makes the Call as a Monopolization Mode Call, then the value of this argument is set to as "PHONE-TelCP", "DP-TelCP" or "DP-User". If the TelCP makes the Call as a Non-Monopolization Mode Call, then the value of this argument is set to as "Non-Monopolize".

5.5.9.3 Service Requirements

When the CallManagement service receives an incoming Call, this service shall event the CallInfo state variable with the value of the <callStatus> element set to as “Ringing”.

If the CallInfo state variable includes TelCPName(s) in the <targetNames> element for this Call, the CallManagement service shall check whether an authorized TelCP is invoking this action by checking if the values of the TelCPName and the SecretKey input arguments are correspond to the authorized TelCP.

When the CallInfo state variable includes TelCPID(s) or Username(s) in the <targetNames> element, then the CallManagement service shall check whether an authorized TelCP or an authorize User controlling the TelCP is invoking this action. The CallManagement Service shall check if the Control Point Identity or Username associated with the TLS connection as defined in DeviceProtection is listed in the <targetNames> element of the CallInfo state variable for the Call. See [15] for more detail.

In the case of a Monopolization Mode Call, when the CallManagement service receives this action, the service shall check whether an authorized TelCP is invoking this action. In the case of PHONE-Based Monopolization, the service checks the input arguments TelCPName and the SecretKey, and if authorized then associate the TelCPName as the Monopolizer for the Call. In the case of “DP-TelCP” or “DP-User”, this service checks the Control Point Identity (TelCPID) or Username associated with the TLS connection as defined in DeviceProtection, and then associate the TelCPID or Username as an Monopolizer for the Call. See [15] for more detail.

If the Call is a Monopolization Mode Call, the CallManagement service shall keep track of the identity (i.e. a TelCP or a User identified by TelCPName, TelCPID or Username) which is accepting the Call.

If the Call is established between the Caller and the Callee, then the CallManagement service shall event CallInfo state variable with the <callStatus> element set to as “Connected” and the <mediaCapability> element set to as the value of Media Capabilities of the connected Call.

5.5.9.4 Control Point Requirements When Calling The Action

The TelCP shall keep track of the CallID of the accepted Call. The TelCP can use this CallID to control the Call.

After receiving the CallInfo event with TelCPName(s) in the <targetNames> element, and/or when the TelCP wants to Monopolize the Call using Phone-Based Monopolization mechanism, then the TelCP shall include the TelCPName and SecretKey as input arguments to accept the Call.

After receiving the CallInfo event with TelCPID(s) in the <targetNames> element, and/or when the TelCP wants to Monopolize the Call using “DP-TelCP” based Monopolization mechanism, then the TelCP shall invoke the action via TLS tunnel as defined in DeviceProtection to accept the Call. See [15] for more detail.

After receiving the CallInfo event with Username(s) in the <targetNames> element, and/or when the TelCP wants to Monopolize the Call using “DP-User” based Monopolization mechanism then the TelCP shall check whether a User has been logged in or not as defined in DeviceProtection before invoking the action to accept the Call. See [15] for more detail.

5.5.9.5 Dependency on Device State

None

5.5.9.6 Effect on Device State

None

5.5.9.7 Errors

Table 32 — Error Codes for AcceptCall()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
600-699	TBD	See Control clause in [1].
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>TargetCallID</u> is invalid. The value of <u>TargetCallID</u> shall be the same as the value of <callID> element in the <u>CallInfo</u> state variable.
706	Invalid CallMode	The specified <u>CallMode</u> is invalid.
707	Invalid Media Capability	The specified Media Capability is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.10 RejectCall()

This action allows a TelCP to reject an incoming Call or to reject a request to modify the Media Capabilities of an existing Call. The input argument TargetCallID identifies the Call to be rejected.

5.5.10.1 Arguments

Table 33 — Arguments for RejectCall()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	IN	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	IN	<u>A_ARG_TYPE_SecretKey</u>
<u>TargetCallID</u>	IN	<u>A_ARG_TYPE_CallID</u>
<u>RejectReason</u>	IN	<u>A_ARG_TYPE_RejectReason</u>

5.5.10.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key for the TelCP. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

In the case of "PHONE-TelCP", a TelCP with the correct values of the TelCPName and the SecretKey can only reject the Call.

If the Call is a Non-Monopolization Mode Call or "DP-TelCP" or "DP-User", the CallManagement service may ignore the arguments TelCPName and SecretKey.

The input argument TargetCallID identifies the Call to be rejected.

The input argument RejectReason describes the reason for rejecting the incoming Call or the request to modify the existing Call.

5.5.10.3 Service Requirements

Once the action succeeds, the CallManagement service shall event the CallInfo state variable. In the case of rejecting an incoming Call, the value of the <callStatus> element in the CallInfo state variable set to as "Disconnected" and the value of the reason attribute set to as

“Rejected”. Once the Call is rejected, the CallManagement service deletes all the resources associated with the Call.

If the CallInfo state variable includes the TelCPName(s) in the <targetNames> element, then the CallManagement service shall check whether an authorized TelCP is invoking this action by checking if the value of the TelCPName and the SecretKey input arguments are correspond to the authorized TelCP.

When the CallInfo state variable includes TelCPID(s) or Username(s) in the <targetNames> element, the CallManagement service shall check whether an authorized TelCP or an authorize User controlling the TelCP is invoking this action by checking if the Control Point Identity or Username associated with the TLS connection as defined in DeviceProtection is listed in the <targetNames> element of the CallInfo state variable for this Call. See [15] for more detail.

In the case of rejecting an incoming modify request for an existing Phone-Based Monopolized Call, the CallManagement service shall check whether an authorized TelCP is invoking this action by checking if the values of the TelCPName and SecretKey are corresponds to the Monopolizer of the Call

In the case of rejecting an incoming modify request for an existing “DP-TelCP” or “DP-User” based Monopolized Call,, the CallManagement service shall check whether an authorized TelCP or a User controlling a TelCP is invoking this action by checking if the Control Point Identity or the Username associated with the TLS connection as defined in DeviceProtection is corresponds to the Monopolizer of the Call. See [15] for more detail.

In the case of rejecting a request to modify an existing Call, the value of the <callStatus> element in the CallInfo state variable set to as “ModifyFailed” and the value of the reason attribute set to as “Rejected”. If the request to modify an existing Call is rejected, the Call continues without any changes to the Media Capabilities.

5.5.10.4 Control Point Requirements When Calling The Action

Once the incoming Call is rejected, the TelCP deletes all the resources associated with the Call.

If the request to modify an existing Call is rejected, the Call continues without any changes to the Media Capabilities.

If TelCP wants to reject an incoming Call request after receiving the CallInfo event with TelCPName(s) in the <targetNames> element for that incoming Call or if TelCP wants to reject an incoming modify request for an existing PHONE-Based Monopolized Call, then TelCP shall include the TelCPName and the SecretKey input arguments in this action.

If TelCP wants to reject an incoming Call request after receiving the CallInfo event with TelCPName(s) in the <targetNames> element for that incoming Call or if TelCP wants to reject an incoming modify request for an existing “DP-TelCP” based Monopolized Call, then the TelCP shall invoke this action via TLS tunnel as defined in DeviceProtection. See [15] for more detail.

If TelCP wants to reject an incoming Call request after receiving the CallInfo event with Username(s) in the <targetNames> element for that incoming Call or if TelCP wants to reject an incoming modify request for an existing “DP-User” based Monopolized Call, then the TelCP shall check whether a User has been logged in or not as defined in DeviceProtection. See [15] for more detail

5.5.10.5 Dependency on Device State

None

5.5.10.6 Effect on Device State

None

5.5.10.7 Errors

Table 34 — Error Codes for RejectCall()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.11 ModifyCall()

This action allows a TelCP to request to modify the Media Capabilities of an existing Call for the Remote Party. If the Remote Party accepts the request, the Media Capabilities of the Call identified by the CallID will be changed to the new Media Capabilities.

The ModifyCall() action can perform the following functions:

- Requesting to add a new Media Stream to the Call. (e.g., changing audio Call to video Call by adding a video stream.)
- Requesting to delete an existing Media Stream from the Call. (e.g., changing video Call to audio Call by deleting a video stream.)
- Request to change the Media Capabilities such as media codec parameters. (e.g., changing the framerate of the video stream in a video Call.)

5.5.11.1 Arguments

Table 35 — Arguments for ModifyCall()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>TargetCallID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallID</u>
<u>MediaCapabilityInfo</u>	<u>IN</u>	<u>A_ARG_TYPE_MediaCapabilityInfo</u>

5.5.11.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

If the Call is a Non-Monopolization Mode Call or "DP-TelCP" or "DP-User", the CallManagement service may ignore the arguments TelCPName and SecretKey.

The input argument TargetCallID identifies the Call to be modified.

The input argument MediaCapabilityInfo contains the new Media Capabilities for the Call.

5.5.11.3 Service Requirements

Once the action succeeds, the CallManagement service shall event the CallInfo state variable with the value of the <callStatus> element set to as "SendingModifyRequest".

Once the modification of the Media Capabilities are accepted by the Remote Party in the WAN side, this service shall notify the CallInfo state variable with the value of the <callStatus> element set to as "Modified". If the modification of the Media Capabilities are rejected by the WAN side, then the service shall notify the CallInfo state variable with the value of the <callStatus> element set to as "ModifyFailed".

When the CallManagement service receives this action for a "PHONE-TelCP" based Monopolized Call, then the service shall check whether the correct TelCP is invoking this action by checking the input arguments TelCPName and SecretKey of this action. This action will succeed if the TelCPName and SecretKey are correspond to the Monopolizer of this Call.

When the CallManagement service receives this action for a "DP-TelCP" based Monopolized Call, then the service shall check whether the correct TelCP is invoking this action by checking the Control Point Identity (TelCPID) associated with the TLS connection as defined in DeviceProtection. This action will succeed if the TelCPID corresponds to the Monopolizer of this Call.

When the CallManagement service receives this action for a "DP-User" based Monopolized Call, then the service shall check whether the correct User controlling the TelCP is invoking this action by checking the Username associated with the TLS connection as defined in DeviceProtection. This action will succeed if the Username corresponds to the Monopolizer of this Call. See [15] for the more detail.

When receiving this action with the <contentSharingCapability> element in the input argument MediaCapabilityInfo, this service shall check if the Call needs to be updated for sharing the content to the Remote Party. If not (e.g., the TS enables mixing), this service shall first event the CallInfo state variable with the value of the <callStatus> element set as "SendingModifyRequest", and then event the CallInfo state variable with the value of the <callStatus> element as "Modified".

5.5.11.4 Control Point Requirements When Calling The Action

Once the TelCP receives an event for the CallInfo state variable with the <callStatus> element set to as "Modified", the TelCP shall invoke the StartMediaTransfer() action to update the Media Session as the result of the ModifyCall() action.

A TelCP shall include the TelCPName and SecretKey as input arguments when invoking this action to modify a "PHONE-TelCP" based Monopolized Call.

A TelCP shall invoke the action via TLS tunnel as defined in DeviceProtection to modify a "DP-TelCP" or "DP-User" based Monopolized Call. See [15] for more detail.

A TelCP shall check whether a User has been logged in or not as defined in DeviceProtection before invoking this action to modify a "DP-User" based Monopolized Call. See [15] for more detail.

When updating a Call for sharing content to the Remote Party using the TS-Based Content Sharing mechanism, a TelCP shall provide the content metadata information in the <contentSharingCapability> element of the input argument MediaCapabilityInfo.

5.5.11.5 Dependency on Device State

None

5.5.11.6 Effect on Device State

None

5.5.11.7 Errors

Table 36 — Error Codes for ModifyCall()

ErrorCode	errorDescription	Description
-----------	------------------	-------------

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid.
707	Invalid Media Capability	The specified Media Capability is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.12 AcceptModifyCall()

This action allows a TelCP to accept a request from the Remote Party to modify the Media Capabilities of an existing Call. If the TelCP accepts the request, the Media Capabilities of the Call identified by the CallID will be changed to the new Media Capabilities.

The AcceptModifyCall() action can perform the following functions:

- Accepting a request of adding a new Media Stream to the Call. (e.g., changing audio Call to video Call by adding a video stream.)
- Accepting a request of deleting an existing Media Stream from the Call. (e.g., changing video Call to audio Call by deleting a video stream.)
- Accepting a request of changing the Media Capabilities such as media codec parameters. (e.g., changing the framerate of the video stream in a video Call.)

5.5.12.1 Arguments

Table 37 — Arguments for AcceptModifyCall()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>TargetCallID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallID</u>
<u>MediaCapabilityInfo</u>	<u>IN</u>	<u>A_ARG_TYPE_MediaCapabilityInfo</u>

5.5.12.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

If the Call is a Non-Monopolization Mode Call or "DP-TelCP" or "DP-User", the CallManagement service may ignore the arguments TelCPName and SecretKey.

The input argument TargetCallID identifies the Call to be modified.

The input argument MediaCapabilityInfo contains the new Media Capabilities for the Call.

5.5.12.3 Service Requirements

When the CallManagement service receives the request of modifying the Media Capabilities of an existing Call, the CallManagement service shall event the CallInfo state variable with the value of the <callStatus> element set to as "ReceivingModifyRequest".

Once the modification of the Media Capabilities are accepted by the TelCP in the WAN side, this service shall notify the CallInfo state variable with the value of the <callStatus> element

set to as “Modified”. If the modification of the Media Capabilities are rejected by the WAN side, then the service shall notify the CallInfo state variable with the value of the <callStatus> element set to as “ModifyFailed”.

When the CallManagement service receives this action for a “PHONE-TelCP” based Monopolized Call, then the service shall check whether the correct TelCP is invoking this action by checking the input arguments TelCPName and SecretKey of this action. This action will succeed if the TelCPName and SecretKey are correspond to the Monopolizer of the Call.

When the CallManagement service receives this action for a “DP-TelCP” based Monopolized Call, then the service shall check whether the correct TelCP is invoking this action by checking the Control Point Identity (TelCPID) associated with the TLS connection as defined in DeviceProtection. This action will succeed if the TelCPID corresponds to the Monopolizer of the Call.

When the CallManagement service receives this action for a “DP-User” based Monopolized Call, then the service shall check whether the correct User controlling the TelCP is invoking this action by checking the Username associated with the TLS connection as defined in DeviceProtection. This action will succeed if the Username corresponds to the Monopolizer of the Call. See [15] for the more detail.

5.5.12.4 Control Point Requirements When Calling The Action

Once the TelCP receives an event for the CallInfo state variable with the <callStatus> element set to as “Modified”, the TelCP shall invoke the StartMediaTransfer() action to update the Media Session as the result of the AcceptModifyCall() action.

A TelCP shall include the TelCPName and SecretKey as input arguments when invoking this action to accept the modify request for an existing “PHONE-TelCP” based Monopolized Call.

A TelCP shall invoke the action via TLS tunnel as defined in DeviceProtection to accept the modify request for an existing “DP-TelCP” or “DP-User” based Monopolized Call. See [15] for more detail.

A TelCP shall check whether a User has been logged in or not as defined in DeviceProtection before invoking this action to accept the modify request for an existing “DP-User” based Monopolized Call. See [15] for more detail.

5.5.12.5 Dependency on Device State

None

5.5.12.6 Effect on Device State

None

5.5.12.7 Errors

Table 38 — Error Codes for AcceptModifyCall()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
700		Reserved for future extensions.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid.
707	Invalid Media Capability	The specified Media Capability is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.13 StartMediaTransfer()

This action is used to allow a TelCP to request the CallManagement service to start transferring Media Streams. If this action succeeds, then the TS and the TC start transferring the media.

5.5.13.1 Arguments**Table 39 — Arguments for StartMediaTransfer()**

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>TargetCallID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallID</u>
<u>TCList</u>	<u>IN</u>	<u>A_ARG_TYPE_TCList</u>
<u>MediaCapabilityInfo</u>	<u>IN</u>	<u>A_ARG_TYPE_MediaCapabilityInfo</u>

5.5.13.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

If the Call is a Non-Monopolization Mode Call or "DP-TelCP" or "DP-User", the CallManagement service may ignore the arguments TelCPName and SecretKey.

The input argument TargetCallID identifies the Call.

The argument TCList contains the list of the TCs associated with the Call.

The input argument MediaCapabilityInfo contains the Media Capabilities for the Call.

5.5.13.3 Service Requirements

After this action succeeds, the CallManagement service starts sending and/or receiving Media Streams to or from the TC(s). If the Call Status is "Connected" when this action is invoked, then the Call Status is changed to "Talling". If the Call Status is "Calling" and the value of the reason attribute is "Receiving Early Media Response", then the Call Status is remain as it is and the value of the reason attribute is set to "Early Media Started". Then the CallInfo state variable is evented.

When the CallManagement service receives this action for a "PHONE-TelCP" based Monopolized Call, then the service shall check whether the correct TelCP is invoking this action by checking the input arguments TelCPName and SecretKey of this action. This action will succeed if the TelCPName and SecretKey are correspond to the Monopolizer of the Call.

When the CallManagement service receives this action for a "DP-TelCP" based Monopolized Call, then the service shall check whether the correct TelCP is invoking this action by checking the Control Point Identity (TelCPID) associated with the TLS connection as defined in DeviceProtection. This action will succeed if the TelCPID corresponds to the Monopolizer of the Call.

When the CallManagement service receives this action for a "DP-User" based Monopolized Call, then the service shall check whether the correct User controlling the TelCP is invoking this action by checking the Username associated with the TLS connection as defined in DeviceProtection. This action will succeed if the Username corresponds to the Monopolizer of the Call. See [15] for the more detail.

When receiving this action with the <contentSharingCapability> element in the MediaCapabilityInfo input argument, this service shall start retrieving the content based on the provided metadata information, process and send the content media to the Remote Party using the Media Session of the Call.

5.5.13.4 Control Point Requirements When Calling The Action

The TelCP shall invoke this action after receiving an event for the CallInfo state variable with the value of the <callStatus> element set to as either “Connected” or, “Modified” or, “ModifyFailed”.

A TelCP shall include the TelCPName and SecretKey as input arguments when invoking this action to start the media transfer between TS and TC for the “PHONE-TelCP” based Monopolized Call.

A TelCP shall invoke the action via TLS tunnel as defined in DeviceProtection to start the media transfer between TS and TC for the “DP-TelCP” or “DP-User” based Monopolized Call. See [15] for more detail.

A TelCP shall check whether a User has been logged in or not as defined in DeviceProtection before invoking this action to start the media transfer between TS and TC for the “DP-User” based Monopolized Call. See [15] for more detail.

A TelCP shall invoke the StartCall() or the ModifyCall() action before invoking this action to set up the TS-Based Content Sharing session. When invoking this action, the TelCP shall provide the same content information in the MediaCapabilityInfo input argument as when invoking the StartCall() or the ModifyCall() action.

5.5.13.5 Dependency on Device State

None

5.5.13.6 Effect on Device State

None

5.5.13.7 Errors

Table 40 — Error Codes for StartMediaTransfer()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid.
707	Invalid Media Capability	The specified Media Capability is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.

5.5.14 ChangeMonopolizer()

This action allows to change the control of a Monopolize Mode Call. In the case of a Monopolization Mode Call, the TelCP that established the Call can only manage the Call. Therefore, this action is used to change the owner of the Call from one TelCP to another.

5.5.14.1 Arguments

Table 41 — Arguments for ChangeMonopolizer()

Argument	Direction	relatedStateVariable
<u>CurrentMonopolizer</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>

Argument	Direction	relatedStateVariable
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>CallID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallID</u>
<u>NewMonopolizer</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>

5.5.14.2 Argument Descriptions

The input argument CurrentMonopolizer contains the name of the TelCP which invokes this action.

The input argument SecretKey contains the Secret Key of the current TelCP whose TelCPName is identified by the input argument CurrentMonopolizer.

The input argument CallID identifies the Call.

The input argument NewMonopolizer identifies the new TelCP that will get the control of the Call.

5.5.14.3 Service Requirements

When the CallManagement service receives this action for a “PHONE-TelCP” Based Monopolized Call, the service shall check whether the correct TelCP is invoking this action by checking the input arguments CurrentMonopolizer and SecretKey. This action will succeed if the TelCPName and SecretKey corresponds to the Monopolizer of the Call..

If succeeds, the TelCP as identified by the input argument NewMonopolizer will get the ownership of the Call and will be able to manage the Call. This service sends the CallInfo event with the value of the <targetNames> element set to the value of the input argument NewMonopolizer.

5.5.14.4 Control Point Requirements When Calling The Action

A TelCP shall not invoke this action when the value of the Call Status is “Ringing”.

A TelCP shall provide its TelCPName and the SecretKey as input arguments when invoking this action to switch the Monopolizer of a “PHONE-TelCP” Based Monopolized Call.

5.5.14.5 Dependency on Device State

None

5.5.14.6 Effect on Device State

None

5.5.14.7 Errors

Table 42 — Error Codes for ChangeMonopolizer()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid. e.g. CallID does not exist, a Call which CallID is the same as the value of <u>TargetCallID</u> is not a Monopolization Mode Call.
710	Invalid NewMonopolizer	The specified <u>NewMonopolizer</u> is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.15 InitiateCall()

This action allows a TelCP to initiate an outgoing Call to a Callee identified by the input argument CalleeID. The TelCP only initiates the Call at the TS to the Callee. However this Call is not managed by the UPnP Telephony functionalities. The TS does not establish any Media Sessions with any TC in the home network for this particular Call.

5.5.15.1 Arguments**Table 43 — Arguments for InitiateCall()**

Argument	Direction	relatedStateVariable
<u>CalleeID</u>	<u>IN</u>	<u>A_ARG_TYPE_CalleeID</u>
<u>CallID</u>	<u>OUT</u>	<u>A_ARG_TYPE_CallID</u>

5.5.15.2 Argument Descriptions

The input argument CalleeID contains the identity of the Callee.

The output argument CallID is the return value of this action and uniquely identifies the Call. The CallManagement service assigns a unique CallID for every Call made through this service.

5.5.15.3 Service Requirements

The CallManagement service may event the CallInfo state variable for the status of the Call which is initiated by this action.

5.5.15.4 Control Point Requirements When Calling The Action

None

5.5.15.5 Dependency on Device State

None

5.5.15.6 Effect on Device State

None

5.5.15.7 Errors**Table 44 — Error Codes for InitiateCall()**

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
709	Invalid CalleeID	The specified <u>CalleeID</u> is invalid.

5.5.16 GetCallInfo()

This action allows a TelCP to get information about existing Calls. If the input argument TargetCallID contains the empty string(""), this action returns information about all the existing Calls such as CallID, Call Status, Media Capabilities etc. If the value of the input argument TargetCallID contains a valid CallID, then the action returns information about that specific Call.

5.5.16.1 Arguments**Table 45 — Arguments for GetCallInfo()**

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>

Argument	Direction	relatedStateVariable
<u>TargetCallID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallID</u>
<u>CallInfoList</u>	<u>OUT</u>	<u>A_ARG_TYPE_CallInfoList</u>

5.5.16.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action or the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP or the empty string("").

If the TelCP wants to get the information of the Call which mode is "PHONE-TelCP", the input arguments TelCPName and SecretKey shall contain the TelCPName and Secret Key which are related to the Monopolization Mode Call. Otherwise, the input arguments TelCPName and SecretKey may contain the empty string("").

The input argument TargetCallID identifies a specific Call or the empty string("").

If the value of the input argument TargetCallID contains a valid CallID, the output argument CallInfoList contains information about the Call specified by TargetCallID. If the value of TargetCallID is the empty string(""), then the output argument CallInfoList contains information about all the Calls which are being managed by the CallManagement service.

5.5.16.3 Service Requirements

If the Call is managed by the CallManagement service in a Monopolization mode, then the CallManagement service returns only a subset of information such as the <callID>, <TelCPNames>, and <callStatus> elements in the CallInfo state variable. However, if the action is invoked by the TelCP or the User controlling the TelCP which is the Monopolizer of the Call, then the CallManagement service returns all the elements of the CallInfo state variable.

5.5.16.4 Control Point Requirements When Calling The Action

A TelCP shall include the TelCPName and SecretKey as input arguments when invoking this action to retrieve complete information of the Call.

A TelCP shall invoke the action via TLS tunnel as defined in DeviceProtection to retrieve the complete information of the Call for the "DP-TelCP" or "DP-User" based Monopolized Call. See [15] for more detail.

A TelCP shall check whether a User has been logged in or not as defined in DeviceProtection before invoking this action for the "DP-User" based Monopolized Call. See [15] for more detail.

5.5.16.5 Dependency on Device State

None

5.5.16.6 Effect on Device State

None

5.5.16.7 Errors

Table 46 — Error Codes for GetCallInfo()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified <u>TelCPName</u> or <u>SecretKey</u> is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid.

ErrorCode	errorDescription	Description
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.17 GetCallLogs()

This action allows a TelCP to retrieve the logs of all the Calls managed by the CallManagement service. The log information contains the identification of the Caller and the Callee, the time and duration of the Calls, and the types of the Calls etc. The CallManagement service keeps tracks of the logs for all the Calls managed by the CallManagement service until the logs are cleared, and returns the logs as the output argument of this action. The logs can be cleared by the CallManagement service when it reaches the maximum limit. The policy and the size of the limit are implementation specific. The logs can be cleared by a TelCP by the invocation of the ClearCallLogs() action.

5.5.17.1 Arguments

Table 47 — Arguments for GetCallLogs()

Argument	Direction	relatedStateVariable
<u>CallLogs</u>	<u>OUT</u>	<u>A_ARG_TYPE_CallLogs</u>

5.5.17.2 Argument Descriptions

The input argument CallLogs contains the logs of all the Calls. If the CallLogs is empty, then the output argument includes the XML fragment of type CallLogs without any CallLog elements.

5.5.17.3 Service Requirements

None.

5.5.17.4 Control Point Requirements When Calling The Action

None.

5.5.17.5 Dependency on Device State

None.

5.5.17.6 Effect on Device State

None.

5.5.17.7 Errors

Table 48 — Error Codes for GetCallLogs()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.

5.5.18 ClearCallLogs()

This action allows a TelCP to clear the logs of all the Calls managed by the CallManagement service. This includes all incoming, outgoing, and missed Calls managed by the CallManagement service. The CallManagement service will start keeping new logs for Calls managed by the CallManagement service since the invocation of this action.

5.5.18.1 Arguments

None.

5.5.18.2 Argument Descriptions

None.

5.5.18.3 Service Requirements

None.

5.5.18.4 Control Point Requirements When Calling The Action

None.

5.5.18.5 Dependency on Device State

None.

5.5.18.6 Effect on Device State

None.

5.5.18.7 Errors

Table 49 — Error Codes for ClearCallLogs()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.

5.5.19 RegisterCallBack()

This action allows a TelCP to register a Call back with the CallManagement service. If succeeds, the action returns an ID that uniquely identifies the Call back. If the Call back is already registered before for the Callee, then this action still succeed without any changes in the Call back information.

5.5.19.1 Arguments

Table 50 — Arguments for RegisterCallBack()

Argument	Direction	relatedStateVariable
<u>CalleeID</u>	<u>IN</u>	<u>A_ARG_TYPE_CalleeID</u>
<u>CallBackID</u>	<u>OUT</u>	<u>A_ARG_TYPE_CallBackID</u>

5.5.19.2 Argument Descriptions

The input argument CalleeID uniquely identify the Callee.

The output argument CallBackID uniquely identifies the Call back which is being registered with the CallManagement service by the invocation of this action.

5.5.19.3 Service Requirements

None.

5.5.19.4 Control Point Requirements When Calling The Action

None.

5.5.19.5 Dependency on Device State

None.

5.5.19.6 Effect on Device State

None.

5.5.19.7 Errors

Table 51 — Error Codes for RegisterCallback()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
709	Invalid CalleeID	The specified <u>CalleeID</u> is invalid.

5.5.20 ClearCallback()

This action allows a TelCP to clear a Call back registered with the CallManagement service. The CallbackID identifies the Call back that will be cleared. The action will fail with an error code if the specified CallbackID does not exist. If the value of the CallbackID is the empty string("") then all the Call backs registered with the CallManagement service will be cleared.

5.5.20.1 Arguments

Table 52 — Arguments for ClearCallback()

Argument	Direction	relatedStateVariable
<u>CallbackID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallbackID</u>

5.5.20.2 Argument Descriptions

The input argument CallbackID uniquely identifies the Call back which is to be cleared.

5.5.20.3 Service Requirements

None.

5.5.20.4 Control Point Requirements When Calling The Action

None.

5.5.20.5 Dependency on Device State

None.

5.5.20.6 Effect on Device State

None.

5.5.20.7 Errors

Table 53 — Error Codes for ClearCallback()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
711	Invalid CallbackID	The specified <u>CallbackID</u> is invalid.

5.5.21 GetCallbackInfo()

This action allows a TelCP to retrieve information about all the Call backs registered with the CallManagement service.

5.5.21.1 Arguments

Table 54 — Arguments for GetCallbackInfo()

Argument	Direction	relatedStateVariable
----------	-----------	----------------------

Argument	Direction	relatedStateVariable
<u>CallBackInfo</u>	<u>OUT</u>	<u>A_ARG_TYPE_CallBackInfoList</u>

5.5.21.2 Argument Descriptions

The output argument CallBackInfo contains Call back information registered with the CallManagemnet service.

5.5.21.3 Service Requirements

None.

5.5.21.4 Control Point Requirements When Calling The Action

None.

5.5.21.5 Dependency on Device State

None.

5.5.21.6 Effect on Device State

None.

5.5.21.7 Errors

Table 55 — Error Codes for GetCallBackInfo()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.

5.5.22 ChangeCallMode()

This action allows a TelCP to change the CallMode of an existing Call:

- Changing the CallMode from Monopolization to Non-Monopolization
- Changing the CallMode from Non-Monopolization to Monopolization

5.5.22.1 Arguments

Table 56 — Arguments for ChangeCallMode()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>CallID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallID</u>
<u>CallMode</u>	<u>IN</u>	<u>A_ARG_TYPE_CallMode</u>

5.5.22.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

In the case of "PHONE-TelCP", a TelCP that provides the correct values for the TelCPName and the SecretKey can change the mode of the Call.

The argument CallID contains the identification of the Call.

The argument CallMode contains the new CallMode for the Call.

5.5.22.3 Service Requirements

In the case of Monopolization Mode Call, when the CallManagement service receives this action, the service shall check whether an authorized TelCP is invoking this action. Then the CallManagement service changes the CallMode to the value of the input argument CallMode.

When the CallMode is changed from Monopolization Mode Call to Non-Monopolization Mode Call, the TelCP which invokes this action loses the exclusive right to control the Call and all TelCPs or users are able to control the Call.

After the CallMode is changed from Non-Monopolization Mode Call to Monopolization Mode Call or vice versa, the CallManagement service events the CallInfo state variable. The TargetNames element contains the approximate name of the TelCP or user that can control the Call.

5.5.22.4 Control Point Requirements When Calling The Action

A TelCP shall include its TelCPName and SecretKey as input arguments when changing the CallMode to or from PHONE-Based Monopolized Call.

A TelCP shall invoke this action via TLS tunnel as defined in DeviceProtection when changing the CallMode to or from DP-Based Monopolized Call.

A TelCP shall check whether a User is logged in as defined in DeviceProtection before invoking this action while changing the CallMode to or from "DP-User" based Monopolized Call.

5.5.22.5 Dependency on Device State

None.

5.5.22.6 Effect on Device State

None.

5.5.22.7 Errors

Table 57 — Error Codes for ChangeCallMode()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid.
706	Invalid CallMode	The specified <u>CallMode</u> is invalid.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.23 GetPushInfo()

This action allows a TelCP to get a list of evented Push Information.

5.5.23.1 Arguments

Table 58 — Arguments for GetPushInfo()

Argument	Direction	relatedStateVariable
<u>PushInfoList</u>	<u>OUT</u>	<u>A_ARG_TYPE_PushInfoList</u>

5.5.23.2 Argument Descriptions

The output argument PushInfoList contains a list of evented Push Information.

5.5.23.3 Service Requirements

None

5.5.23.4 Control Point Requirements When Calling The Action

None

5.5.23.5 Dependency on Device State

None.

5.5.23.6 Effect on Device State

None.

5.5.23.7 Errors

Table 59 — Error Codes for GetPushInfo()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.

5.5.24 IgnoreCall()

This action allows a TelCP to ignore an incoming Call or an incoming modify request. The input argument CallID identifies the Call.

After a TelCP ignores an incoming Call or a incoming modify request, the TelCP is not authorized to control the Call any longer. If there are other TelCP(s) subscribed to the same TS, they can accept the Call or the request.

5.5.24.1 Arguments

Table 60 — Arguments for IgnoreCall()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>CallID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallID</u>
<u>IgnoreReason</u>	<u>IN</u>	<u>A_ARG_TYPE_RejectReason</u>

5.5.24.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP which invokes this action. If the mode of the Call is not "PHONE-TelCP", the value of this input argument can be set to the empty string("").

In the case of "PHONE-TelCP", a TelCP with the correct values of the TelCPName and the SecretKey can only pass the Call.

The input argument CallID identifies the Call to be passed.

The input argument IgnoreReason describes the reason for ignoring the incoming Call or the request to modify the existing Call.

5.5.24.3 Service Requirements

When the CallManagement service receives this action, the service waits for the AcceptCall(), AcceptModifyCall(), RejectCall(), StopCall() and IgnoreCall() actions from other TelCP(s) for the specific duration based on the TS implementation. Then, when the CallManagement service receives AcceptCall(), AcceptModifyCall(), RejectCall() or StopCall() action, the CallManagement service works according to the Service Requirement in the received action.

For example, if the TS receives the IgnoreCall() action from all the subscribed TelCP(s), the service may return an error to the Remote Party immediately.

5.5.24.4 Control Point Requirements When Calling The Action

A TelCP shall include its TelCPName and SecretKey as input arguments when the CallMode is PHONE-Based Monopolized Call.

A TelCP shall invoke the action via TLS tunnel as defined in DeviceProtection when a current CallMode is DP-Based Monopolized Call. See [15] for more detail.

A TelCP shall check whether a User has been logged in as defined in DeviceProtection before invoking this action to when a current CallMode is "DP-User" based Monopolized Call. See [15] for more detail.

5.5.24.5 Dependency on Device State

None.

5.5.24.6 Effect on Device State

None.

5.5.24.7 Errors

Table 61 — Error Codes for IgnoreCall()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
703	Invalid CallID	The specified <u>CallID</u> is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.

5.5.25 GetVoiceMail()

This action allows a TelCP to retrieve currently available voice mails. If the input argument VoiceMailID contains the empty string (""), this action returns information about all the existing voice mails. If the value of the input argument VoiceMailID contains a valid value, then the action returns information for the voice mail identified by the VoiceMailID.

5.5.25.1 Arguments

Table 62 — Arguments for GetVoiceMail()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>

Argument	Direction	relatedStateVariable
<u>VoiceMailID</u>	<u>IN</u>	<u>A_ARG_TYPE_VoiceMailID</u>
<u>VoiceMailInfoList</u>	<u>OUT</u>	<u>A_ARG_TYPE_VoiceMailInfoList</u>

5.5.25.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. The value of this input argument can be set to the empty string ("").

The input argument SecretKey contains the Secret Key of the TelCP. The value of this input argument can be set to the empty string ("").

The input argument VoiceMailID is the unique ID of the voice mail. The value of this input argument can be empty string ("").

The output argument VoiceMailInfoList contains all existing voice mails if the input argument VoiceMailID contains the empty string (""). This output argument contains only a single voice mail information if the input argument VoiceMailID contains a valid identification.

5.5.25.3 Service Requirements

None

5.5.25.4 Control Point Requirements When Calling The Action

None

5.5.25.5 Dependency on Device State

None.

5.5.25.6 Effect on Device State

None.

5.5.25.7 Errors

Table 63 — Error Codes for GetVoiceMail()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
715	Invalid Voice Mail ID	The ID of the voice mail does not exist.

5.5.26 DeleteVoiceMail()

This action allows a TelCP to delete an existing voice mail. If the input argument VoiceMailID contains the empty string (""), this action deletes all the existing voice mails. If the value of the input argument VoiceMailID contains a valid identification for a voice mail, then the action deletes corresponding voice mail.

5.5.26.1 Arguments

Table 64 — Arguments for DeleteVoiceMail()

Argument	Direction	relatedStateVariable
<u>TelCPName</u>	<u>IN</u>	<u>A_ARG_TYPE_TelCPName</u>
<u>SecretKey</u>	<u>IN</u>	<u>A_ARG_TYPE_SecretKey</u>
<u>VoiceMailID</u>	<u>IN</u>	<u>A_ARG_TYPE_VoiceMailID</u>

5.5.26.2 Argument Descriptions

The input argument TelCPName contains the name of the TelCP which invokes this action. The value of this input argument can be set to the empty string("").

The input argument SecretKey contains the Secret Key of the TelCP. The value of this input argument can be set to the empty string("").

The input argument VoiceMailID contains the ID of a voice mail.

5.5.26.3 Service Requirements

None

5.5.26.4 Control Point Requirements When Calling The Action

None

5.5.26.5 Dependency on Device State

None.

5.5.26.6 Effect on Device State

None.

5.5.26.7 Errors

Table 65 — Error Codes for DeleteVoiceMail()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.
715	Invalid Voice Mail ID	The ID of the voice mail does not exist

5.5.27 EnhancedInitiateCall()

This action allows a TelCP to initiate an outgoing Call to a Callee identified by the input argument CalleeID. Using this action, the TelCP configures a Call which can be video, audio or both. The input CallType argument represents the type of the Call and the quality of audio and video resolution. This action initiates the Call to the Callee but does not establish the Media Session between the TS and TC.

5.5.27.1 Arguments

Table 66 — Arguments for EnhancedInitiateCall()

Argument	Direction	relatedStateVariable
<u>CalleeID</u>	<u>IN</u>	<u>A_ARG_TYPE_CalleeID</u>
<u>CallType</u>	<u>IN</u>	<u>A_ARG_TYPE_CallType</u>
<u>CallID</u>	<u>OUT</u>	<u>A_ARG_TYPE_CallID</u>

5.5.27.2 Argument Descriptions

The input argument CalleeID contains the identity of the Callee.

The input argument CallType contains setup information of a Call established by the TelCP.

The output argument CallID is the return value of this action and uniquely identifies the Call. The CallManagement service assigns a unique CallID for every Call made through this service.

5.5.27.3 Service Requirements

The CallManagement service shall decide specific information, such as media format, resolution of video, or bitrate of voice, for the Call whose type is described in the CallType argument. The CallType does not specify the detailed information but indicates recommendation about the type of the Call.

The CallManagement service may event the CallInfo state variable for the status of the Call which is initiated by this action.

5.5.27.4 Control Point Requirements When Calling The Action

None

5.5.27.5 Dependency on Device State

None.

5.5.27.6 Effect on Device State

None.

5.5.27.7 Errors

Table 67 — Error Codes for EnhancedInitiateCall()

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
709	Invalid CalleeID	The specified <u>CalleeID</u> is invalid
716	Invalid CallType	The specified <u>CallType</u> is invalid

5.5.28 WaitingForCall()

This action allows a TS to wait for an incoming Call. If a TelCP invokes the WaitingForCall() action to the TS, the TS should be in the waiting process to accept a Call from a specific Caller identified by the input argument CallerID. The value of the input argument MaxWaitingTime determines the duration for the wait.

5.5.28.1 Arguments

Table 68 — Arguments for WaitingForCall()

Argument	Direction	relatedStateVariable
<u>CallerID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallerID</u>
<u>MaxWaitingTime</u>	<u>IN</u>	<u>A_ARG_TYPE_MaxWaitingTime</u>

5.5.28.2 Argument Descriptions

The input argument CallerID contains the identity of the Caller.

The input argument MaxWaitingTime contains maximum duration of the waiting time.

5.5.28.3 Service Requirements

If the WaitingForCall() action is invoked by a TelCP, the CallManagement service shall wait for the Parallel Call initiation during the time which is described in the MaxWaitingTime argument. If the service receives the initiation, it checks whether the identity of the Parallel Call initiation is the same as the value of the CallerID argument. If the identities are the same, the service accepts the Parallel Call automatically.

Once the CallManagement service automatically accepts a Parallel Call initiation, it shall send the CallInfo event with the value of the <callStatus> element set as "Connected".

5.5.28.4 Control Point Requirements When Calling The Action

When the TelCP receives the ParallelCallInfo event with the value of the <informationType> elements set as "ParallelCallRequest", the TelCP may discover the CallManagement service and invoke the WaitingForCall() to the service.

5.5.28.5 Dependency on Device State

None.

5.5.28.6 Effect on Device State

None.

5.5.28.7 Errors**Table 69 — Error Codes for WaitingForCall()**

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
710	Invalid CallerID	The specified <u>CallerID</u> is invalid

5.5.29 InitiateParallelCall()

This action allows a TelCP to request a TS to send Parallel Call request to the remote TS. The actual mechanism to communicate with the remote TS is out of scope of this specification. Once this action is invoked, the TS should send the Parallel Call setup message to the remote TS. The input argument of the action includes a URI of the device trying to establish the Parallel Call. The remote TS sends the ParallelCallInfo event to subscribed TelCP(s) in its network about this incoming request.

5.5.29.1 Arguments**Table 70 — Arguments for InitiateParallelCall()**

Argument	Direction	relatedStateVariable
<u>ParallelCallerID</u>	<u>IN</u>	<u>A_ARG_TYPE_CallerID</u>

5.5.29.2 Argument Descriptions

The input argument ParallelCallerID contains the identity of the device which is trying to establish a Parallel Call.

5.5.29.3 Service Requirements

If the first CallManagement service receives an InitiateParallelCall() action, the service shall send a request for Parallel Call Setup to the second CallManagement service which is connecting with first service through out of band mechanism. Once the request is received, the second service sends the ParallelCallInfo event with the value of the <informationType> element set as "ParallelCallRequest".

The CallManagement service shall store the information of the TelCP so that it can reply the Parallel Call Setup to the TelCP.

5.5.29.4 Control Point Requirements When Calling The Action

If the TelCP invokes the InitiateParallelCall() action, it shall wait for the ParallelCallInfo event and check the value of the <informationType> element of the event. If the value is set as "ParallelCallAccepted" the TelCP shall discover the CallManagement service and invoke the EnhancedInitiateCall() to the service.

5.5.29.5 Dependency on Device State

None.

5.5.29.6 Effect on Device State

None.

5.5.29.7 Errors**Table 71 — Error Codes for InitiateParallelCall()**

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.

5.5.30 AcceptParallelCall()

This action allows a TelCP to request a TS to send the response of an incoming Parallel Call request. The input argument of this action includes an URI of the terminal device that will receive the Parallel Call request.

5.5.30.1 Arguments**Table 72 — Arguments for AcceptParallelCall()**

Argument	Direction	relatedStateVariable
<u>ParallelCalleeID</u>	<u>IN</u>	<u>A_ARG_TYPE_CalleeID</u>

5.5.30.2 Argument Descriptions

The input argument ParallelCalleeID contains the identity of the receiver device which will receive the Parallel Call.

5.5.30.3 Service Requirements

If the service receives an AcceptParallelCall() action, it shall send a reply to establish Parallel Call Setup to another CallManagement service which has requested the Parallel Call Setup request.

Once received the reply the service sends the ParallelCallInfo event with the value of the <informationType> element set as "ParallelCallAccepted" to the TelCP which invoked the InitiateParallelCall() action.

5.5.30.4 Control Point Requirements When Calling The Action

When the TelCP receives the ParallelCallInfo event with the value of the <informationType> element set as "ParallelCallRequest" the TelCP shall discover a CallManagement service which will receive the Parallel Call and invokes the GetTelephonyIdentity() to the service. If the TelCP gets identity of the device, it shall invoke the AcceptParallelCall() to the service which has sent the ParallelCallInfo event.

5.5.30.5 Dependency on Device State

None.

5.5.30.6 Effect on Device State

None.

5.5.30.7 Errors**Table 73 — Error Codes for AcceptParallelCall()**

ErrorCode	errorDescription	Description
-----------	------------------	-------------

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
709	Invalid CalleeID	The specified <u>CalleeID</u> is invalid.

5.5.31 Relationships Between Actions

There is dependency between Call Status and invokable actions. (See 5.6)

5.5.32 Error Code Summary

Table 74 lists error codes common to actions for this service type. If an action results in multiple errors, the most specific error should be returned.

Table 74 — Error Code Summary

ErrorCode	errorDescription	Description
400-499	TBD	See Control clause in [1].
500-599	TBD	See Control clause in [1].
606	Action not Authorized	The CP does not have privileges to invoke this action.
700		Reserved for future extensions.
701	Invalid TelCP	The specified TelCPName or Secret Key is invalid.
702	TelCPName is already used	The specified NewTelCPName is already used.
703	Invalid CallID	The specified CallID is invalid.
704	Invalid peerID	The specified peerID is invalid
705	Invalid priority	The specified priority is invalid.
706	Invalid CallMode	The specified CallMode is invalid.
707	Invalid Media Capability	The specified Media Capability is invalid.
708	Busy	No resource to create a Call is left.
709	Invalid CalleeID	The specified CalleeID is invalid.
710	Invalid NewMonopolizer	The specified NewMonopolizer is invalid.
711	Invalid CallBackID	The specified CallBackID is invalid.
712	Invalid Call Status	The action is called when it is not allowed for the Call Status.
713	Unauthorized Identity	The identity(a TelCP or a user) is not authorized to invoke the action.
714	Identity does not exist	The identity of the TS has not been assigned yet by the telephony service in the WAN side.
715	Invalid Voice Mail ID	The ID of the voice mail does not exist
716	Invalid CallType	The specified CallType is invalid

Note: 800-899 Error Codes are not permitted for standard actions. See Control clause in [1] for more details.

5.6 Service Behavioral Model

5.6.1 State Diagram

Subclause 5.6.1 describes the state diagram of a Call which is managed by the CallManagement service. The Call Status may change when an action is invoked or when an event is received.

5.6.1.1 State Diagram for a Caller

Figure 11 illustrates the state diagram of a Call with respect to a Caller.

- When a Caller(TelCP) invokes the StartCall() action or the InitiateCall() action, a new instance of a Call is created and the status of the instance is set to as "Dialing". The TS then sends a request to the Callee to start a new Call.
- When the TS receives a provisional response from the Callee (i.e., the Callee has received the request but has not accepted the Call yet), the status of the Call is changed from "Dialing" to "Calling".
- When the TS receives the final response (i.e., the Callee has accepted the Call) from the Callee, the status is changed to "Connected".
- When a Caller(TelCP) invokes the StartMediaTransfer() action, the status is changed to "Talking".
- A Caller can invoke the StopCall() action at any time and the status of the Call is changed to "Disconnected" when this action is invoked.
- When the TS receives a response to reject the Call or a request to terminate the Call, the status is changed to "Disconnected".

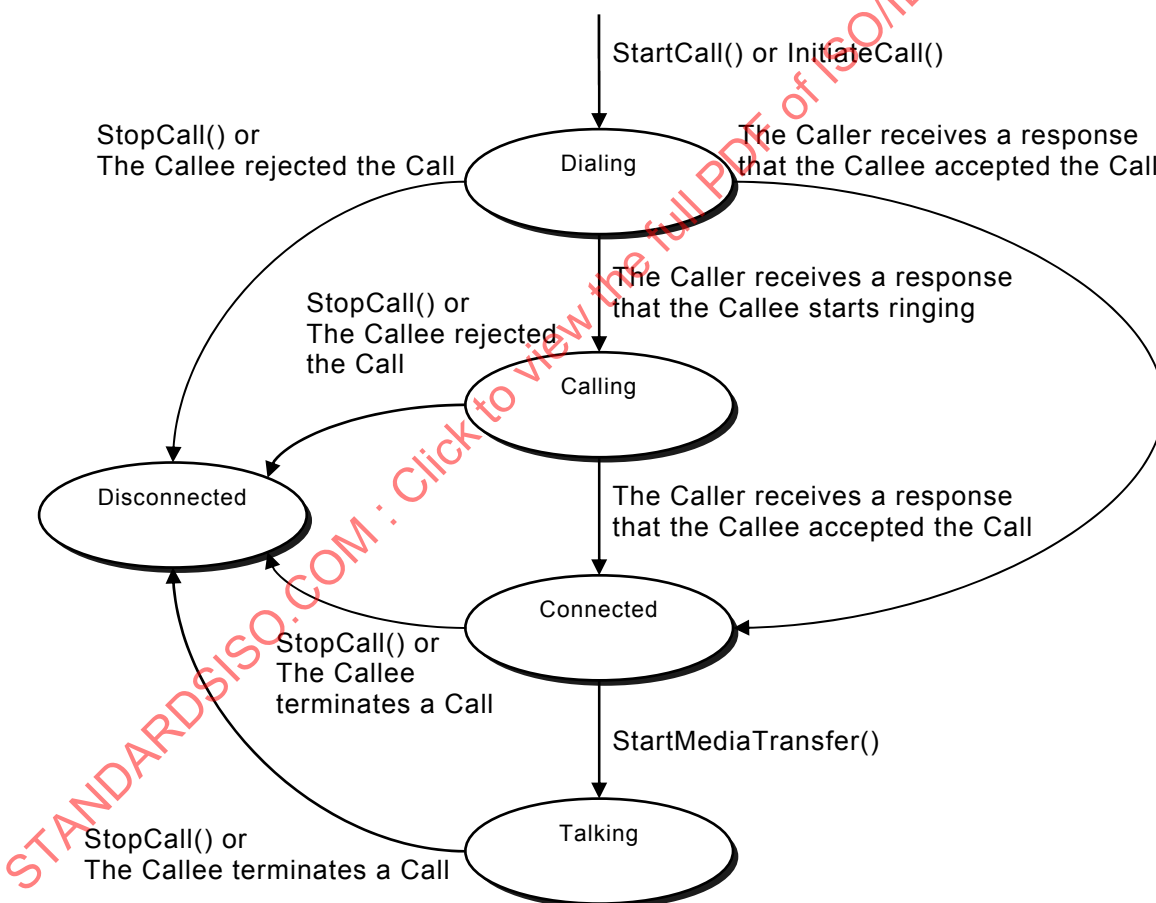


Figure 11 — Call State Diagram for a Caller

5.6.1.2 State Diagram for a Callee

Figure 12 illustrates the state diagram of a Call with respect to a Callee.

- When the TS receives an incoming Call, a new instance of a Call is created and the status of the instance is set to as "Ringing".

- b) When a Callee(TelCP) invokes the AcceptCall() action, the status of the Call is changed to "Connected".
- c) When a Callee(TelCP) invokes the StartMediaTransfer() action, the status of the Call is changed to "Talking".
- d) When a Callee(TelCP) invokes the RejectCall() action or the StopCall() action, the status of the Call is changed to "Disconnected".
- e) When the TS receives a response to cancel the Call or a request to terminate the Call, the status of the Call is changed to "Disconnected".

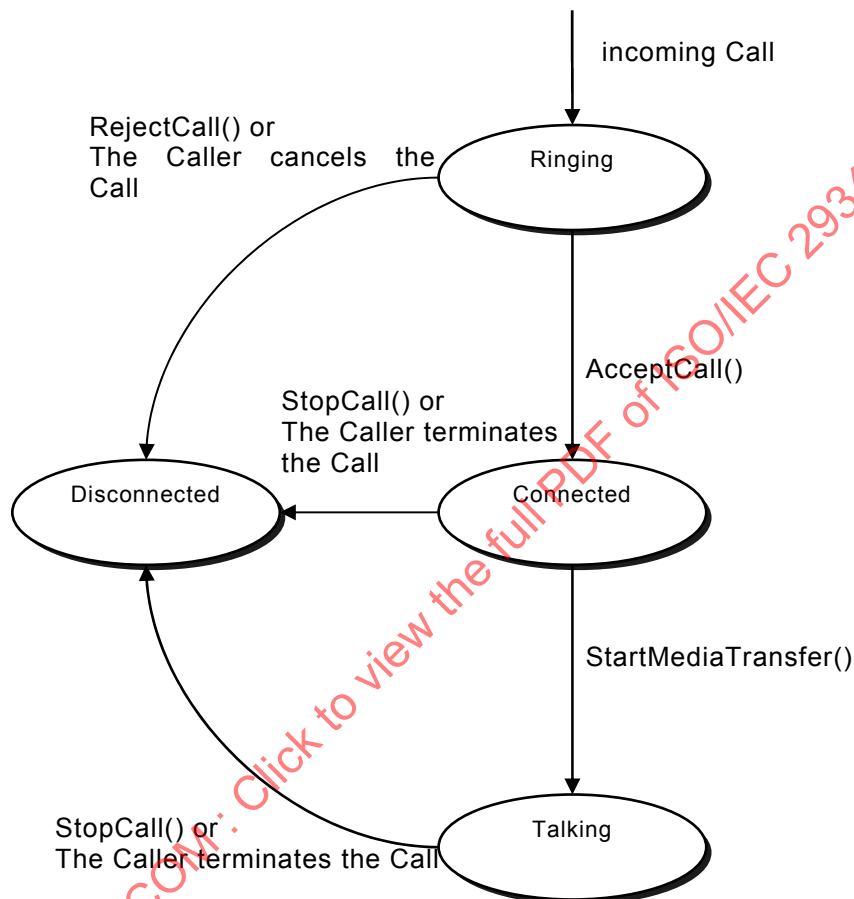


Figure 12 — Call State Diagram for a Callee

5.6.1.3 State Diagram to request the modification of a Call

When the Call Status is "Talking", a TelCP can modify the Media Capabilities of the Call. Figure 13 illustrates the state diagram of a Call when a TelCP requests to modify the Media Capabilities of the Call.

- a) When a TelCP invokes the ModifyCall() action, the status of the Call is changed to "SendingModifyRequest".
- b) When the TS receives a response to accept the request to modify the Call, the status of the Call is changed to "Modified".
- c) When the TS receives a response to reject the request to modify the Call, the status of the Call is changed to "ModifyFailed".
- d) When the status is "Modified" and a TelCP invokes the StartMediaTransfer() action, the status of the Call is changed to "Talking".

- e) When the status is "ModifyFailed", the status of the Call is automatically changed to "Talking".
- f) .When a TelCP invokes the StoptCall() action, the status of the Call is changed to "Disconnected".
- g) When the TS receives a request to terminate the Call, the status of the Call is changed to "Disconnected".

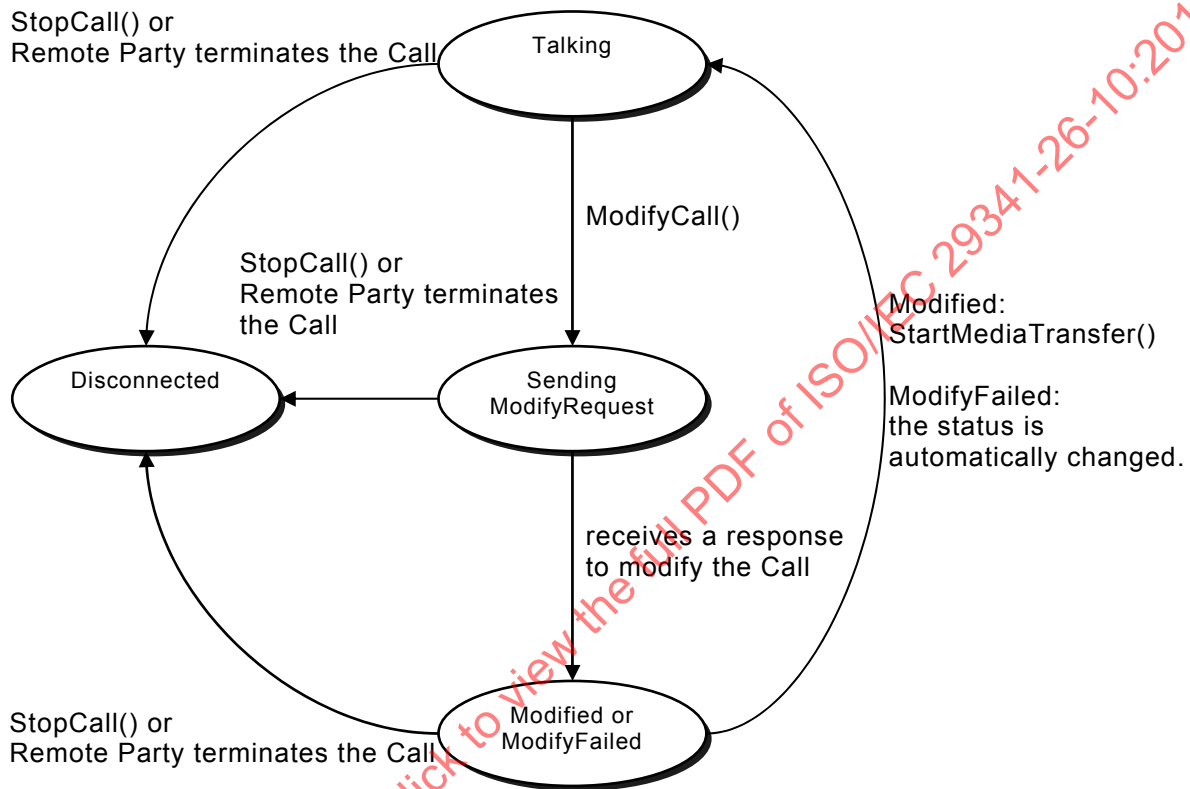


Figure 13 — Call State Diagram to request the modification of a Call

5.6.1.4 State Diagram to accept the modification of a Call

When the Call Status is "Talking" and the TS receives a request to modify the Media Capabilities of the Call, a TelCP can accept or reject the request. Figure 14 illustrates the state diagram of a Call when a TelCP accepts or rejects the request.

- a) When the TS receives a request to modify the Call, the status of the Call is changed to "ReceivingModifyRequest".
- b) When a TelCP invokes the AcceptModifyCall() action, the status of the Call is changed to "Modified".
- c) When a TelCP invokes the RejectCall() action, the status of the Call is changed to "ModifiedFailed".
- d) When the status is "Modified" and a TelCP invokes the StartMediaTransfer() action, the status of the Call is changed to "Talking".
- e) When the status is "ModifyFailed", the status of the Call is automatically changed to "Talking".
- f) When a TelCP invokes the StoptCall() action, the status of the Call is changed to "Disconnected".

- g) When the TS receives a request to terminate the Call, the status of the Call is changed to "Disconnected".

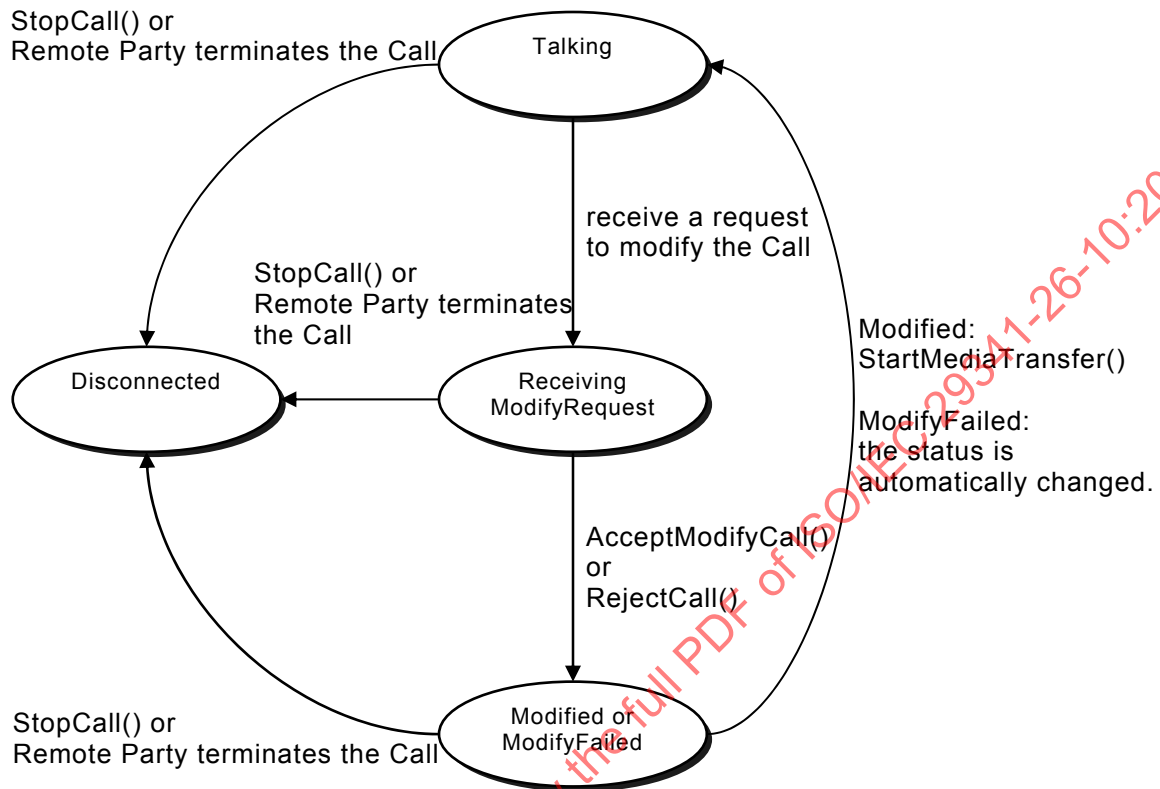


Figure 14 — Call State Diagram to accept the modification of a Call

6 XML Service Description

```

<?xml version="1.0"?>
<scpd xmlns="urn:schemas-upnp-org:service-1-0">

  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>

  <actionList>
    <action>
      <name>AcceptCall</name>
      <argumentList>
        <argument>
          <name>TelCPName</name>
          <direction>in</direction>
          <relatedStateVariable>
            A_ARG_TYPE_TelCPName
          </relatedStateVariable>
        </argument>
        <argument>
          <name>SecretKey</name>
          <direction>in</direction>
          <relatedStateVariable>

```

```

        A ARG TYPE SecretKey
        </relatedStateVariable>
    </argument>
    <argument>
        <name>TargetCallID</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE CallID
            </relatedStateVariable>
        </argument>
    <argument>
        <name>MediaCapabilityInfo</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE MediaCapabilityInfo
            </relatedStateVariable>
        </argument>
    <argument>
        <name>CallMode</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE CallMode
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>AcceptModifyCall</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE TelCPName
                </relatedStateVariable>
            </argument>
        <argument>
            <name>SecretKey</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE SecretKey
                </relatedStateVariable>
            </argument>
        <argument>
            <name>TargetCallID</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE CallID
                </relatedStateVariable>
            </argument>
        <argument>
            <name>MediaCapabilityInfo</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE MediaCapabilityInfo
                </relatedStateVariable>
            </argument>
    </argumentList>
</action>

```

```

<action>
  <name>ChangeMonopolizer</name>
  <argumentList>
    <argument>
      <name>CurrentMonopolizer</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPName
      </relatedStateVariable>
    </argument>
    <argument>
      <name>SecretKey</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE SecretKey
      </relatedStateVariable>
    </argument>
    <argument>
      <name>TargetCallID</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE CallID
      </relatedStateVariable>
    </argument>
    <argument>
      <name>NewMonopolizer</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPName
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>ChangeTelCPName</name>
  <argumentList>
    <argument>
      <name>CurrentTelCPName</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPName
      </relatedStateVariable>
    </argument>
    <argument>
      <name>CurrentSecretKey</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE SecretKey
      </relatedStateVariable>
    </argument>
    <argument>
      <name>NewTelCPName</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPName
      </relatedStateVariable>
    </argument>
    <argument>
      <name>NewSecretKey</name>
      <direction>out</direction>
    </argument>
  </argumentList>
</action>

```

```

        <relatedStateVariable>
            A ARG TYPE SecretKey
        </relatedStateVariable>
    </argument>
    <argument>
        <name>Expires</name>
        <direction>out</direction>
        <relatedStateVariable>
            A ARG TYPE Expires
        </relatedStateVariable>
    </argument>
</argumentList>
</action>

<action>
    <name>ClearCallBack</name>
    <argumentList>
        <argument>
            <name>CallBackID</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE CallBackID
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>ClearCallLogs</name>
</action>

<action>
    <name>GetCallBackInfo</name>
    <argumentList>
        <argument>
            <name>CallBackInfo</name>
            <direction>out</direction>
            <relatedStateVariable>
                A ARG TYPE CallBackInfoList
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>GetCallInfo</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE TelCPName
            </relatedStateVariable>
        </argument>
        <argument>
            <name>SecretKey</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE SecretKey
            </relatedStateVariable>
        </argument>
    </argumentList>

```

```

<argument>
  <name>TargetCallID</name>
  <direction>in</direction>
  <relatedStateVariable>
    A ARG TYPE CallID
  </relatedStateVariable>
</argument>
<argument>
  <name>CallInfoList</name>
  <direction>out</direction>
  <relatedStateVariable>
    A ARG TYPE CallInfoList
  </relatedStateVariable>
</argument>
</argumentList>
</action>

<action>
  <name>GetCallLogs</name>
  <argumentList>
    <argument>
      <name>CallLogs</name>
      <direction>out</direction>
      <relatedStateVariable>
        A ARG TYPE CallLogs
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>GetMediaCapabilities</name>
  <argumentList>
    <argument>
      <name>TCMediaCapabilityInfo</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE MediaCapabilityInfo
      </relatedStateVariable>
    </argument>
    <argument>
      <name>SupportedMediaCapabilityInfo</name>
      <direction>out</direction>
      <relatedStateVariable>
        A ARG TYPE MediaCapabilityInfo
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>GetTelephonyIdentity</name>
  <argumentList>
    <argument>
      <name>TelephonyIdentity</name>
      <direction>out</direction>
      <relatedStateVariable>
        A ARG TYPE TelephonyServierIdentity
      </relatedStateVariable>
    </argument>
  </argumentList>

```

```

</action>

<action>
  <name>GetTelCPNameList</name>
  <argumentList>
    <argument>
      <name>TelCPNameList</name>
      <direction>out</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPNameList
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>InitiateCall</name>
  <argumentList>
    <argument>
      <name>CalleeID</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE CalleeID
      </relatedStateVariable>
    </argument>
    <argument>
      <name>CallID</name>
      <direction>out</direction>
      <relatedStateVariable>
        A ARG TYPE CallID
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>ModifyCall</name>
  <argumentList>
    <argument>
      <name>TelCPName</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPName
      </relatedStateVariable>
    </argument>
    <argument>
      <name>SecretKey</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE SecretKey
      </relatedStateVariable>
    </argument>
    <argument>
      <name>TargetCallID</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE CallID
      </relatedStateVariable>
    </argument>
    <argument>
      <name>MediaCapabilityInfo</name>

```

```

        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE MediaCapabilityInfo
        </relatedStateVariable>
    </argument>
</argumentList>
</action>

<action>
    <name>RegisterCallBack</name>
    <argumentList>
        <argument>
            <name>CalleeID</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE CalleeID
            </relatedStateVariable>
        </argument>
        <argument>
            <name>CallBackID</name>
            <direction>out</direction>
            <relatedStateVariable>
                A ARG TYPE CallBackID
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>RegisterTelCPName</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE TelCPName
            </relatedStateVariable>
        </argument>
        <argument>
            <name>CurrentSecretKey</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE SecretKey
            </relatedStateVariable>
        </argument>
        <argument>
            <name>NewSecretKey</name>
            <direction>out</direction>
            <relatedStateVariable>
                A ARG TYPE SecretKey
            </relatedStateVariable>
        </argument>
        <argument>
            <name>Expires</name>
            <direction>out</direction>
            <relatedStateVariable>
                A ARG TYPE Expires
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

```

```

<action>
  <name>RejectCall</name>
  <argumentList>
    <argument>
      <name>TelCPName</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPName
      </relatedStateVariable>
    </argument>
    <argument>
      <name>SecretKey</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE SecretKey
      </relatedStateVariable>
    </argument>
    <argument>
      <name>TargetCallID</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE CallID
      </relatedStateVariable>
    </argument>
    <argument>
      <name>RejectReason</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE RejectReason
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>StartCall</name>
  <argumentList>
    <argument>
      <name>TelCPName</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPName
      </relatedStateVariable>
    </argument>
    <argument>
      <name>SecretKey</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE SecretKey
      </relatedStateVariable>
    </argument>
    <argument>
      <name>CalleeID</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE CalleeID
      </relatedStateVariable>
    </argument>
    <argument>
      <name>CallPriority</name>

```

```

    <direction>in</direction>
    <relatedStateVariable>
      A ARG TYPE CallPriority
    </relatedStateVariable>
  </argument>
  <argument>
    <name>MediaCapabilityInfo</name>
    <direction>in</direction>
    <relatedStateVariable>
      A ARG TYPE MediaCapabilityInfo
    </relatedStateVariable>
  </argument>
  <argument>
    <name>CallMode</name>
    <direction>in</direction>
    <relatedStateVariable>
      A ARG TYPE CallMode
    </relatedStateVariable>
  </argument>
  <argument>
    <name>CallID</name>
    <direction>out</direction>
    <relatedStateVariable>
      A ARG TYPE CallID
    </relatedStateVariable>
  </argument>
</argumentList>
</action>

<action>
  <name>StartMediaTransfer</name>
  <argumentList>
    <argument>
      <name>TelCPName</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TelCPName
      </relatedStateVariable>
    </argument>
    <argument>
      <name>SecretKey</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE SecretKey
      </relatedStateVariable>
    </argument>
    <argument>
      <name>TargetCallID</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE CallID
      </relatedStateVariable>
    </argument>
    <argument>
      <name>TCList</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE TCList
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

```

        <name>MediaCapabilityInfo</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE MediaCapabilityInfo
        </relatedStateVariable>
    </argument>
</argumentList>
</action>

<action>
    <name>StopCall</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE TelCPName
            </relatedStateVariable>
        </argument>
        <argument>
            <name>SecretKey</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE SecretKey
            </relatedStateVariable>
        </argument>
        <argument>
            <name>CallID</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE CallID
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>UnregisterTelCPName</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE TelCPName
            </relatedStateVariable>
        </argument>
        <argument>
            <name>SecretKey</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE SecretKey
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>ChangeCallMode</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>

```

```

        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE TelCPName
        </relatedStateVariable>
    </argument>
    <argument>
        <name>SecretKey</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE SecretKey
        </relatedStateVariable>
    </argument>
    <argument>
        <name>CallID</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE CallID
        </relatedStateVariable>
    </argument>
    <argument>
        <name>CallMode</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE CallMode
        </relatedStateVariable>
    </argument>
</argumentList>
</action>

<action>
    <name>GetPushInfo</name>
    <argumentList>
        <argument>
            <name>PushInfoList</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE PushInfoList
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>IgnoreCall</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE TelCPName
            </relatedStateVariable>
        </argument>
        <argument>
            <name>SecretKey</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE SecretKey
            </relatedStateVariable>
        </argument>
        <argument>
            <name>CallID</name>

```

```

        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE CallID
        </relatedStateVariable>
    </argument>
    <argument>
        <name>IgnoreReason</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE RejectReason
        </relatedStateVariable>
    </argument>
</argumentList>
</action>

<action>
    <name>GetVoiceMail</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE TelCPName
            </relatedStateVariable>
        </argument>
        <argument>
            <name>SecretKey</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE SecretKey
            </relatedStateVariable>
        </argument>
        <argument>
            <name>VoiceMailID</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE VoiceMailID
            </relatedStateVariable>
        </argument>
        <argument>
            <name>VoiceMailInfoList</name>
            <direction>out</direction>
            <relatedStateVariable>
                A ARG TYPE VoiceMailInfoList
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>DeleteVoiceMail</name>
    <argumentList>
        <argument>
            <name>TelCPName</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE TelCPName
            </relatedStateVariable>
        </argument>
        <argument>
            <name>SecretKey</name>

```

```

        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE SecretKey
        </relatedStateVariable>
    </argument>
    <argument>
        <name>VoiceMailID</name>
        <direction>in</direction>
        <relatedStateVariable>
            A ARG TYPE VoiceMailID
        </relatedStateVariable>
    </argument>
</argumentList>
</action>

<action>
    <name>EnhancedInitiateCall</name>
    <argumentList>
        <argument>
            <name>CalleeID</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE CalleeID
            </relatedStateVariable>
        </argument>
        <argument>
            <name>CallType</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE CallType
            </relatedStateVariable>
        </argument>
        <argument>
            <name>CallID</name>
            <direction>out</direction>
            <relatedStateVariable>
                A ARG TYPE CallID
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

<action>
    <name>WaitingForCall</name>
    <argumentList>
        <argument>
            <name>CallerID</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE CallerID
            </relatedStateVariable>
        </argument>
        <argument>
            <name>MaxWaitingTime</name>
            <direction>in</direction>
            <relatedStateVariable>
                A ARG TYPE MaxWaitingTime
            </relatedStateVariable>
        </argument>
    </argumentList>
</action>

```

```

<action>
  <name>InitiateParallelCall</name>
  <argumentList>
    <argument>
      <name>ParallelCallerID</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE CallerID
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>AcceptParallelCall</name>
  <argumentList>
    <argument>
      <name>ParallelCalleeID</name>
      <direction>in</direction>
      <relatedStateVariable>
        A ARG TYPE CalleeID
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

</actionList>

<serviceStateTable>

  <stateVariable sendEvents="no">
    <name>A ARG TYPE TelephonyServerIdentity</name>
    <dataType>string</dataType>
  </stateVariable>

  <stateVariable sendEvents="no">
    <name>A ARG TYPE CallBackID</name>
    <dataType>string</dataType>
    <allowedValueList>
      <allowedValue>Busy</allowedValue>
      <allowedValue>Not Acceptable Capability</allowedValue>
      <allowedValue>Rejected</allowedValue>
      <allowedValue>Cancelled</allowedValue>
    </allowedValueList>
  </stateVariable>

  <stateVariable sendEvents="no">
    <name>A ARG TYPE CalleeID</name>
    <dataType>string</dataType>
  </stateVariable>

  <stateVariable sendEvents="yes">
    <name>TelCPNameList</name>
    <dataType>string</dataType>
  </stateVariable>

  <stateVariable sendEvents="no">
    <name>A ARG TYPE CallLogs</name>
    <dataType>string</dataType>
  </stateVariable>

```

```

<stateVariable sendEvents="no">
  <name>A ARG TYPE CallbackInfoList</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE SecretKey</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE MediaCapabilityInfo</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE TCLList</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE CallInfoList</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE TelCPNameList</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="yes">
  <name>CallInfo</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE Expires</name>
  <dataType>i4</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE TelCPName</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE CallID</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="yes">
  <name>CallbackAvailability</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE CallMode</name>
  <dataType>string</dataType>
</stateVariable>

```

```

<stateVariable sendEvents="no">
  <name>A ARG TYPE RejectReason</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE CallPriority</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="yes">
  <name>PushInfo</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="yes">
  <name>VoiceMailInfo</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="yes">
  <name>ParallelCallInfo</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE PushInfoList</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE VoiceMailInfoList</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE VoiceMailID</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE CallType</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE CallerID</name>
  <dataType>string</dataType>
</stateVariable>

<stateVariable sendEvents="no">
  <name>A ARG TYPE MaxWaitingTime</name>
  <dataType>i4</dataType>
</stateVariable>
</serviceStateTable>
</scpd>

```

Annex A (normative)

XML complex type *peerType*

A communication means the exchange of an information between two or more end entities. These end entities are herein referred as Peers. The Peer can be a caller of a phone call, recipient of an email message, or group of participants in a communication session, or a contact in an Address book.

In order to have a uniform representation of a Peer across all the services in the UPnP Telephony, the XML complex type *peerType* is defined. The same XML complex type can be reused by other UPnP Telephony services.

The complex type *peerType* contains the information to properly identify a contact and its communication address for e.g. a phone call needs a telephone number, an email message needs an email address etc. Along with the communication address it is also important to include additional information about the Peer for e.g. photo, location information of user etc. If TS supports the PhoneManagement profile, then the correspondence between the Peer element and either a contact or a group of contacts in the Address book is also included in the complex *peerType* element.

A.1 Using the *peerType* within XML Schemas

The complex type *peerType* can be used in the XML schemas by including the following statement:

```
<import
  namespace="urn:schemas-upnp-org:phone:peer"
  schemaLocation="http://www.upnp.org/schemas/phone/peer-v2.xsd"/>
```

where the `schemaLocation` refers to the last updated schema file for the *Peer*.

A.2 Description of fields of a *peerType* complex type

Clause A.2 gives a description of the elements defined in the *peerType* complex type.

```
<?xml version="1.0" encoding="UTF-8"?>
<peer:peer
  xsi:schemaLocation="urn:schemas-upnp-org:phone:peer
    http://www.upnp.org/schemas/phone/peer-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <peer:id>
    The identifier of the Peer (e.g., phone number, etc.)
  </peer:id>
  <peer:name>A user friendly name for the Peer</peer:name>
  <peer:contactInstanceId>
    The instance identifier for a contact referred by the Peer
  </peer:contactInstanceId>
  <peer:groupInstanceId>
    The instance identifier for a group referred by the Peer
  </peer:groupInstanceId>
  <peer:image type="URL"/"thumbnail">
    <peer:contentType>
      content type information for the image data
    </peer:contentType>
    <peer:contentTransferEncoding>
      Encoding information
    </peer:contentTransferEncoding>
    <peer:imageData>actual image information</peer:imageData>
  </peer:image>
  <peer:location type="map"/"coordinates">
```

Either a URL points to the map information; URL for a the map image OR it could include longitude and latitude information in ["LAT LON"] order.

```
</peer:location>
</peer:peer>
```

id

Required, xsd:string. Indicates the communication address or the identifier for the Peer (e.g., a telephone number, an e-mail address, an identifier of a group of contacts, etc).

name

Allowed, xsd:string. Indicates a user friendly name for the Peer.

contactInstanceId

Allowed, xsd:unsignedInt. Is the instance identifier of the contact present in the *Phone Data Model's Address Book* for the referenced *Peer*. The value of the contactInstanceId is an unsigned integer. If there is no Instance in the *Address Book* for the referenced *Peer*, then the contactInstanceId value shall be 0 (no match with the list of contacts in the *Address Book*). If the *PhoneManagement* profile is not supported or the relationship between the *Address Book* and the *Peer* is not used by the service, then this element shall not be used. The contactInstanceId and groupInstanceId are mutually exclusive elements.

groupInstanceId

Allowed, xsd:unsignedInt. Is the instance identifier of a group present in the *Phone Data Model's Address Book* for the referenced *Peer*. The value of groupInstanceId is an unsigned integer. If there is no Instance in the *Address Book* for this referenced *Peer*, then the groupInstanceId value shall be 0 (no match with the list of groups in the *Address Book*). If the *PhoneManagement* profile is not supported or the relationship between the *Address Book* and the *Peer* is not used by the service, then this element shall not be used. The contactInstanceId and groupInstanceId are mutually exclusive elements.

image

Allowed, This element represents the image information for the contact. The image can be represented as an URL pointing to the image or small thumbunail image data information. This element shall include following attributes and elements.

type

Allowed, xsd:string. This attribute indicates how image information of the contact is represented. The image information can be represented either an URL to the image or actual thumbnail image data. This attribute can have value either "URL" or "thumbnail".

contentType

Allowed, xsd:string. This element indicates MIME type information for the image as defined in [11]. This element should be present if type attribute is set to "thumbnail".

contentTransferEncoding

Allowed, xsd:string. This element indicates encoding mechanism for the image data as defined in [11]. This element should be present if type attribute is set to "thumbnail".

ImageData

Allowed, xsd:string. This element carries actual image information either as an URL to the image or actual image encoded information. If type attribute is set to "URL" then this element carries an URL to the image else type attribute is set to "Thumbnail" then it carries an actual encoded image information.

location

Allowed, xsd:string. This element carries the location information of the contact. The location information is represented either in longitude and latitude format or as an URL pointing to map information which includes location of the contact. The URL can also point to the map image. The information in this element is interpreted based on the type attribute of this element.

type

Allowed, xsd:string. This attribute indicates how to interpret the location information. If the location information is to be represented as a map information or as a map image, then this attribute is set to "map" and location element wll carry an URL of the map information. Else it is set to "coordinates" to represent the location information in [latitude, Longitude] format.

any

Allowed. Attachment point for custom extensions.

A.3 peerType Schema

The following XML schema defines the peerType complex type.

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:tns="urn:schemas-upnp-org:phone:peer"
targetNamespace="urn:schemas-upnp-org:phone:peer" elementFormDefault="qualified"
attributeFormDefault="qualified" version="2">
  <complexType name="peerType">
    <sequence>
      <element name="id" type="tns:id" nillable="0"/>
      <element name="name" type="string" nillable="0" minOccurs="0">
        <annotation>
          <documentation>Textual name of the peer. In case the Phone Data Model is supported,
this element shall be the FormattedName in the address book.</documentation>
        </annotation>
      </element>
      <choice minOccurs="0">
        <element name="contactInstanceId" type="tns:contactInstanceId" nillable="0"/>
        <element name="groupInstanceId">
          <annotation>
            <documentation>The Instance Identifier of a Group in the PDM address
book.</documentation>
          </annotation>
        </element>
      </choice>
      <element name="image" minOccurs="0">
        <complexType>
          <sequence>
            <element name="contentType" type="string" minOccurs="0"/>
            <element name="contentTransferEncoding" type="string" minOccurs="0"/>
            <element name="imageData" type="string" minOccurs="0"/>
          </sequence>
          <attribute name="type" use="optional">
            <simpleType>
              <restriction base="string">
                <enumeration value="URL"/>
                <enumeration value="thumbnail"/>
              </restriction>
            </simpleType>
          </attribute>
        </complexType>
      </element>
      <element name="location" minOccurs="0">
        <annotation>
          <documentation>Either a URL points to the map information; URL for a the map image
OR it could include longitude and latitude information in ["LAT LON"] order.</documentation>
        </annotation>
        <complexType>
          <simpleContent>
            <extension base="string">
              <attribute name="type" use="optional" default="map">
                <simpleType>
                  <restriction base="string">
                    <enumeration value="map"/>
                    <enumeration value="coordinates"/>
                  </restriction>
                </simpleType>
              </attribute>
            </extension>
          </simpleContent>
        </complexType>
      </element>
      <any namespace="##other" minOccurs="0">
        <annotation>
          <documentation>Vendor defined extensions attachment point.</documentation>
        </annotation>
      </any>
    </sequence>
  </complexType>
</schema>
```

ISO/IEC 29341-26-10:2017(E)

```
<restriction base="string"/>
</simpleType>
<simpleType name="contactInstanceId">
  <restriction base="unsignedInt"/>
</simpleType>
</schema>
```

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 29341-26-10:2017

Annex B (normative)

XML Schema

Annex B provides the global XML Schema for syntactical validation of all the XML fragments used in the CallManagement service.

```
<?xml version="1.0" encoding="utf-8"?>
<schema xmlns:cams="urn:schemas-upnp-org:phone:cams" xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:peer="urn:schemas-upnp-org:phone:peer" targetNamespace="urn:schemas-upnp-org:phone:cams"
elementFormDefault="qualified" attributeFormDefault="qualified">
  <import namespace="urn:schemas-upnp-org:phone:peer"
schemaLocation="http://www.upnp.org/schemas/phone/peer-v2.xsd"/>
  <simpleType name="targetNamesType">
    <restriction base="string">
      <enumeration value="TelCPName"/>
      <enumeration value="TelCPID"/>
      <enumeration value="Username"/>
      <enumeration value="*" />
    </restriction>
  </simpleType>
  <simpleType name="callStatusValueType">
    <restriction base="string">
      <enumeration value="Ringing"/>
      <enumeration value="Dialing"/>
      <enumeration value="Calling"/>
      <enumeration value="Connected"/>
      <enumeration value="Talking"/>
      <enumeration value="Disconnected"/>
      <enumeration value="SendingModifyRequest"/>
      <enumeration value="ReceivingModifyRequest"/>
      <enumeration value="Modified"/>
      <enumeration value="ModifyFailed"/>
    </restriction>
  </simpleType>
  <simpleType name="callStatusReasonType">
    <restriction base="string">
      <enumeration value="Accepted by TS"/>
      <enumeration value="Initiated by TS"/>
      <enumeration value="Busy"/>
      <enumeration value="Not Acceptable Capability"/>
      <enumeration value="Not Found"/>
      <enumeration value="Rejected by TS"/>
      <enumeration value="Rejected"/>
      <enumeration value="Cancelled"/>
      <enumeration value="Receiving Early Media Response"/>
      <enumeration value="Early Media Started"/>
    </restriction>
  </simpleType>
  <simpleType name="priorityValueType">
    <restriction base="string">
      <enumeration value="Emergency"/>
      <enumeration value="Normal"/>
    </restriction>
  </simpleType>
  <simpleType name="mediaCapabilityFormatType">
    <restriction base="string">
      <enumeration value="SDP"/>
    </restriction>
  </simpleType>
  <complexType name="callBackInfoType">
    <sequence>
      <element name="callBackID" type="string"/>
      <element name="callee" type="peer:peerType"/>
    </sequence>
  </complexType>
  <complexType name="TCListType">
    <sequence>
      <element name="TC" form="unqualified" minOccurs="0" maxOccurs="unbounded">
        <complexType>
          <sequence>
            <element name="UDN" type="string" form="unqualified" minOccurs="0"/>
          </sequence>
        </complexType>
      </element>
    </sequence>
  </complexType>
```

```

        <element name="mediaSessionID" type="string" form="unqualified" minOccurs="0"/>
      </sequence>
    </complexType>
  </element>
</sequence>
</complexType>
<complexType name="callInfoType">
  <sequence>
    <element name="callID" type="string" form="unqualified"/>
    <element name="targetNames" form="unqualified" nillable="true">
      <complexType>
        <simpleContent>
          <extension base="string">
            <attribute name="type" type="cams:targetNamesType" use="required"
form="unqualified"/>
          </extension>
        </simpleContent>
      </complexType>
    </element>
    <element name="callStatus" form="unqualified" nillable="true">
      <complexType>
        <simpleContent>
          <extension base="cams:callStatusValueType">
            <attribute name="reason" type="cams:callStatusReasonType" use="optional"
form="unqualified"/>
          </extension>
        </simpleContent>
      </complexType>
    </element>
    <element name="priority" type="cams:priorityValueType" form="unqualified" minOccurs="0"/>
    <element name="remoteParty" type="peer:peerType" form="unqualified" minOccurs="0"
maxOccurs="unbounded"/>
    <element name="TCLList" type="cams:TCLListType" form="unqualified" minOccurs="0"/>
    <element name="enableMediaMixing" type="integer" form="unqualified" minOccurs="0"/>
    <element name="shareMedia" type="string" form="unqualified" minOccurs="0"/>
    <element name="mediaCapability" form="unqualified" nillable="true" minOccurs="0">
      <complexType>
        <simpleContent>
          <extension base="string">
            <attribute name="format" type="cams:mediaCapabilityFormatType" use="optional"
form="unqualified"/>
          </extension>
        </simpleContent>
      </complexType>
    </element>
    <element name="nativeMediaCapability" form="unqualified" nillable="true" minOccurs="0">
      <complexType>
        <simpleContent>
          <extension base="string">
            <attribute name="format" type="string" use="optional" form="unqualified"/>
          </extension>
        </simpleContent>
      </complexType>
    </element>
  </sequence>
</complexType>
<complexType name="contentSharingCapability">
  <sequence>
    <element name="protocolInfo" type="string" minOccurs="0"/>
    <element name="URI" type="anyURI" minOccurs="0"/>
    <element name="sharingType" minOccurs="0">
      <simpleType>
        <restriction base="string">
          <enumeration value="transfer"/>
          <enumeration value="streaming"/>
        </restriction>
      </simpleType>
    </element>
  </sequence>
</complexType>
<simpleType name="pushInfoPriorityValueType">
  <restriction base="string">
    <enumeration value="High"/>
    <enumeration value="Normal"/>
    <enumeration value="Low"/>
  </restriction>

```

```

</simpleType>
<complexType name="pushInfoType">
  <sequence>
    <element name="priority" type="cams:pushInfoPriorityValueType"/>
    <element name="summary" type="string"/>
    <element name="url" type="anyURI"/>
    <element name="import" type="cams:import" minOccurs="0"/>
  </sequence>
</complexType>
<complexType name="import">
  <simpleContent>
    <extension base="string">
      <attribute name="format"/>
    </extension>
  </simpleContent>
</complexType>
<complexType name="voiceMailType">
  <sequence>
    <element name="voiceMailID" type="string"/>
    <element name="callee" type="string" minOccurs="0"/>
    <element name="url" type="anyURI" minOccurs="0"/>
    <element name="duration" type="string" minOccurs="0"/>
    <element name="time" type="string" minOccurs="0"/>
    <!-- Any other envelope information (if any) go here.-->
  </sequence>
</complexType>
<simpleType name="videoQualityValueType">
  <restriction base="string">
    <enumeration value="HD"/>
    <enumeration value="SD"/>
  </restriction>
</simpleType>
<complexType name="videoType">
  <sequence>
    <element name="videoQuality" type="cams:videoQualityValueType"/>
  </sequence>
</complexType>
<simpleType name="audioQualityValueType">
  <restriction base="string">
    <enumeration value="SWB"/>
    <enumeration value="WB"/>
    <enumeration value="NB"/>
  </restriction>
</simpleType>
<complexType name="audioType">
  <sequence>
    <element name="audioQuality" type="cams:audioQualityValueType"/>
  </sequence>
</complexType>
<simpleType name="parallelCallInformationValueType">
  <restriction base="string">
    <enumeration value="ParallelCallRequest"/>
    <enumeration value="ParallelCallAccepted"/>
    <enumeration value="ParallelCallDenied"/>
  </restriction>
</simpleType>
<complexType name="remoteParty">
  <sequence>
    <element name="id" type="peer:id"/>
  </sequence>
</complexType>
<element name="callInfo" type="cams:callInfoType"/>
<element name="telCPNameList" type="string"/>
<element name="callBackAvailability">
  <complexType>
    <choice maxOccurs="unbounded">
      <element name="callBack" form="unqualified">
        <complexType>
          <sequence>
            <element name="callBackID" type="string" form="unqualified"/>
          </sequence>
        </complexType>
      </element>
    </choice>
  </complexType>
</element>

```

ISO/IEC 29341-26-10:2017(E)

```
<element name="pushInfo" type="cams:pushInfoType"/>
<element name="voiceMailInfo" type="cams:voiceMailType"/>
<element name="parallelCallInfo">
  <complexType>
    <sequence>
      <element name="informationType" type="cams:parallelCallInformationValueType"/>
      <element name="remoteParty" type="cams:remoteParty" minOccurs="0"/>
    </sequence>
  </complexType>
</element>
<element name="telephonyServerIdentity" type="string"/>
<element name="telCPName" type="string"/>
<element name="expires" type="integer"/>
<element name="mediaCapabilityInfo">
  <complexType>
    <sequence>
      <element name="mediaMixingCapability" type="string" form="unqualified" minOccurs="0"/>
      <element name="mediaSharingCapability" type="string" form="unqualified" minOccurs="0"/>
      <element name="enableMediaMixing" type="integer" form="unqualified" minOccurs="0"/>
      <element name="shareMedia" type="string" form="unqualified" minOccurs="0"/>
      <element name="mediaCapability" form="unqualified" nillable="true" minOccurs="0">
        <complexType>
          <simpleContent>
            <extension base="string">
              <attribute name="format" type="cams:mediaCapabilityFormatType" use="optional"
form="unqualified"/>
            </extension>
          </simpleContent>
        </complexType>
      </element>
      <element name="nativeMediaCapability" form="unqualified" nillable="true" minOccurs="0">
        <complexType>
          <simpleContent>
            <extension base="string">
              <attribute name="format" type="cams:mediaCapabilityFormatType" use="optional"
form="unqualified"/>
            </extension>
          </simpleContent>
        </complexType>
      </element>
      <element name="contentSharingCapability" type="cams:contentSharingCapability"
minOccurs="0"/>
    </sequence>
  </complexType>
</element>
<element name="calleeID" type="string"/>
<element name="callPriority" type="cams:priorityValueType"/>
<element name="callMode">
  <simpleType>
    <restriction base="string">
      <enumeration value="PHONE-TelCP"/>
      <enumeration value="DP-TelCP"/>
      <enumeration value="DP-User"/>
      <enumeration value="Non-Monopolize"/>
    </restriction>
  </simpleType>
</element>
<element name="callID" type="string"/>
<element name="secretKey" type="string"/>
<element name="rejectReason" type="string"/>
<element name="TCList" type="cams:TCListType"/>
<element name="callInfoList">
  <complexType>
    <choice minOccurs="0" maxOccurs="unbounded">
      <element name="callInfo" type="cams:callInfoType" form="unqualified"/>
    </choice>
  </complexType>
</element>
<element name="callLogs">
  <complexType>
    <choice minOccurs="0" maxOccurs="unbounded">
      <element name="call" form="unqualified">
        <complexType>
          <sequence>
            <element name="type" type="string" form="unqualified"/>
          </sequence>
        </complexType>
      </element>
    </choice>
  </complexType>
</element>
```

```

        <element name="caller" type="peer:peerType" form="unqualified"
maxOccurs="unbounded"/>
        <element name="callee" type="peer:peerType" form="unqualified"
maxOccurs="unbounded"/>
        <element name="startDateTime" type="string" form="unqualified"/>
        <element name="duration" type="string" form="unqualified" minOccurs="0"/>
    </sequence>
    <attribute name="direction" type="string" form="unqualified"/>
    <attribute name="success" type="string" form="unqualified"/>
</complexType>
</element>
</choice>
</complexType>
</element>
<element name="callBackID" type="string"/>
<element name="callBackInfoList">
    <complexType>
        <choice minOccurs="0" maxOccurs="unbounded">
            <element name="callBackInfo" type="cams:callBackInfoType" form="unqualified"/>
        </choice>
    </complexType>
</element>
<element name="pushInfoList ">
    <complexType>
        <choice minOccurs="0" maxOccurs="unbounded">
            <element name="pushInfo" type="cams:pushInfoType"/>
        </choice>
    </complexType>
</element>
<element name="voiceMailInfoList">
    <complexType>
        <choice minOccurs="0" maxOccurs="unbounded">
            <element name="voiceMail" type="cams:voiceMailType"/>
        </choice>
    </complexType>
</element>
<element name="voiceMailID" type="string"/>
<element name="callType">
    <complexType>
        <sequence>
            <element name="video" type="cams:videoType" minOccurs="0"/>
            <element name="audio" type="cams:audioType" minOccurs="0"/>
        </sequence>
    </complexType>
</element>
<element name="callerID" type="string"/>
<element name="maxWaitingTime" type="string"/>
</schema>

```

Annex C (informative)

Theory of Operation

C.1 TC-Based Media Handling

C.1.1 Create an outgoing Call

The following sequence describes how to create an outgoing Call in the Non-Monopolization Mode using TC-Based Media Handling.

- a) The TelCP determines the Media Capabilities of the TS by invoking the GetMediaCapabilities() action.
- b) The TelCP determines the Media Capabilities of the TC by invoking the GetMediaCapabilities() action.
- c) The TelCP selects the set of appropriate Media Capabilities for the Call by matching the Media Capabilities of the TS and the TC.
- d) The TelCP invokes the StartCall() action on the TS and the TS initiates the Call on the WAN side.
- e) The TS sends the CallInfo state variable to the TelCP about the state of the Call (“Dialing”, “Calling” and “Connected”). The Call between the TS and the WAN side is established at this point.
- f) The TelCP invokes the StartMediaSession() action on the TC after the TelCP receives the CallInfo state variable with the Call Status as “Connected”. The TC starts sending and/or receiving the Media Streams.
- g) The TC sends the MediaSessionInfo state variable to the TelCP with the status of the Media Session as “Started”.
- h) The TelCP invokes the StartMediaTransfer() action on the TS. The TS starts sending and/or receiving the Media Streams.
- i) The TS sends the CallInfo state variable to the TelCP with the CallStatus as “Talking”.
- j) If the TelCP wants to disconnect the Call, the TelCP invokes the StopCall() action on the TS.
- k) The TelCP then invokes the StopMediaSession() action on the TC.
- l) The TS sends the CallInfo state variable to the TelCP with the CallStatus as “Disconneted” and the TC sends the MediaSessionInfo state variable to the TelCP with the status of the Media Session as “Stopped”.

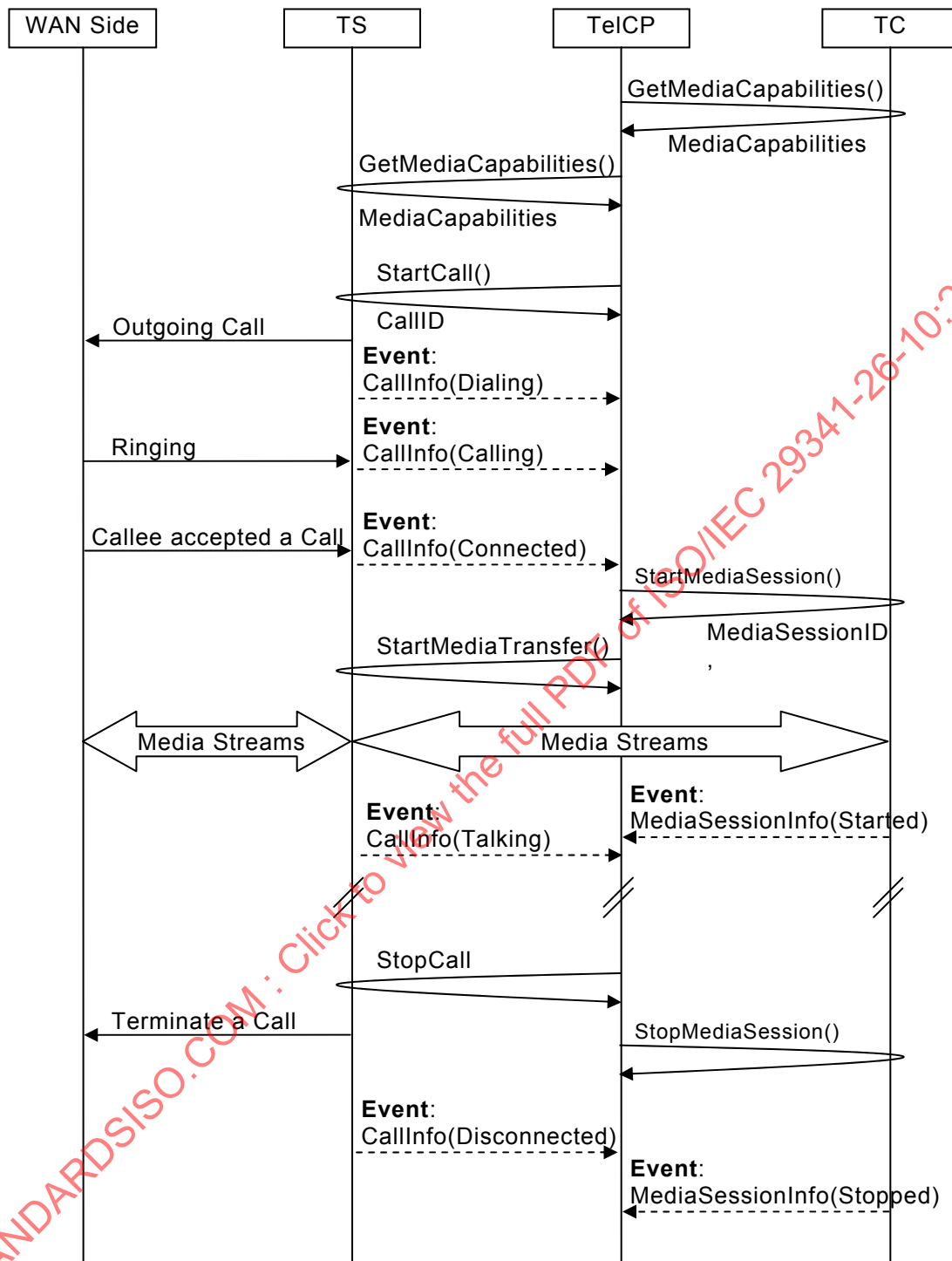


Figure C.1 — Flow basics for creating an outgoing Call (TC-Based Media Handling)

C.1.2 Accept an incoming Call

The following sequence describes how to accept an incoming Call in the Non-Monopolization Mode using TC-Based Media Handling.

- When the TS receives an incoming Call on the WAN side, the TS sends the *CallInfo* state variable to the TelCP with the Call Status as "*Ringing*". The *CallInfo* state variable also includes the Media Capabilities of the Caller.

ISO/IEC 29341-26-10:2017(E)

- b) The TelCP determines the Media Capabilities of the TC by invoking the GetMediaCapabilities() action.
- c) The TelCP selects the set of appropriate Media Capabilities for the Call by matching the Media Capabilities of the TS and the TC.
- d) The TelCP invokes the AcceptCall() action on the TS and the TS accepts the Call on the WAN side.
- e) After the Call between the TS and the WAN side is established, the TS sends the CallInfo state variable to the TelCP with the Call Status as "Connected".
- f) The TelCP invokes the StartMediaSession() action on the TC after the TelCP receives the CallInfo state variable with the Call Status as "Connected". The TC starts sending and/or receiving the Media Streams.
- g) The TC sends the MediaSessionInfo state variable to the TelCP with the status of the Media Session as "Started".
- h) The TelCP invokes the StartMediaTransfer() action on the TS. The TS starts sending and/or receiving the Media Streams.
- i) The TS sends the CallInfo state variable to the TelCP with the Call Status as "Talking".
- j) If the TelCP wants to disconnect the Call, the TelCP invokes the StopCall() action on the TS.
- k) The TelCP then invokes the StopMediaSession() action on the TC.
- l) The TS sends the CallInfo state variable to the TelCP with the Call Status as "Disconnected" and the TC sends the MediaSessionInfo state variable to the TelCP with the status of the Media Session as "Stopped".

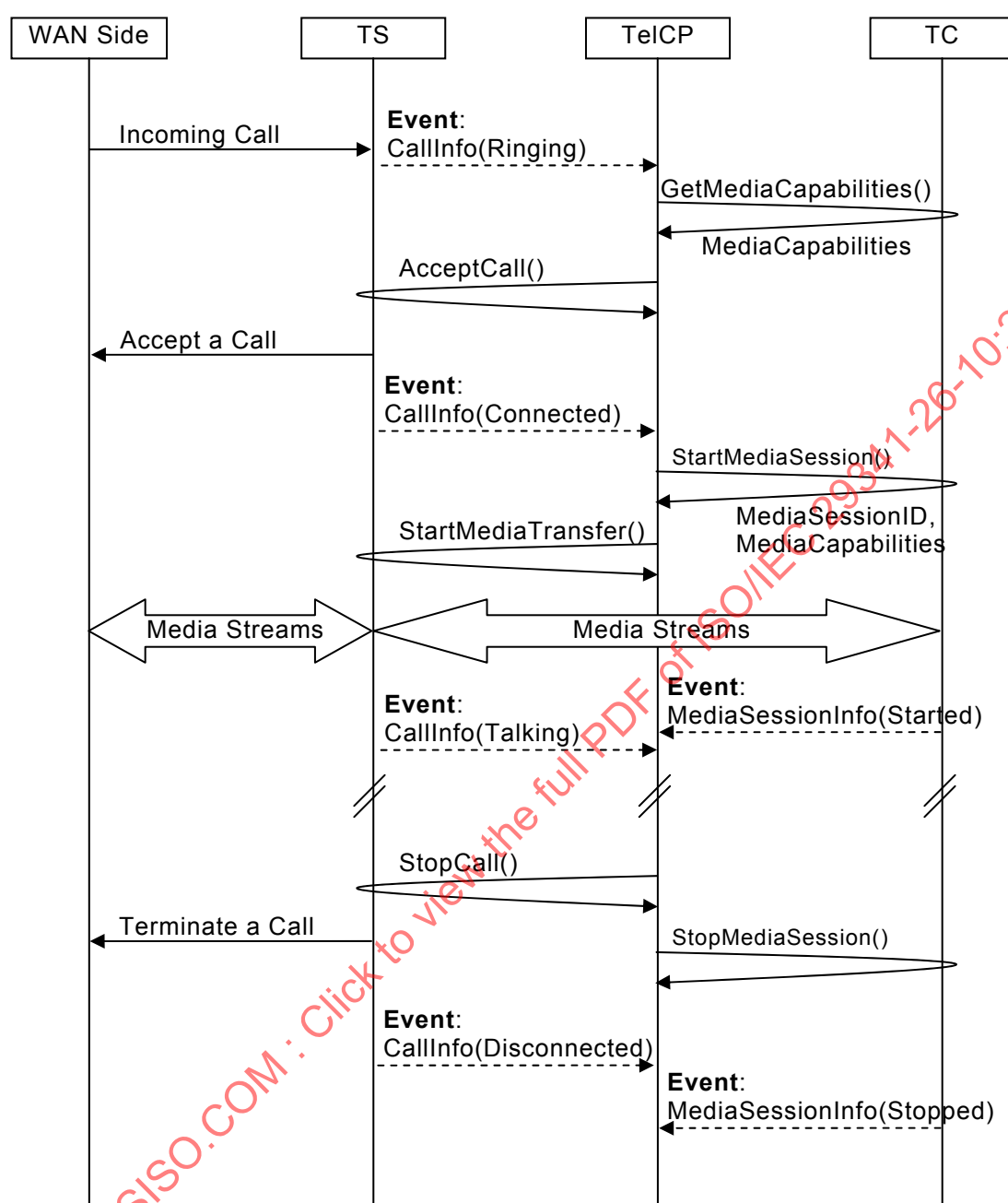


Figure C.2 — Flow basics for accepting an incoming Call (TC-Based Media Handling)

C.1.3 Modify an ongoing Call

The following sequence describes how to modify an ongoing Call from an audio to a video Call in the Non-Monopolization Mode using TC-Based Media Handling. In this sequence, the TC1 supports the audio capabilities and the TC2 supports the video capabilities. At first, the TelCP creates an audio Call with TC1. Then the TelCP modifies the Call from audio to a video Call by adding TC2.

- The TelCP creates an audio Call with the TC1. The sequence is the same as the sequence shown in C.1.1.
- The TelCP determines the Media Capabilities of the TS by invoking the GetMediaCapabilities() action.
- The TelCP determines the Media Capabilities of the TC2 by invoking the GetMediaCapabilities() action.

ISO/IEC 29341-26-10:2017(E)

- d) The TelCP selects the set of appropriate Media Capabilities for the Call to change the Media Capabilities from an audio to a video Call by matching the Media Capabilities of the TS and the TC1 and TC2.
- e) The TelCP invokes the ModifyCall() action on the TS and the TS modifies the Call on the WAN side.
- f) The TS sends the CallInfo state variable to the TelCP about the state of the Call ("SendingModifyRequest" and "Modified"). The Call between the TS and the WAN side is modified at this point.
- g) The TelCP invokes the StartMediaSession() action on the TC2 after the TelCP receives the CallInfo state variable with the Call Status as "Modified". The TC2 starts sending and/or receiving the Media Streams.
- h) The TC2 sends the MediaSessionInfo state variable to the TelCP with the status of the Media Session as "Started".
- i) The TelCP invokes the StartMediaTransfer() action on the TS. The TS starts sending and/or receiving the Media Streams according to the modified Media Capabilities.
- j) The TS sends the CallInfo state variable to the TelCP with the Call Status as "Talking".
- k) If the TelCP wants to disconnect the Call, the TelCP invokes the StopCall() action on the TS.
- l) The TelCP then invokes the StopMediaSession() action on the TC1 and TC2.
- m) The TS sends the CallInfo state variable to the TelCP with the CallStatus as "Disconneted" and the TC1 and TC2 send the MediaSessionInfo state variable to the TelCP with the status of the Media Session as "Stopped".

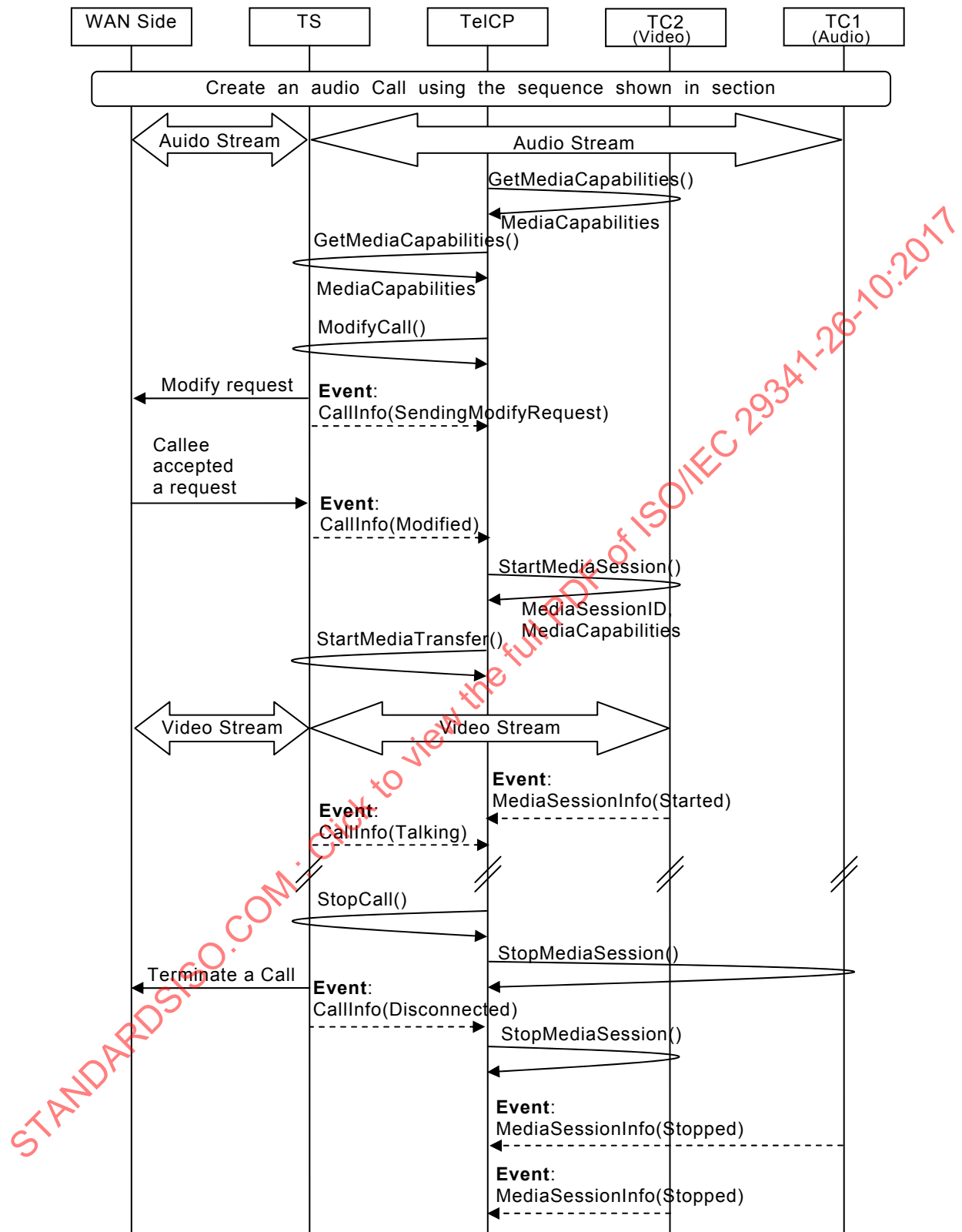


Figure C.3 — Flow basics for modifying an ongoing Call (TC-Based Media Handling)

C.2 TS-Based Media Handling

C.2.1 Create an outgoing Call

The following sequence describes how to create an outgoing Call in the Non-Monopolization Mode using TS-Based Media Handling.

- a) The TelCP determines the Media Capabilities of the TS by invoking the GetMediaCapabilities() action.
- b) The TelCP selects the set of appropriate Media Capabilities for the Call from the Media Capabilities of the TS using the Native Media Capabilities of the TS.
- c) The TelCP invokes the StartCall() action on the TS and the TS initiates the Call on the WAN side.
- d) The TS continuously sends the CallInfo state variable to the TelCP about the state of the Call ("Dialing", "Calling" and "Connected"). The Call between the TS and the WAN side is established at this point.
- e) The TelCP invokes the StartMediaSession() action on the TC after the TelCP receives the CallInfo state variable with the Call Status as "Connected". The TC starts sending and/or receiving the Media Streams.
- f) The TelCP invokes the StartMediaTransfer() action on the TS. The TS starts sending and/or receiving the Media Streams.
- g) The TS sends the CallInfo state variable to the TelCP with the CallStatus as "Talking".
- h) If the TelCP wants to disconnect the Call, the TelCP invokes the StopCall() action on the TS.
- i) The TS sends the CallInfo state variable to the TelCP with the CallStatus as "Disconnected".

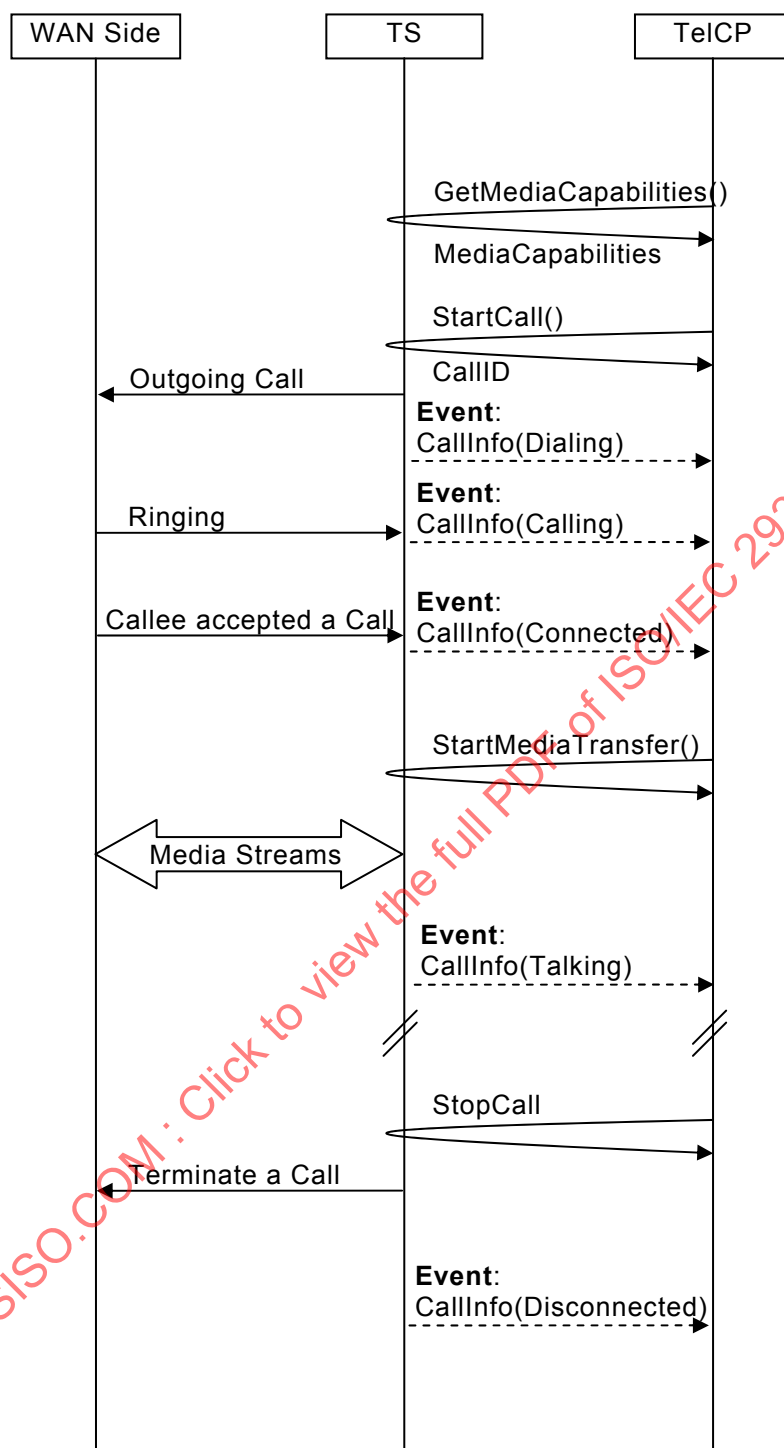


Figure C.4 — Flow basics for creating an outgoing Call (TS-Based Media Handling)

C.2.2 Accept an incoming Call

The following sequence describes how to accept an incoming Call in the Non-Monopolization Mode using TS-Based Media Handling.

- When the TS receives an incoming Call on the WAN side, the TS sends the CallInfo state variable to the TelCP with the Call Status as "Ringing". The CallInfo state variable also includes the Media Capabilities of the Caller.

ISO/IEC 29341-26-10:2017(E)

- b) The TelCP selects the set of appropriate Media Capabilities for the Call from the Native Media Capabilities included in the received CallInfo state variable.
- c) The TelCP invokes the AcceptCall() action on the TS and the TS accepts the Call on the WAN side.
- d) After the Call between the TS and the WAN side is established, the TS sends the CallInfo state variable to the TelCP with the Call Status as "Connected".
- e) The TelCP invokes the StartMediaTransfer() action on the TS. The TS starts sending and/or receiving the Media Streams.
- f) The TS sends the CallInfo state variable to the TelCP with the Call Status as "Talking".
- g) If the TelCP wants to disconnect the Call, the TelCP invokes the StopCall() action on the TS.
- h) The TS sends the CallInfo state variable to the TelCP with the Call Status as "Disconnected".

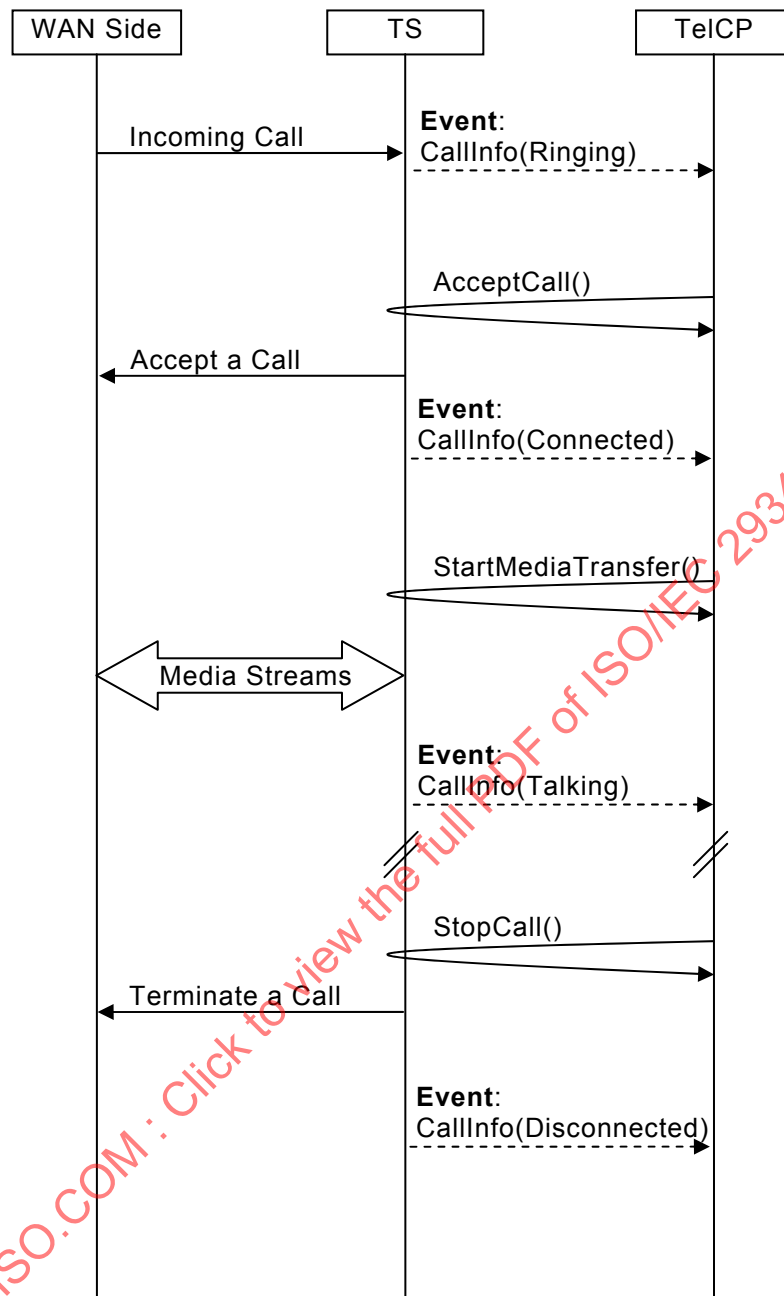


Figure C.5 — Flow basics for accepting an incoming Call (TS-Based Media Handling)

C.2.3 Modify an ongoing Call

The following sequence describes how to modify an ongoing Call from an audio to a video Call in the Non-Monopolization Mode using both TC-Based Media Handling and TS-Based Media Handling. In this sequence, the TelCP uses TS-Based Media Handling for audio capabilities and TC-Based Media Handling for video capabilities. The TC1 supports the video capabilities and is used for TC-Based Media Handling. At first, the TelCP creates an audio Call using TS-Based Media Handling. Then the TelCP modifies the Call from audio to a video Call by adding TC1.

- The TelCP creates an audio Call using TS-Based Media Handling. The sequence is the same as the sequence shown in C.2.1.
- The TelCP determines the Media Capabilities of the TS by invoking the [GetMediaCapabilities\(\)](#) action.

ISO/IEC 29341-26-10:2017(E)

- c) The TelCP determines the Media Capabilities of the TC1 by invoking the GetMediaCapabilities() action.
- d) The TelCP selects the set of appropriate Media Capabilities for the Call to change the Media Capabilities from an audio to a video Call by matching the Media Capabilities of the TS and the TC1.
- e) The TelCP invokes the ModifyCall() action on the TS and the TS modifies the Call on the WAN side.
- f) The TS sends the CallInfo state variable to the TelCP about the state of the Call ("SendingModifyRequest" and "Modified"). The Call between the TS and the WAN side is modified at this point.
- g) The TelCP invokes the StartMediaSession() action on the TC1 after the TelCP receives the CallInfo state variable with the Call Status as "Modified". The TC1 starts sending and/or receiving the Media Streams.
- h) The TC1 sends the MediaSessionInfo state variable to the TelCP with the status of the Media Session as "Started".
- i) The TelCP invokes the StartMediaTransfer() action on the TS. The TS starts sending and/or receiving the Media Streams according to the modified Media Capabilities.
- j) The TS sends the CallInfo state variable to the TelCP with the Call Status as "Talking".
- k) If the TelCP wants to disconnect the Call, the TelCP invokes the StopCall() action on the TS.
- l) The TelCP then invokes the StopMediaSession() action on the TC1.
- m) The TS sends the CallInfo state variable to the TelCP with the CallStatus as "Disconnected" and the TC1 sends the MediaSessionInfo state variable to the TelCP with the status of the Media Session as "Stopped".

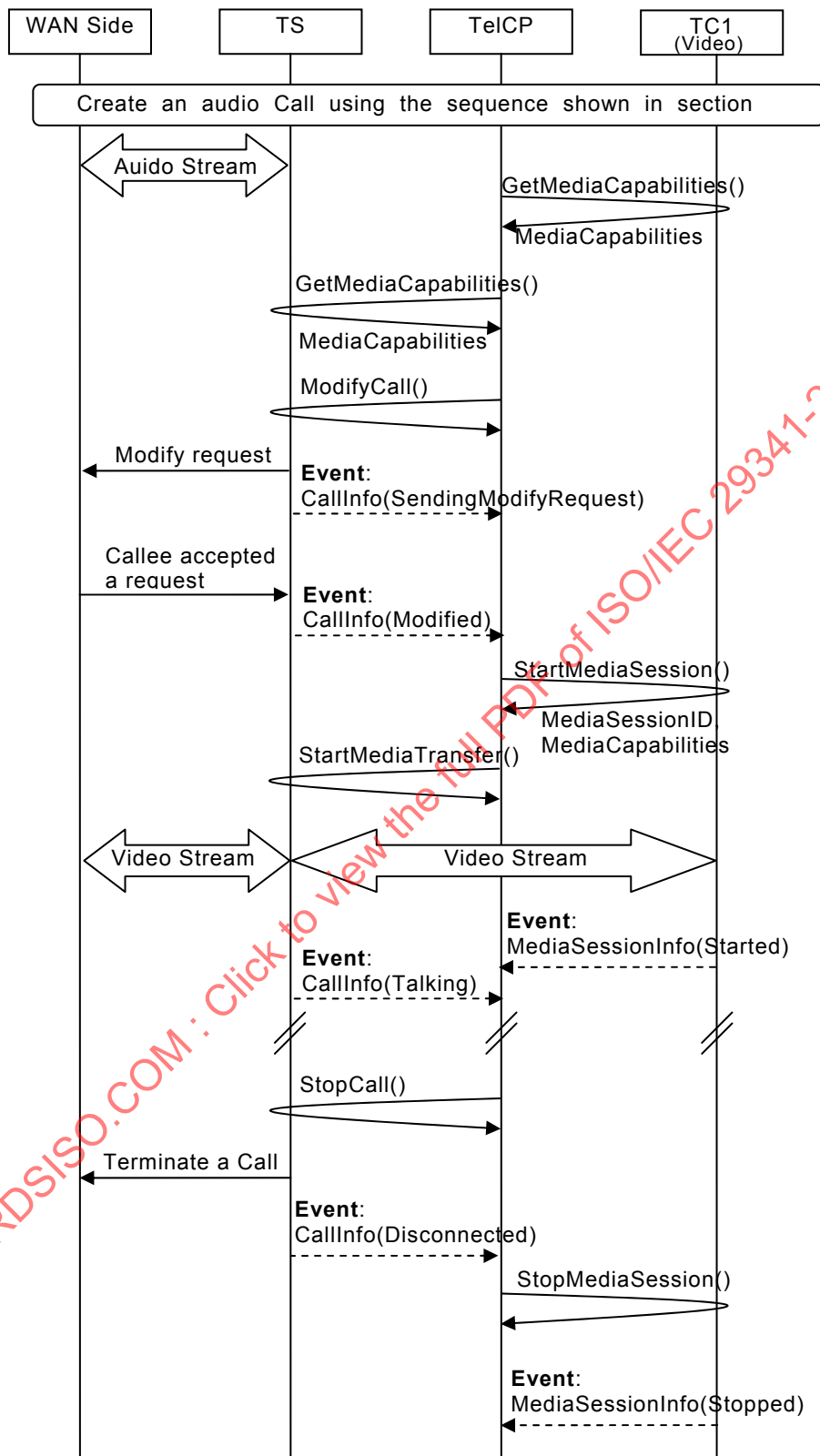


Figure C.6 — Flow basics for modifying an ongoing Call (TS-Based Media Handling)

C.3 Change CallMode

C.3.1 Change CallMode From Non-Monopolization to Monopolization

Figure C.7 shows an example of a sequence to change the CallMode from Non-Monopolization to "PHONE-TeICP". In this sequence, there is an ongoing Non-Monopolization Mode Call with the value of the CallID as "CallID1". The TeICP, whose TeICPName is "TeICP1" and SecretKey is "SecretKey1", changes the CallMode to "PHONE-TeICP" by invoking the ChangeCallMode() action.

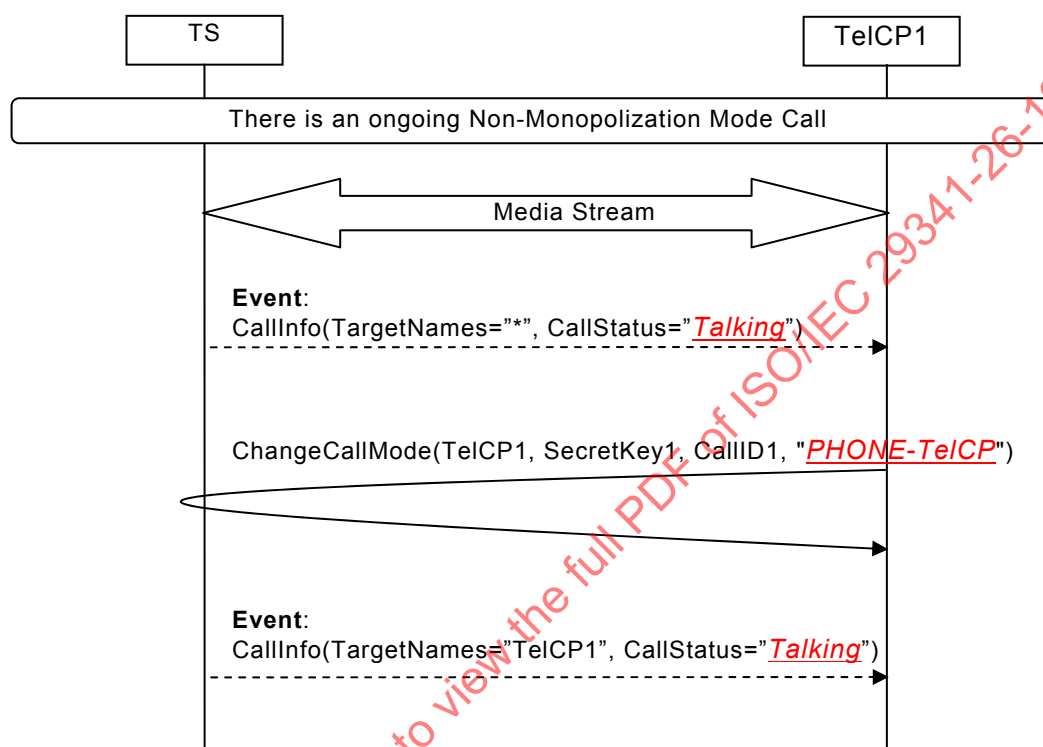


Figure C.7 — Flow basics for changing CallMode from Non-Monopolize to Monopolize

C.3.2 Change CallMode From Monopolization to Non-Monopolization

Figure C.8 shows an example of a sequence to change the CallMode from "PHONE-TeICP" to Non-Monopolization. In this sequence, there is an ongoing Monopolization Mode Call with the value of the CallID as "CallID1". The TeICP, whose TeICPName is "TeICP1" and SecretKey is "SecretKey1", changes the CallMode to Non-Monopolization Mode Call by invoking the ChangeCallMode() action.

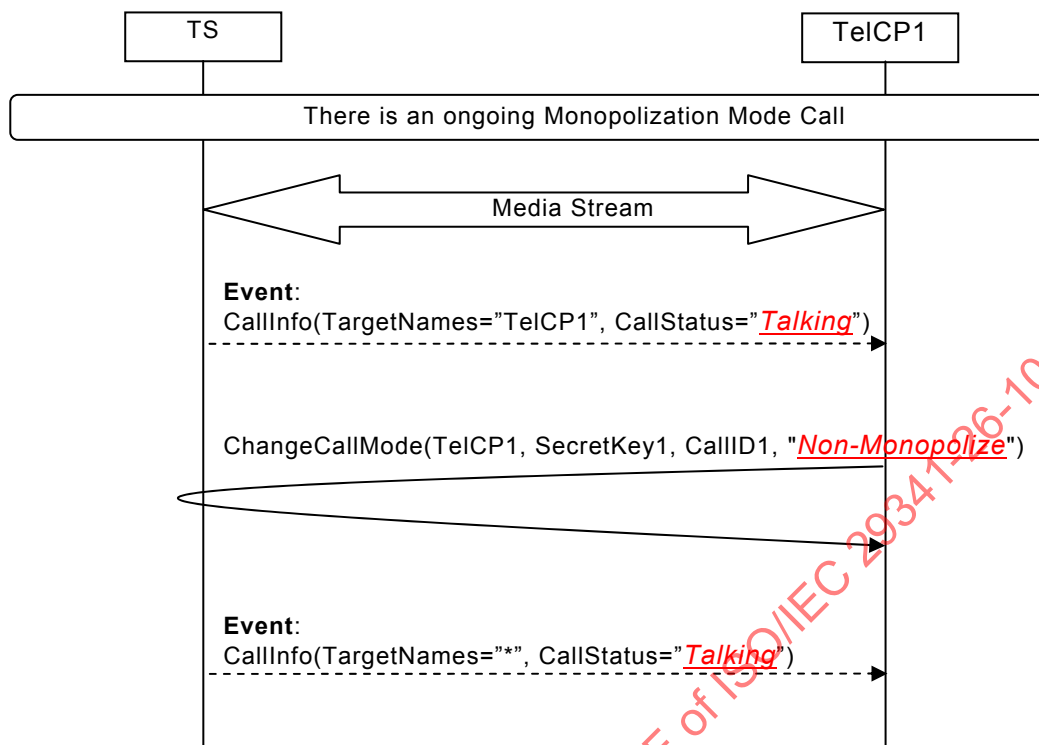


Figure C.8 — Flow basics for changing CallMode from Monopolize to Non-Monopolize

C.4 Early Media

Figure C.9 shows an example sequence for Early Media. The Early Media sequence is different from creating a usual outgoing Call or modifying an existing Call in the following manner:

- When a TelCP receives the CallInfo state variable with the value of the Call Status as "Calling" and the value of the reason attribute as "ReceivingEarlyMediaResponse", the TelCP sets up the Media Session(s) for Early Media between the TS and the TC(s) by invoking the StartMediaSession() action to the TC(s) and the StartMediaTransfer() action to the TS. The Call Status remains as "Calling" and the value of the reason attribute is set to "Early Media Started". The CallInfo state variable is then evented.
- When a TelCP receives the CallInfo state variable with the value of the Call Status as "Connected", the TelCP invokes the ModifyMediaSession() action to the current TC(s) and the StartMediaTransfer() action to the TS. Then the encoding parameters of the Media Streams(s) are modified according to the determined Media Capabilities between the Peers.

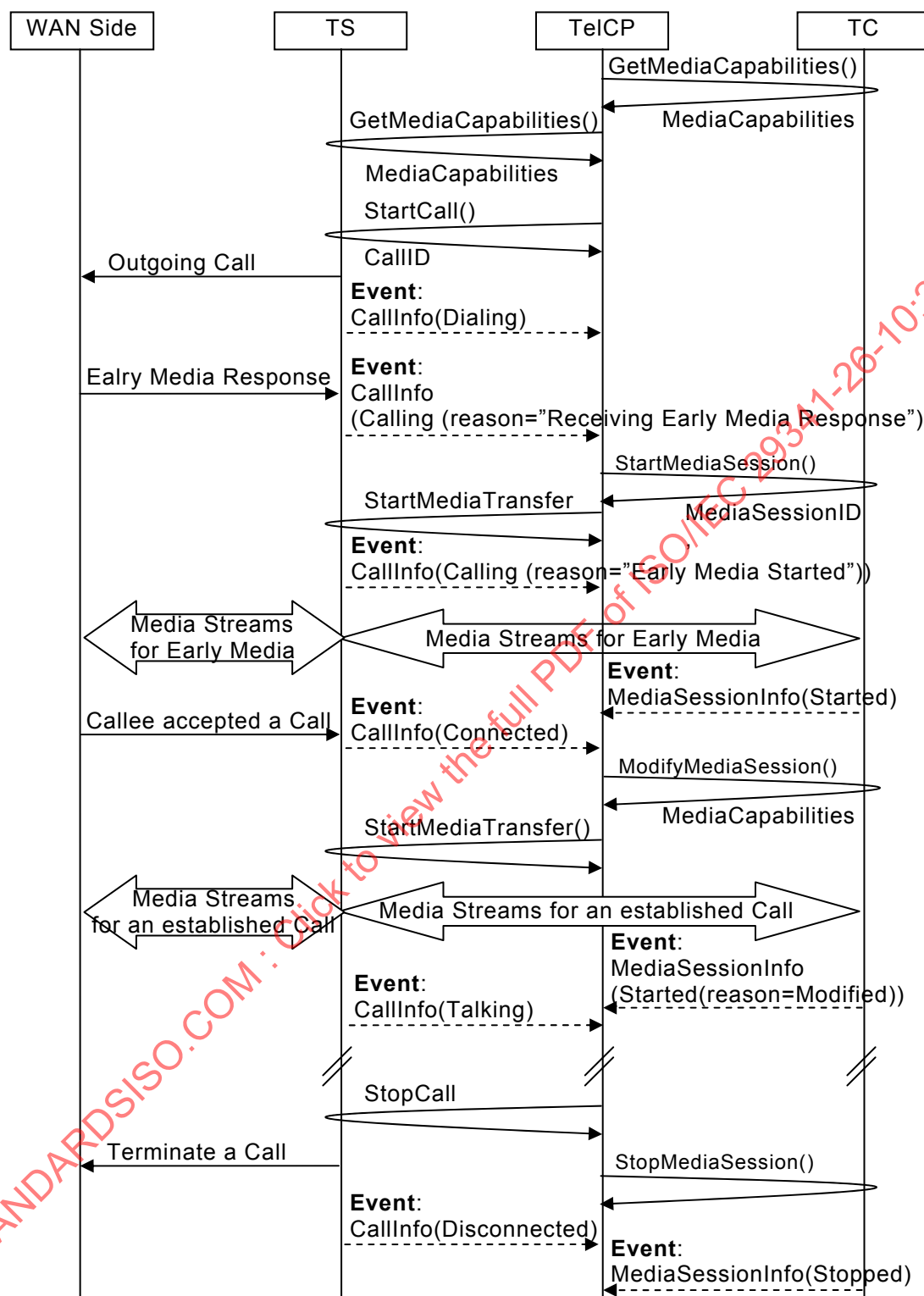


Figure C.9 — Flow basics for Early Media

C.5 Content Sharing during a Call

C.5.1 TS-Based Content Sharing

The Content Sharing feature allows media content to be shared to the Remote Party during a Call. The TS-Based Content Sharing can be initiated after a Call is created using the [ModifyCall\(\)](#) action, or at the same time when initiating a Call using the [StartCall\(\)](#) action. The

TS-Based Content Sharing requires a TS to retrieve the content indicated by a TelCP and send it to the Remote Party as file transfer or streaming means. Figure C.10 and Figure C.11 illustrate how to initiate the TS-Based Content Sharing between the TS and the Remote Party by a TelCP.

Case 1: initiate a Call to share content to the Remote Party. Depending on user operation, a Call could be initiated for an A/V telephony Call as well as Content Sharing, or for sharing the content to the Remote Party only.

- a) The TelCP invokes the GetMediaCapabilities() action on the TS to see if it supports the TS-Based Content Sharing mechanism by checking the presence of the <mediaSharingCapabilities> element of the output argument SupportedMediaCapabilityInfo argument. The <protocolInfo> element indicates the supported Media Capabilities of the TS for Content Sharing.
- b) The TelCP application may also have UPnP AV control point functionalities. The CP invokes the AV Browse() or Search() action of the ContentDirectory service on a MediaServer device in the home network to select content for sharing to the Remote Party. The content metadata is returned by the MS device. The content could also be discovered by other means besides UPnP AV. The content could also be stored locally on the TS device.
- c) When content is selected, the TelCP invokes the StartCall() action to indicate to the TS to initiate a new Call for sharing the content. The metadata information of the content is provided in the <contentSharingCapability> element of the MediaCapabilityInfo input argument of the StartCall() action, which indicates to the TS that the Call would be used to share the content to the Remote Party. The MediaCapabilityInfo argument can also contain parameters for a regular A/V telephony Call.
- d) When receiving the StartCall() action with the <contentSharingCapability> parameters, the TS should negotiate with the Remote Party for a Media Session in the WAN side to transfer the content. The Media Session would contain several Media Streams for the content and the A/V Call (if any) media respectively, or if capable the TS could mix the audio media of the content with the speech media which is out of scope of the UPnP interfaces.
- e) When the WAN side Media Session with the Remote Party is successfully established, the TS events the CallInfo state variable with the "Connected" Call Status to the TelCP.
- f) The TelCP when receiving the event, invokes the StartMediaTransfer() action on the TS to trigger the TS to fetch the content and share to the Remote Party. The MediaCapabilityInfo input argument should contain the same <contentSharingCapability> element as in the StartCall() action.
- g) The TS retrieves the content from the MS device based on the protocol (e.g. HTTP GET) indicated by the <protocolInfo> element in the MediaCapabilityInfo input argument of the StartMediaTransfer() action, and transfers the content to the Remote Party using the established WAN side Media Session.
- h) If the Remote Party device is also a TS and has TelCP and TC in the home network, the content could be rendered on a TC by the operations of a TelCP in the Remote Party network.
- i) The Content Sharing process could be terminated when the content is transferred, or the Call stopped or modified by the TelCP in either side.

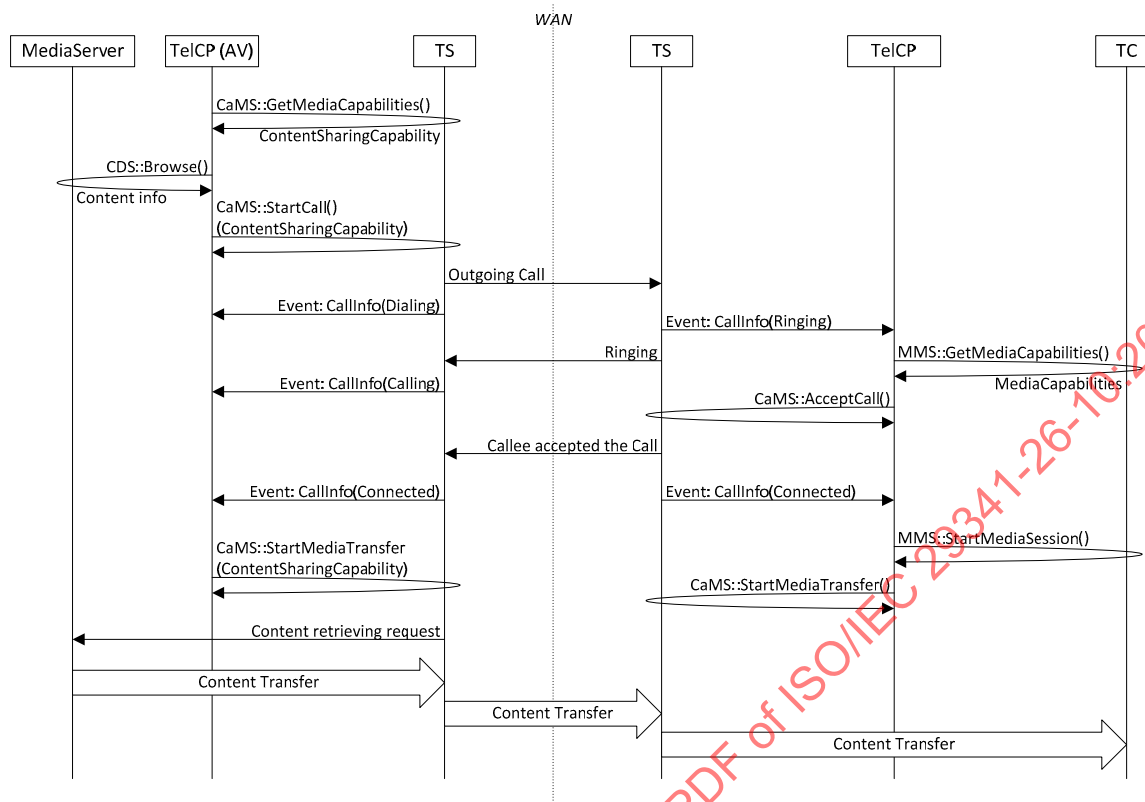


Figure C.10 — Example flow of TS-Based Content Sharing by initiating a Call

Case 2: update an existing Call with the Remote Party to add Content Sharing. The sequence flow as in Figure C.11 is almost the same as starting a new Call in Figure C.10, but to update an existing Call with the Remote Party to add the content streams. The TelCP in this case after selecting the content, invokes the ModifyCall() action on the TS to indicate to the TS that the selected content should be shared to the Remote Party. The TS based on its implementation could renegotiate with the Remote Party to add new Media Streams to the existing Media Session in the WAN side for sharing the content, or if capable mix the content media into the existing Media Streams of the Media Session with the Remote Party.

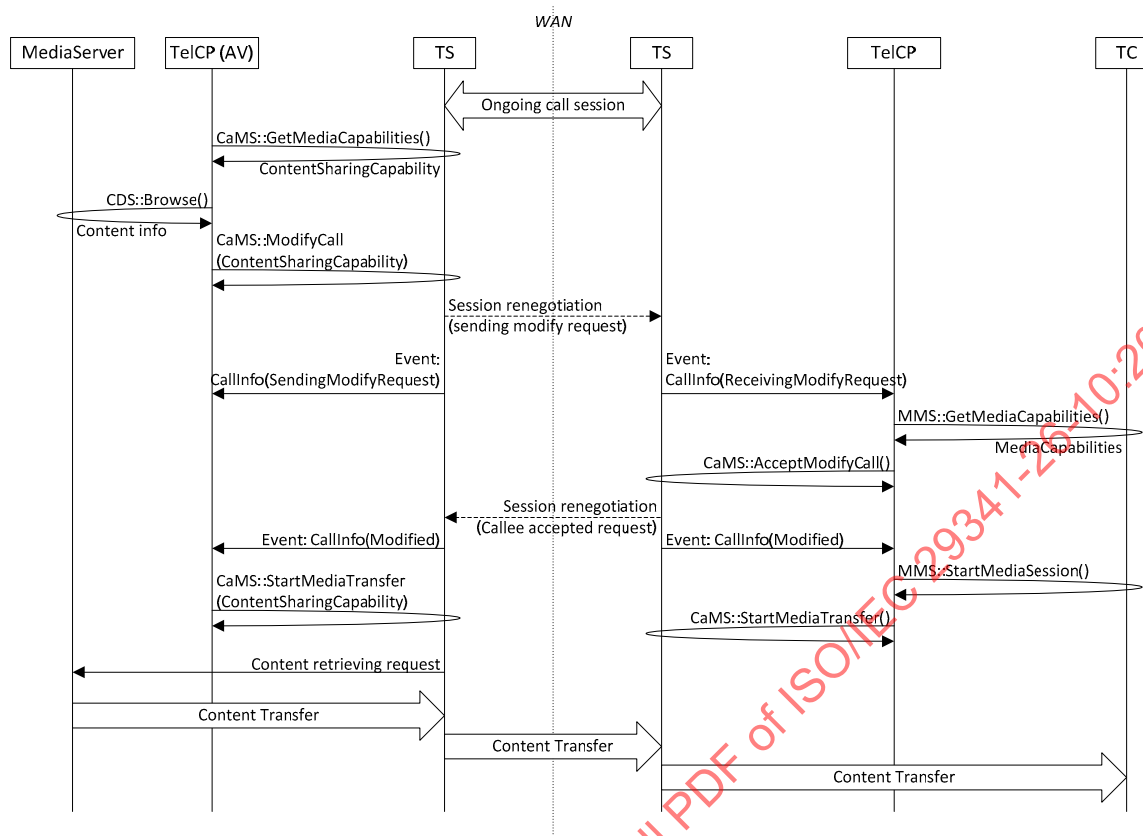


Figure C.11 — Example flow of TS-Based Content Sharing in an existing Call

C.5.2 TC-Based Content Sharing

The TC-Based Content Sharing mechanism requires a TC to participate in a Call to transfer the content received from a content hosting device to the TS using the capabilities supported by both of the TC and the TS, so that the content would be transferred to the Remote Party by the TC via the TS. This mechanism can be initiated after a Call is created or at the same time when initiating a Call. Figure C.12, Figure C.13 and Figure C.14 illustrate how to initiate the TC-Based Content Sharing between the Peers by a TelCP.

Case 1: update an existing Call to share content to the Remote Party. In this case, a TelCP selects the content and provides the content information to the TC, and receives the TC feedback for updating the Media Session with the TS. The TelCP requests the TS to update the WAN side Call to add new Media Streams, and trigger the Content Sharing process when the Call is updated.

- The TelCP invokes the GetMediaCapabilities() action on the TC to see if it supports TC-Based Content Sharing mechanism by checking the presence of the `<mediaSharingCapabilities>` element of the output argument SupportedMediaCapabilityInfo argument, and to know what Media Capabilities are supported by the TC by checking the `<protocolInfo>` element.
- The TelCP application may also have UPnP AV control point functionalities. The CP invokes the AV Browse() or Search() action of the ContentDirectory service on a MediaServer device in the home network to select content for sharing to the Remote Party. The content metadata is returned by the MS device. The content could also be discovered by other means besides UPnP AV. The content could also be stored locally on the TC device.
- When the content is selected, the TelCP again invokes the GetMediaCapabilities() action on the TC. This action will provide the selected content information in the `<contentSharingCapability>` element of the TSMediaCapabilityInfo input argument. The `<mediaCapability>` element should also contain the SDP formatted Media Capabilities of

the ongoing Call so that the TC will not terminate the existing Media Session with the TS for the regular Call.

- d) When receiving this action, the TC based on its and the TS Media Capabilities returns the Media Capabilities of the Media Session that will be used for sharing the content as well as the regular Call. The Media Capabilities should be described using the SDP format in the <mediaCapability> element of the SupportedMediaCapabilityInfo output argument.
- e) The TelCP is responsible for determining whether the Call needs to be modified by comparing the SDP parameters returned by the TC with the one used by the ongoing Call, i.e. to see if new Media Streams are added.
- f) If deciding to update the Call for sharing the content, the TelCP invokes the ModifyCall() action on the TS to pass the new Media Capabilities to the TS. The TS will renegotiate with the Remote Party to add new streams for the sharing the content.
- g) When notified that the Call is successfully modified on the WAN side, the TelCP invokes the ModifyMediaSession() action on the TC to trigger the TC to fetch the content, process (e.g. transcode and packet) and send to the TS. The NewMediaCapabilityInfo input argument should contain the same <contentSharingCapability> elements as in the GetMediaCapabilities() action.
- h) Then the TelCP invokes the StartMediaTransfer() action on the TS to trigger the TS to receive the content Media Stream from the TC and relay to the Remote Party. If the Remote Party device is also a TS and has TelCP and TC in the home network, the content could be rendered on a TC by the operations of a TelCP in the Remote Party network.
- i) The Content Sharing process will be terminated by the TelCP either calling the StopCall() action on the TS or the StopMediaSession() action on the TC.

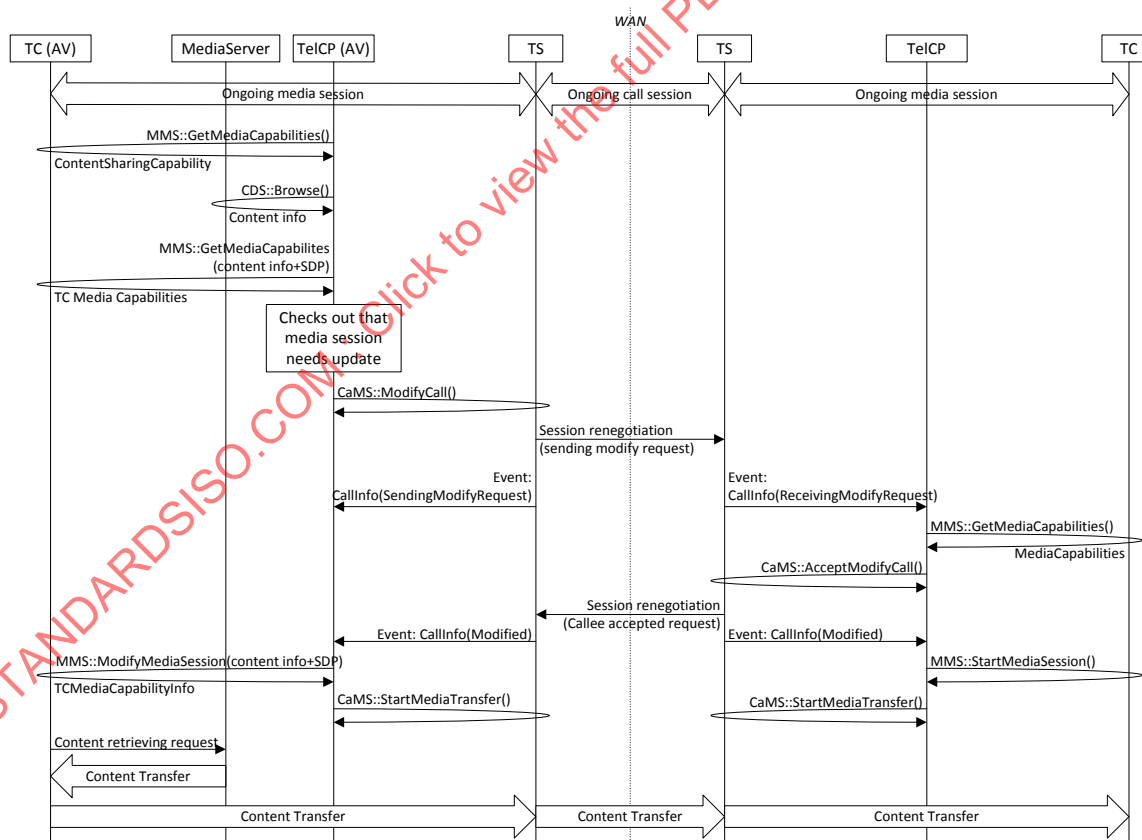


Figure C.12 — Example flow of TC-Based Content Sharing by Updating a Call

Case 2: using an existing Call to share content to the Remote Party without updates, in which case the content media is mixed by the TC into the existing Media Streams of the Media Session with the TS. The operations on the TC are the same as Case 1. The TelCP in this case checks out that the Media Capabilities for the Content Sharing returned by the TC in the

- a) The TelCP based on user operation invokes the GetMediaCapabilities() action on the TC to determine if it supports the TC-Based Content Sharing mechanism. If supports, the TelCP could invoke the UPnP AV actions on a media sever in the home network to select content.
- b) The TelCP invokes the GetMediaCapabilities() action on the TS to know its supported Media Capabilities.
- c) The TelCP then invokes again the GetMediaCapabilities() action on the TC to provide the content info as well as the TS Media Capabilities, in order for the TC to decide on a common Media Capabilities for the Media Session with the TS for Content Sharing as well as a regular Call. The TC returns SDP formatted Media Capabilities description. The TC based on its implementation could decide if to mix the content with the media of the Call, or transfer the content as separate Media Streams, which is out of the scope of the UPnP interfaces.
- d) The TelCP invokes the StartCall() action on the TS to initiate a Call with the Remote Party. The MediaCapabilityInfo input argument should contain the SDP formatted Media Capability information supplied by the TC. Then the TS will initiate a Call to the Remote Party.
- e) When notified that the Call on the WAN side is successfully initiated by the TS, the TelCP invokes the StartMediaSession() action on the TC and the StartMediaTransfer() action on the TS to set up the Media Session for Content Sharing and the regular Call.

© ISO/IEC 2017 – All rights reserved 121

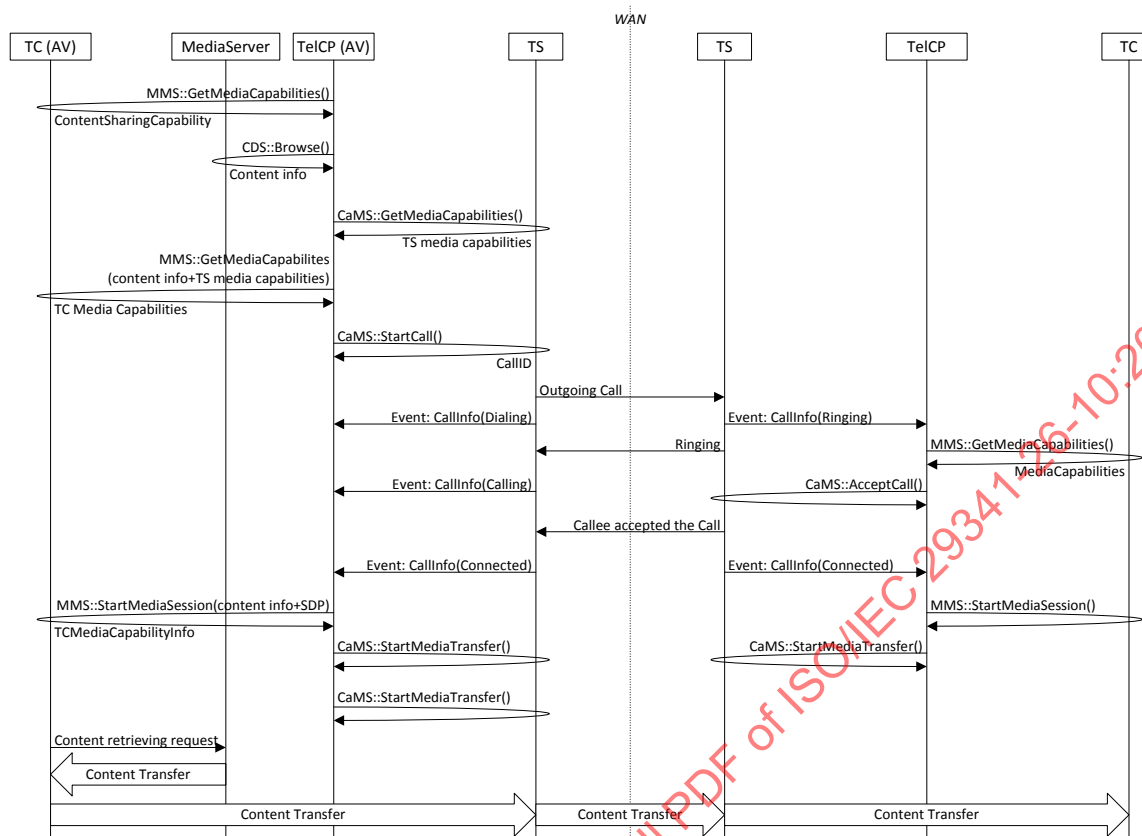


Figure C.14 — Example flow of TC-Based Content Sharing by initiating a Call

C.6 Voice Mail

The sequence of operations for voice mail feature are as follows:

- a) An event for an incoming Call is sent out to all the subscribed TelCP(s).
- b) If no TelCP answers the Call, the TS plays a recorded answering message to the Caller based on user preferences.
 - 1) The user preferences set on the TS may include different answering messages for different Caller. It also can include a default answering message for all the Callers. All these parameters are set using telephony phone data model.
 - 2) An answering message can be located in a UPnP AV *ContentDirectory* Service (CDS) [19] and CDS object identification is used to identify the answering message.
 - 3) The answering message can also be located in any other devices in the home network and URL can be used to identify the message.
- c) If a Call is un-answered, the Caller may decide to leave a voice mail message for the Callee.
- d) If a voice mail is left for the Callee, the TS sends an event to all the subscribed TelCP(s) about an incoming new voice mail. The notification message may include information about the voice mail envelope which include Caller identity, time and duration of the Call, and location of the voice mail etc.

C.7 Parallel Call Initiation

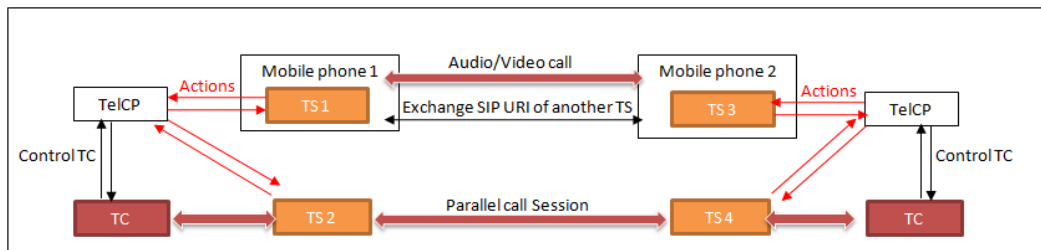


Figure C.15 — Architecture for Parallel Call

Figure C.15 describes a generic model for “Parallel Call initiation”. This model consists of three logical entities such as TS, TC and TelCP. Because these entities are logical unit, the TelCP can reside in any of the physical device. If TS2 and TS4 have the capabilities to render media streams, the TC(s) in this diagram can be omitted.

When mobile phone 1 (MP1) and mobile phone 2 (MP2) are on the Call, the TelCP gets information (e.g., identity) of TS4 through an out of band mechanism. The TelCP then establishes a Call between TS2 and TS4.

C.7.1 Basic Sequence

- a) TS1 and TS3 are on audio Call.
- b) TelCP1 invokes the GetTelephonyIdentity() action on TS2 to get the identity of TS2.
- c) TelCP1 invokes the InitiateParallelCall() to TS1 in the MP1 to get the identity of the TS4. MP1 and MP2 can authenticate each other during this process which is out of the scope of this service.
- d) Using out band mechanism (e.g., SMS etc.), MP1 sends a request to MP2 to get the identity of TS4.
- e) TS3 sends the ParallelCallInfo event with the value of the <informationType> element set as “ParallelCallRequest” event to TelCP2.
- f) TelCP2 invokes the GetTelephonyIdentity() action on TS4.
- g) TelCP2 invokes the WaitingForCall() action on TS4 so that TS4 starts waiting for the incoming Call from TS2.
- h) TelCP2 invokes the AcceptParallelCall() action on TS3 with the identity of TS4.
- i) Using out band mechanism (e.g., SMS etc.), MP2 sends a response to MP1 with the identity of TS4.
- j) TS1 sends the ParallelCallInfo event with the value of the <informationType> element set as “ParallelCallAccepted” with the identity of TS4 to TelCP1.
- k) Using the identity, TelCP1 invokes the EnhancedInitiateCall() action on TS2.
- l) TS2 initiates a Call to TS4.
- m) TS4 automatically accept the Call.
- n) TS2 sends the CallInfo event with the value of the <callStatus> element set as “Connected”.

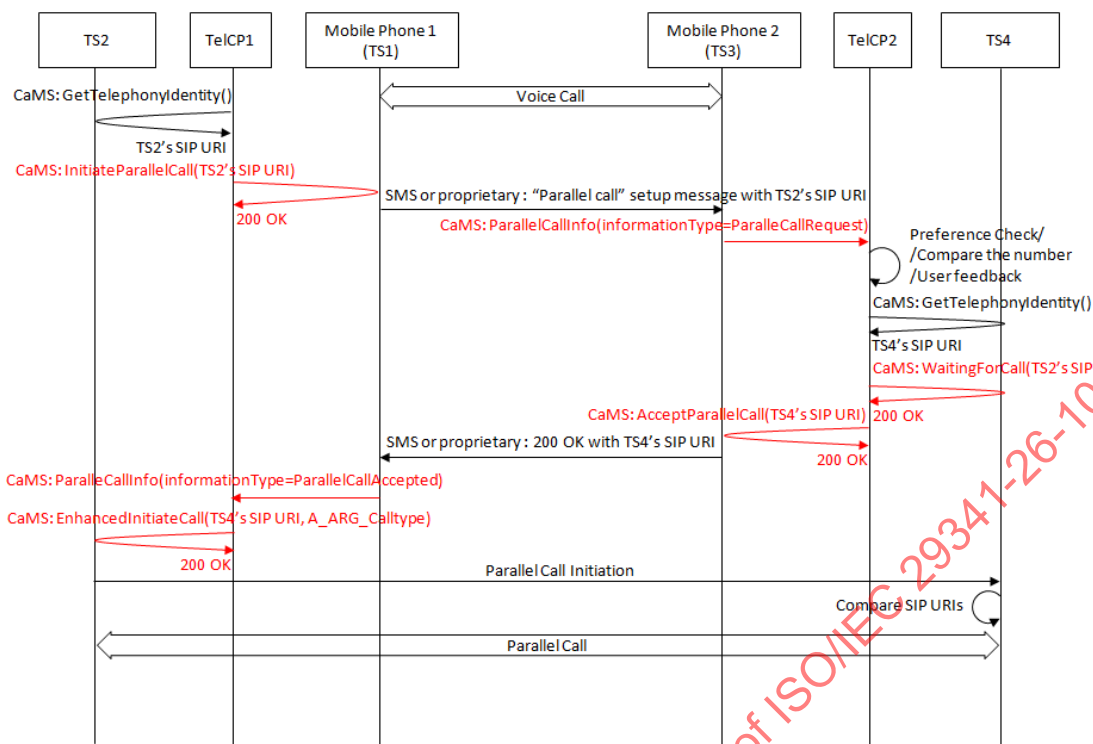


Figure C.16 — Flow basics for Establishing a Parallel Call

Annex D (informative)

Sequence Examples

D.1 Register TelCPName

D.1.1 Figure

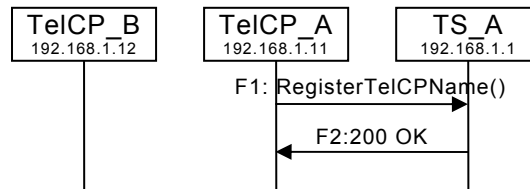


Figure D.1 — Register TelCPName

D.1.2 F1

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#RegisterTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RegisterTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <CurrentSecretKey></CurrentSecretKey>
    </u:RegisterTelCPName>
  </s:Body>
</s:Envelope>
  
```

D.1.3 F2

HTTP/1.1 200 OK
 EXT:
 CONTENT-TYPE: text/xml ; charset="utf-8"
 SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RegisterTelCPNameResponse xmlns:u="urn:schemas-upnp-
      org:service:CallManagement:2">
      <NewSecretKey>secret101</NewSecretKey>
      <Expires>300</Expires>
    </u:RegisterTelCPNameResponse>
  </s:Body>
</s:Envelope>
  
```

D.2 Register TelCPName (But the specified TelCPName is already in use.)

D.2.1 Figure

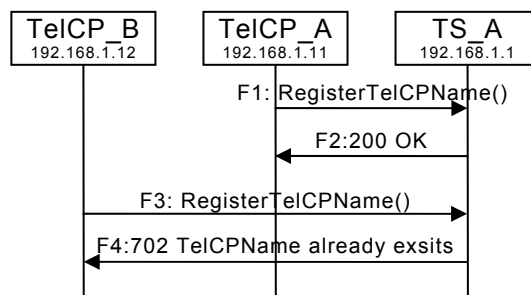


Figure D.2 — Register TelCPName (But the specified TelCPName is already in use.)

D.2.2 F1 and F2 are the same as “D.1 Register TelCPName”

D.2.3 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#RegisterTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RegisterTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <CurrentSecretKey></CurrentSecretKey>
    </u:RegisterTelCPName>
  </s:Body>
</s:Envelope>
    
```

D.2.4 F4

HTTP/1.1 500 Internal Server Error
 Content-Length: [bytes in body]

```

<s:Envelope
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <s:Fault>
      <faultcode>s:Client</faultcode>
      <faultstring>UPnPError</faultstring>
      <detail>
        <UPnPError xmlns="urn:schemas-upnp-org:control-1-0">
          <errorCode>702</errorCode>
          <errorDescription>TelCPName already exists</errorDescription>
        </UPnPError>
      </detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
    
```

D.3 Keep using the same TelCPName

D.3.1 Figure

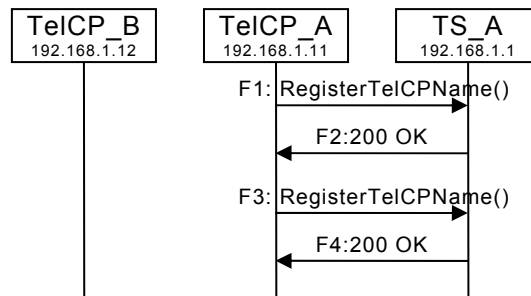


Figure D.3 — Keep using the same TelCPName

D.3.2 F1 and F2 are the same as “D.1 Register TelCPName”

D.3.3 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#RegisterTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RegisterTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <CurrentSecretKey>secret101</CurrentSecretKey>
    </u:RegisterTelCPName>
  </s:Body>
</s:Envelope>
  
```

D.3.4 F4

HTTP/1.1 200 OK
 EXT:
 CONTENT-TYPE: text/xml ; charset="utf-8"
 SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RegisterTelCPNameResponse xmlns:u="urn:schemas-upnp-
org:service:CallManagement:2">
      <NewSecretKey>secret111</NewSecretKey>
      <Expires>300</Expires>
    </u:RegisterTelCPNameResponse>
  </s:Body>
</s:Envelope>
  
```

D.4 Keep using the same TelCPName (But the specified Secret Key is invalid.)

D.4.1 Figure

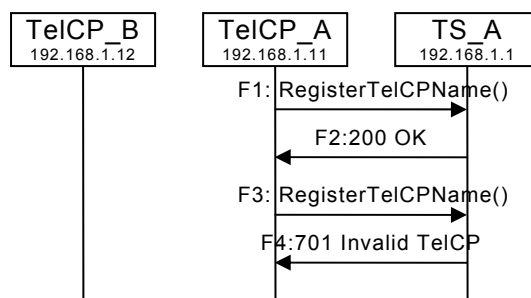


Figure D.4 — Keep using the same TelCPName (But the specified Secret Key is invalid.)

D.4.2 F1 and F2 are the same as “D.1 Register TelCPName”

D.4.3 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#RegisterTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RegisterTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <CurrentSecretKey>wrongkey</CurrentSecretKey>
    </u:RegisterTelCPName>
  </s:Body>
</s:Envelope>
    
```

D.4.4 F4

HTTP/1.1 500 Internal Server Error
 Content-Length: [bytes in body]

```

<s:Envelope
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <s:Fault>
      <faultcode>s:Client</faultcode>
      <faultstring>UPnPError</faultstring>
      <detail>
        <UPnPError xmlns="urn:schemas-upnp-org:control-1-0">
          <errorCode>701</errorCode>
          <errorDescription>Invalid TelCP</errorDescription>
        </UPnPError>
      </detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
    
```

D.5 Change TelCPName

D.5.1 Figure

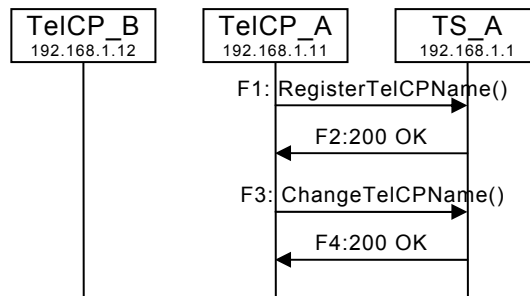


Figure D.5 — Change TelCPName

D.5.2 F1 and F2 are the same as “D.1 Register TelCPName”

D.5.3 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#ChangeTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:ChangeTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <CurrentTelCPName>TV@Living</CurrentTelCPName>
      <CurrentSecretKey>secret101</CurrentSecretKey>
      <NewTelCPName>HD-TV@Living</NewTelCPName>
    </u:ChangeTelCPName>
  </s:Body>
</s:Envelope>
  
```

D.5.4 F4

HTTP/1.1 200 OK
 EXT:
 CONTENT-TYPE: text/xml ; charset="utf-8"
 SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:ChangeTelCPNameResponse xmlns:u="urn:schemas-upnp-
org:service:CallManagement:2">
      <NewSecretKey>secret111</NewSecretKey>
      <Expires>300</Expires>
    </u:ChangeTelCPNameResponse>
  </s:Body>
</s:Envelope>
  
```

D.6 Change TelCPName (But the specified Secret Key is invalid.)

D.6.1 Figure

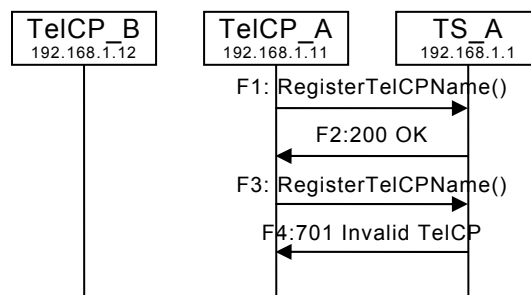


Figure D.6 — Change TelCPName (But the specified Secret Key is invalid.)

D.6.2 F1 and F2 are the same as “D.1 Register TelCPName”

D.6.3 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#ChangeTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

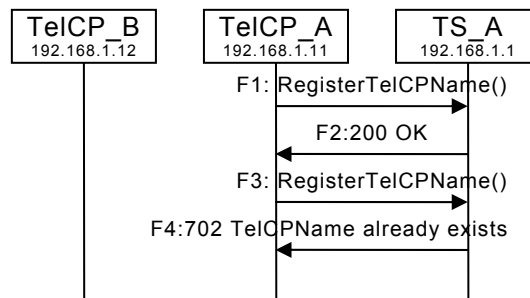
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:ChangeTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <CurrentTelCPName>TV@Living</CurrentTelCPName>
      <CurrentSecretKey>wrongcode</CurrentSecretKey>
      <NewTelCPName>HD-TV@Living</NewTelCPName>
    </u:ChangeTelCPName>
  </s:Body>
</s:Envelope>
    
```

D.6.4 F4

HTTP/1.1 500 Internal Server Error
 Content-Length: [bytes in body]

```

<s:Envelope
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <s:Fault>
      <faultcode>s:Client</faultcode>
      <faultstring>UPnPError</faultstring>
      <detail>
        <UPnPError xmlns="urn:schemas-upnp-org:control-1-0">
          <errorCode>701</errorCode>
          <errorDescription>Invalid TelCP</errorDescription>
        </UPnPError>
      </detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
    
```

D.7 Change TelCPName (But the specified TelCPName is already in use.)**D.7.1 Figure****Figure D.7 — Change TelCPName (But the specified TelCPName is already in use.)****D.7.2 F1 and F2 are the same as “D.1 Register TelCPName”****D.7.3 F3**

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#ChangeTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:ChangeTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <CurrentTelCPName>TV@Living</CurrentTelCPName>
      <CurrentSecretKey>secret101</CurrentSecretKey>
      <NewTelCPName>HD-TV@Living</NewTelCPName>
    </u:ChangeTelCPName>
  </s:Body>
</s:Envelope>
  
```

D.7.4 F4

HTTP/1.1 500 Internal Server Error
 Content-Length: [bytes in body]

```

<s:Envelope
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <s:Fault>
      <faultcode>s:Client</faultcode>
      <faultstring>UPnPError</faultstring>
      <detail>
        <UPnPError xmlns="urn:schemas-upnp-org:control-1-0">
          <errorCode>702</errorCode>
          <errorDescription>TelCPName already exists</errorDescription>
        </UPnPError>
      </detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
  
```

D.8 Unregister TelCPName

D.8.1 Figure

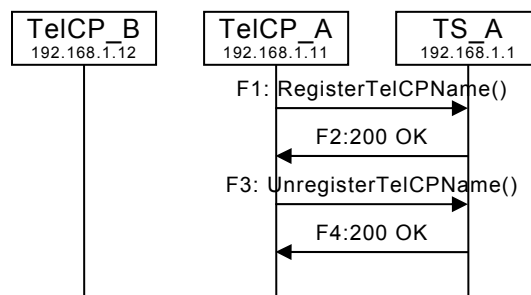


Figure D.8 — Unregister TelCPName

D.8.2 F1 and F2 are the same as “D.1 Register TelCPName”

D.8.3 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#UnregisterTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

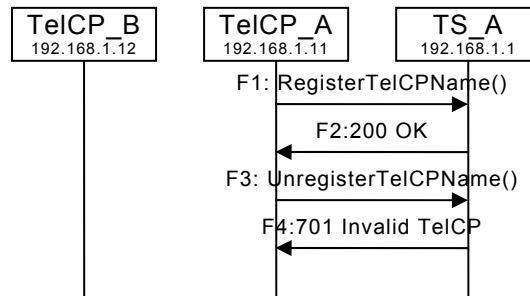
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:UnregisterTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <SecretKey>secret101</SecretKey>
    </u:UnregisterTelCPName>
  </s:Body>
</s:Envelope>
    
```

D.8.4 F4

HTTP/1.1 200 OK
 EXT:
 CONTENT-TYPE: text/xml ; charset="utf-8"
 SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:UnregisterTelCPNameResponse xmlns:u="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
    
```

D.9 Unregister TelCPName (But the specified TelCPName does not exist.)**D.9.1 Figure****Figure D.9 — Unregister TelCPName (But the specified TelCPName does not exist.)****D.9.2 F1 and F2 are the same as “D.1 Register TelCPName”****D.9.3 F3**

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#UnregisterTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:UnregisterTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>Alice's TV</TelCPName>
      <SecretKey>secret101</SecretKey>
    </u:UnregisterTelCPName>
  </s:Body>
</s:Envelope>
  
```

D.9.4 F4

HTTP/1.1 500 Internal Server Error
 Content-Length: [bytes in body]

```

<s:Envelope
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <s:Fault>
      <faultcode>s:Client</faultcode>
      <faultstring>UPnPError</faultstring>
      <detail>
        <UPnPError xmlns="urn:schemas-upnp-org:control-1-0">
          <errorCode>701</errorCode>
          <errorDescription>Invalid TelCP</errorDescription>
        </UPnPError>
      </detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
  
```

D.10 Get existing TelCPNames

D.10.1 Figure

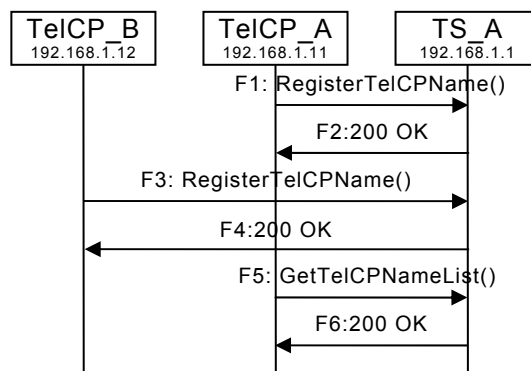


Figure D.10 — Get existing TelCPNames

D.10.2 F1 and F2 are the same as “D.1 Register TelCPName”

D.10.3 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#RegisterTelCPName"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RegisterTelCPName xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>Alice's TV</TelCPName>
      <CurrentSecretKey></CurrentSecretKey>
    </u:RegisterTelCPName>
  </s:Body>
</s:Envelope>
    
```

D.10.4 F4

HTTP/1.1 200 OK
 EXT:
 CONTENT-TYPE: text/xml ; charset="utf-8"
 SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RegisterTelCPNameResponse xmlns:u="urn:schemas-upnp-
      org:service:CallManagement:2">
      <NewSecretKey>secret111</NewSecretKey>
      <Expires>300</Expires>
    </u:RegisterTelCPNameResponse>
  </s:Body>
</s:Envelope>
    
```

D.10.5 F5

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#GetTelCPNameList"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetTelCPNameList xmlns:u="urn:schemas-upnp-org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.10.6 F6

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetTelCPNameListResponse xmlns:u="urn:schemas-upnp-
org:service:CallManagement:2">
      <TelCPNameList>TV@Living, Alice's TV</TelCPNameList>
    </u:GetTelCPNameListResponse>
  </s:Body>
</s:Envelope>
```

D.11 Get existing TelCPNames (But no TelCP exists)

D.11.1 Figure

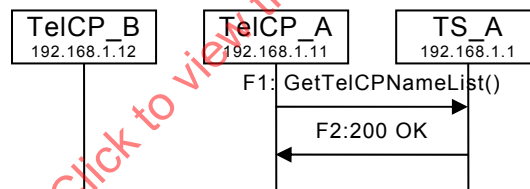


Figure D.11 — Get existing TelCPNames (But no TelCP exists)

D.11.2 F1

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#GetTelCPNameList"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetTelCPNameList xmlns:u="urn:schemas-upnp-org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.11.3 F2

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
```

ISO/IEC 29341-26-10:2017(E)

```
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetTelCPNameListResponse xmlns:u="urn:schemas-upnp-
org:service:CallManagement:2">
      <TelCPNameList></TelCPNameList>
    </u:GetTelCPNameListResponse>
  </s:Body>
</s:Envelope>
```

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 29341-26-10:2017

D.12 Create a Call

D.12.1 Figure

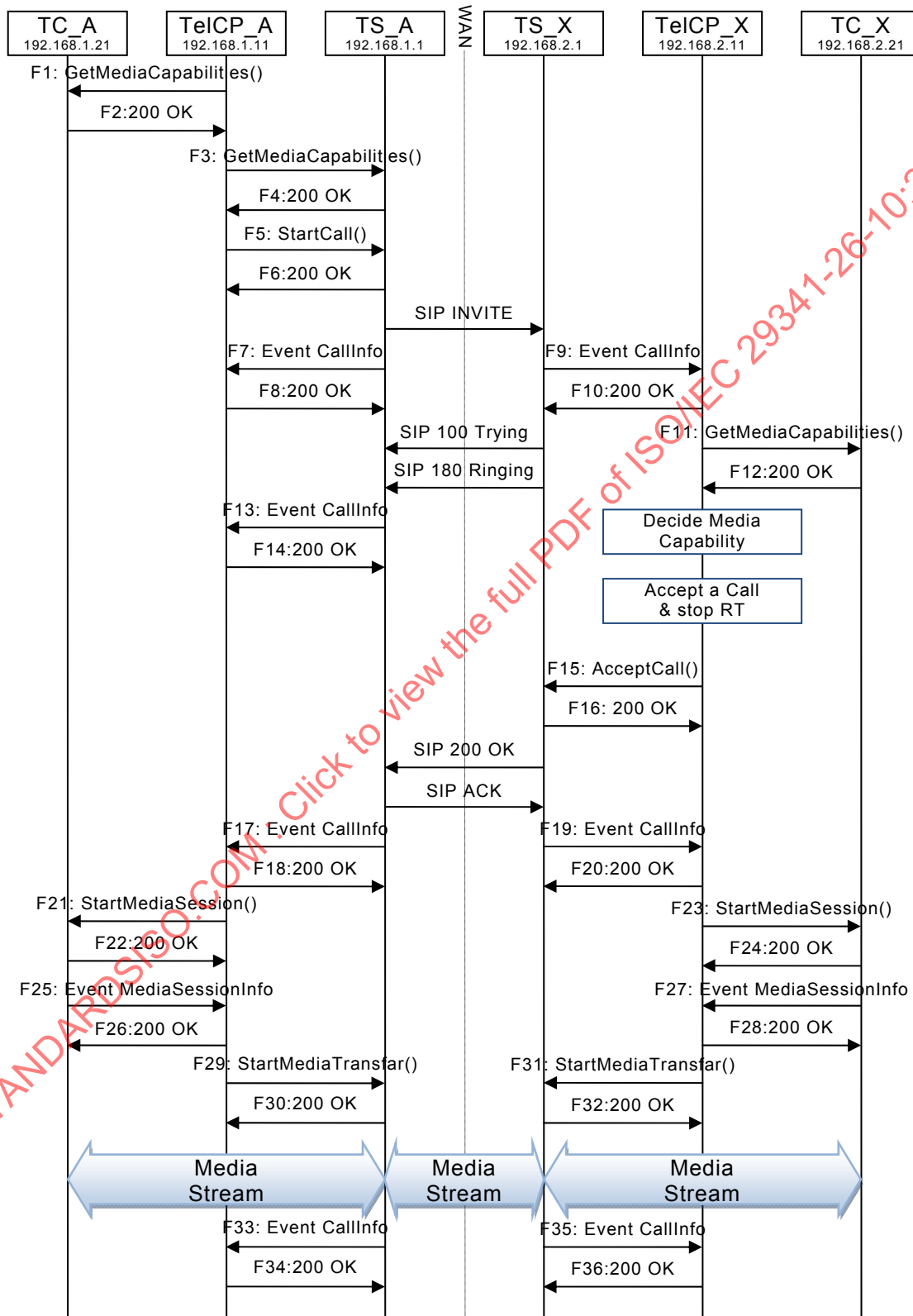


Figure D.12 — Create a Call

ISO/IEC 29341-26-10:2017(E)

D.12.2 F1

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaCapabilities"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.21:10021
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2" >
      <TSMediaCapabilityInfo></TSMediaCapabilityInfo>
    </u:GetMediaCapabilities>
  </s:Body>
</s:Envelope>
```

D.12.3 F2

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <SupportedMediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.21
s=-
c=IN IP4 192.168.1.21
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12021 RTP/AVP 96
c=IN IP4 192.168.1.21
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/mms:mediaCapabilityInfo&gt;
&lt;/SupportedMediaCapabilityInfo>
    </u:GetMediaCapabilitiesResponse>
  </s:Body>
</s:Envelope>
```

D.12.4 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#GetMediaCapabilities"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service: CallManagement:2">
  <TCMediaCapabilityInfo></TCMediaCapabilityInfo>
</u:GetMediaCapabilities>
</s:Body>
</s:Envelope>
```

D.12.5 F4

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:CallManagement:2">
  <SupportedMediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/cams:mediaCapabilityInfo&gt;</SupportedMediaCapabilityInfo>
</u:GetMediaCapabilitiesResponse>
</s:Body>
</s:Envelope>
```

D.12.6 F5

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartCall"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
  <TelCPName></TelCPName>
```

ISO/IEC 29341-26-10:2017(E)

```
<SecretKey></SecretKey>
<CalleeID>tel:0774940201</CalleeID>
<CallPriority>Normal</CallPriority>
<MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
<lt;cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
    <lt;mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
    <lt;/mediaCapability>
    <lt;/cams:mediaCapabilityInfo>
</MediaCapabilityInfo>
    <CallMode>Non-Monopolize</CallMode>
</u:StartCall>
</s:Body>
</s:Envelope>
```

D.12.7 F6

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartCallResponse xmlns:m="urn:schemas-upnp-org:service:CallManagement:2">
      <CallID>call1101</CallID>
    </m:StartCallResponse>
  </s:Body>
</s:Envelope>
```

D.12.8 F7

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10001

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo><?xml version="1.0" encoding="utf-8"?>
    <lt;cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
      http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer">
        <lt;callID>call1101</callID>
        <lt;targetNames type="TelCPName">TV@Living</targetNames>
```

```

    <callStatus><Dialing></callStatus>;
    <priority><Normal></priority>;
    <remoteParty>;
    <peer:id><0774940201></peer:id>;
    </remoteParty>;
    <mediaCapability format="SDP"><v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
    </mediaCapability>;
    </cams:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.12.9 F8

HTTP/1.1 200 OK
Content-Length: 0

D.12.10 F9

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.2.11:20011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-20001
SEQ: [sequence number]
Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
    <cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer">
      <callID><call201></callID>;
      <targetNames type="TelCPName"><*></targetNames>;
      <callStatus><Ringing></callStatus>;
      <priority><Normal></priority>;
      <remoteParty>;
      <peer:id><035550101></peer:id>;
      </remoteParty>;
      <mediaCapability format="SDP"><v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
      </mediaCapability>;
      </cams:callInfo></CallInfo>
    </e:property>
  </e:propertyset>

```

ISO/IEC 29341-26-10:2017(E)

D.12.11 F10

HTTP/1.1 200 OK
Content-Length: 0

D.12.12 F11

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaCapabilities"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.21:20021
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2" >
      <TSMediaCapabilityInfo></TSMediaCapabilityInfo>
    </u:GetMediaCapabilities>
  </s:Body>
</s:Envelope>
```

D.12.13 F12

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <SupportedMediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.21
s=-
c=IN IP4 192.168.2.21
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCM/8000
a=ptime:20
m=video 22021 RTP/AVP 96
c=IN IP4 192.168.2.21
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/mms:mediaCapabilityInfo&gt;</SupportedMediaCapabilityInfo>
    </u:GetMediaCapabilitiesResponse>
  </s:Body>
</s:Envelope>
```

D.12.14 F13

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.1.11:10011
 CONTENT-TYPE: text/xml
 Content-Length: [bytes in body]
 NT: upnp:event
 NTS: upnp:propchange
 SID: uuid:c3580410-1dd1-11b2-8000-10001
 SEQ: [sequence number]
 Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
    &lt;cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
        http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
        &lt;callID&gt;call1101&lt;/callID&gt;
        &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
        &lt;callStatus&gt;Calling&lt;/callStatus&gt;
        &lt;priority&gt;Normal&lt;/priority&gt;
        &lt;remoteParty&gt;
          &lt;peer:id&gt;0774940201&lt;/peer:id&gt;
        &lt;/remoteParty&gt;
        &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
  </e:property>
</e:propertyset>
```

D.12.15 F14

HTTP/1.1 200 OK
 Content-Length: 0

D.12.16 F15

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#AcceptCall"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.2.1:20001
 Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:AcceptCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName></TelCPName>
      <SecretKey></SecretKey>
      <TargetCallID>call1201</TargetCallID>
      <MediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
    &lt;mediaCapability format="SDP"&gt;v=0
```

ISO/IEC 29341-26-10:2017(E)

```
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
<!-- /mediaCapability -->
<!-- /cams:mediaCapabilityInfo -->
  <CallMode>"Non-Monopolize"</CallMode>
</u:AcceptCall>
</s:Body>
</s:Envelope>
```

D.12.17 F16

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<s:Body>
<m:AcceptCallResponse xmlns:m="urn:schemas-upnp-org:service:CallManagement:2" />
</s:Body>
</s:Envelope>
```

D.12.18 F17

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10001

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<!-- /callInfo -->
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
    <!-- callID --> call101</callID>
    <!-- targetNames type="TelCPName" --> *</targetNames>
    <!-- callStatus --> Connected</callStatus>
    <!-- priority --> Normal</priority>
    <!-- remoteParty -->
      <!-- peer:id --> 0774940201</peer:id>
    </remoteParty>
    <!-- mediaCapability format="SDP" --> v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
```

```

a=ptime:20
<lt;/mediaCapability>
<lt;/cams:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.12.19 F18

HTTP/1.1 200 OK
Content-Length: 0

D.12.20 F19

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.2.11:20011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-20001
SEQ: [sequence number]
Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><lt;?xml version="1.0" encoding="utf-8"?>
<lt;cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <lt;callID><lt;callID>
  <lt;targetNames type="TelCPName">*<lt;/targetNames>
  <lt;callStatus><lt;Connected><lt;/callStatus>
  <lt;priority><lt;Normal><lt;/priority>
  <lt;remoteParty>
    <lt;peer:id><lt;0355550101><lt;/peer:id>
  <lt;/remoteParty>
  <lt;mediaCapability format="SDP"><lt;v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
<lt;/mediaCapability>
<lt;/cams:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.12.21 F20

HTTP/1.1 200 OK
Content-Length: 0

D.12.22 F21

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.21:10021
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>

```

ISO/IEC 29341-26-10:2017(E)

```
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <lt;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <lt;mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
          </mediaCapability>
          </mms:mediaCapabilityInfo>
        </TSMediaCapabilityInfo>
      </u:StartMediaSession>
    </s:Body>
  </s:Envelope>
```

D.12.23 F22

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <MediaSessionID>media101</MediaSessionID>
      <TCMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <lt;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <lt;mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.21
s=-
c=IN IP4 192.168.1.21
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
          </mediaCapability>
          </mms:mediaCapabilityInfo>
        </TCMediaCapabilityInfo>
      </u:StartMediaSessionResponse>
    </s:Body>
  </s:Envelope>
```

D.12.24 F23

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.2.21:20021

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <t:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
          </t:mediaCapability>
        </t:mms:mediaCapabilityInfo>
      </TSMediaCapabilityInfo>
    </u:StartMediaSession>
  </s:Body>
</s:Envelope>
```

D.12.25 F24

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <MediaSessionID>media201</MediaSessionID>
      <TCMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <t:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.21
s=-
c=IN IP4 192.168.2.21
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
          </t:mediaCapability>
        </t:mms:mediaCapabilityInfo>
      </TCMediaCapabilityInfo>
    </u:StartMediaSessionResponse>
  </s:Body>
</s:Envelope>
```

ISO/IEC 29341-26-10:2017(E)

D.12.26 F25

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10021

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <MediaSessionInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
      &lt;mms:mediaSessionInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
          http://www.upnp.org/schemas/phone/mms-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
          &lt;mediaSessionID&gt;media101&lt;/mediaSessionID&gt;
          &lt;mediaSessionStatus&gt;Started&lt;/mediaSessionStatus&gt;
          &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.21
s=-
c=IN IP4 192.168.1.21
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/mms:mediaSessionInfo&gt;</MediaSessionInfo>
  </e:property>
</e:propertyset>
```

D.12.27 F26

HTTP/1.1 200 OK

Content-Length: 0

D.12.28 F27

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20021

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <MediaSessionInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
      &lt;mms:mediaSessionInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
          http://www.upnp.org/schemas/phone/mms-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
          &lt;mediaSessionID&gt;media201&lt;/mediaSessionID&gt;
          &lt;mediaSessionStatus&gt;Started&lt;/mediaSessionStatus&gt;
          &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.21
s=-
```

```

c=IN IP4 192.168.2.21
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
</mediaCapability>
</mms:mediaSessionInfo></MediaSessionInfo>
</e:property>
</e:propertyset>

```

D.12.29 F28

HTTP/1.1 200 OK
Content-Length: 0

D.12.30 F29

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.1:10001
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
<TelCPName></TelCPName>
<SecretKey></SecretKey>
<TargetCallID>call101</TargetCallID>
<TCList><?xml version="1.0" encoding="utf-8"?>
<tc:cams:TCList
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
<tc:TC>
<tc:UDN>97ef6efa-ac89-4ea2-0001-udn101</tc:UDN>
<tc:mediaSessionID>media101</tc:mediaSessionID>
</tc:TC>
</tc:cams:TCList></TCList>
<MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
<tc:cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
<tc:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
</mediaCapability>
</tc:cams:mediaCapabilityInfo></MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
</s:Envelope>

```

D.12.31 F30

HTTP/1.1 200 OK
Content-Length: [bytes in body]
CONTENT-TYPE: text/xml; charset="utf-8"

ISO/IEC 29341-26-10:2017(E)

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0
EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.12.32 F31

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.1:20001
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName></TelCPName>
      <SecretKey></SecretKey>
      <TargetCallID>call201</TargetCallID>
      <TCList><?xml version="1.0" encoding="utf-8"?>
        <t:cams:TCList
          xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:cams">
            <t:TC>
              <t:UDN>uid:97ef6efa-ac89-4ea2-0001-udn201</t:UDN>
              <t:mediaSessionID>media201</t:mediaSessionID>
            </t:TC>
          </t:cams:TCList></TCList>
      <MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:cams:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:cams">
          <t:mediaCapability format="SDP">
            o=- 0 0 IN IP4 192.168.2.11
            s=-
            c=IN IP4 192.168.2.11
            t=0 0
            m=audio 21021 RTP/AVP 0
            c=IN IP4 192.168.2.21
            a=rtpmap:0 PCMU/8000
            aptime:20
          </t:mediaCapability>
        </t:cams:mediaCapabilityInfo></MediaCapabilityInfo>
    </u:StartMediaTransfer>
  </s:Body>
</s:Envelope>
```

D.12.33 F32

HTTP/1.1 200 OK
Content-Length: [bytes in body]
CONTENT-TYPE: text/xml; charset="utf-8"
SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0
EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.12.34 F33

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
 HOST: 192.168.1.11:10011
 CONTENT-TYPE: text/xml
 Content-Length: [bytes in body]
 NT: upnp:event
 NTS: upnp:propchange
 SID: uuid:c3580410-1dd1-11b2-8000-10001
 SEQ: [sequence number]
 Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
    &lt;cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
      &lt;callID&gt;call101&lt;/callID&gt;
      &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
      &lt;callStatus&gt;Talking&lt;/callStatus&gt;
      &lt;priority&gt;Normal&lt;/priority&gt;
      &lt;remoteParty&gt;
        &lt;peer:id&gt;0774940201&lt;/peer:id&gt;
      &lt;/remoteParty&gt;
      &lt;TCList&gt;
        &lt;TC&gt;
          &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn101&lt;/UDN&gt;
          &lt;mediaSessionID&gt;media101&lt;/mediaSessionID&gt;
        &lt;/TC&gt;
      &lt;/TCList&gt;
      &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
    &lt;/cams:callInfo&gt;</CallInfo>
  </e:property>
</e:propertyset>
```

D.12.35 F34

HTTP/1.1 200 OK
 Content-Length: 0

D.12.36 F35

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
 HOST: 192.168.2.11:20011
 CONTENT-TYPE: text/xml

ISO/IEC 29341-26-10:2017(E)

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20001

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
      &lt;cams:callInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
          http://www.upnp.org/schemas/phone/cams-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:cams="urn:schemas-upnp-org:phone:cams"
        xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
        &lt;callID&gt;&lt;callID&gt;
        &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
        &lt;callStatus&gt;Talking&lt;/callStatus&gt;
        &lt;priority&gt;Normal&lt;/priority&gt;
        &lt;remoteParty&gt;
          &lt;peer:id&gt;0355550101&lt;/peer:id&gt;
        &lt;/remoteParty&gt;
        &lt;TCList&gt;
          &lt;TC&gt;
          &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn201&lt;/UDN&gt;
          &lt;mediaSessionID&gt;media201&lt;/mediaSessionID&gt;
          &lt;/TC&gt;
        &lt;/TCList&gt;
        &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
  </e:property>
</e:propertyset>
```

D.12.37 F36

HTTP/1.1 200 OK

Content-Length: 0

D.13 Terminate a Call

D.13.1 Figure

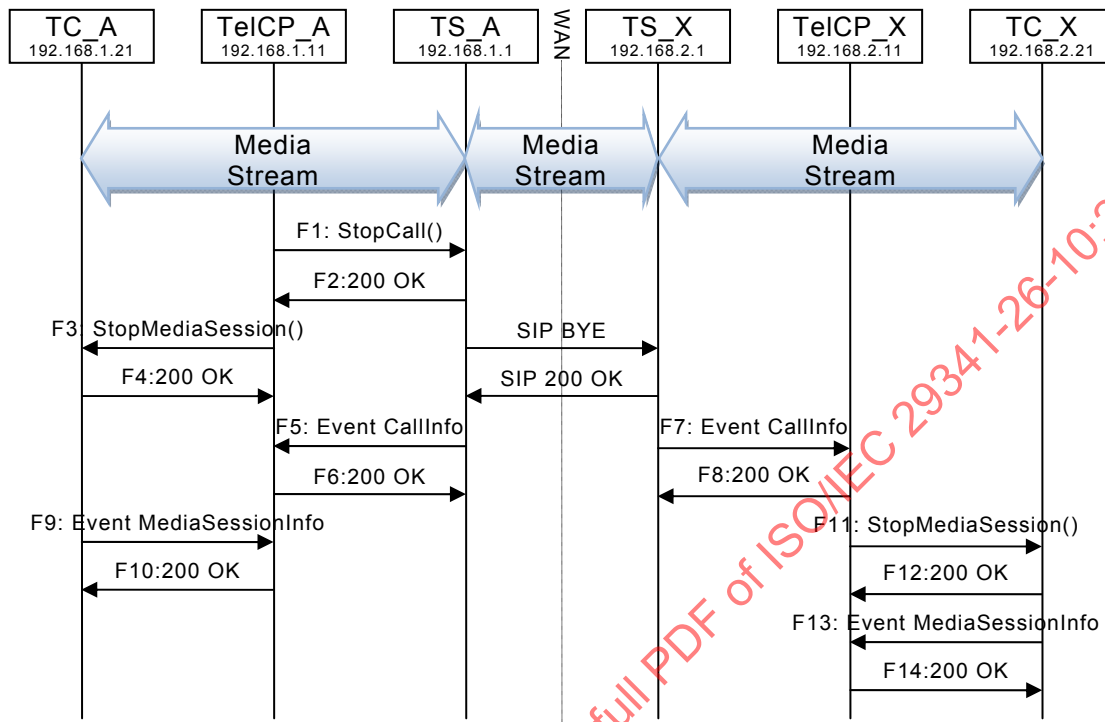


Figure D.13 — Terminate a Call

D.13.2 F1

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StopCall"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName></TelCPName>
      <SecretKey></SecretKey>
      <CallID>call101</CallID>
    </u:StopCall>
  </s:Body>
</s:Envelope>
  
```

D.13.3 F2

HTTP/1.1 200 OK
 EXT:
 CONTENT-TYPE: text/xml ; charset="utf-8"
 SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopCallResponse xmlns:u="urn:schemas-upnp-org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
  
```

ISO/IEC 29341-26-10:2017(E)

```
</s:Body>
</s:Envelope>
```

D.13.4 F3

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StopMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.21:10021
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TargetMediaSessionID>media101</TargetMediaSessionID>
    </u:StopMediaSession>
  </s:Body>
</s:Envelope>
```

D.13.5 F4

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2" />
  </s:Body>
</s:Envelope>
```

D.13.6 F5

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10001
SEQ: [sequence number]
Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo><?xml version="1.0" encoding="utf-8"?>
      <callInfo xmlns:cams="urn:schemas-upnp-org:phone:cams"
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams"
xmlns:peer="urn:schemas-upnp-org:phone:peer">
      <callID>call101</callID>
      <targetNames type="TelCPName">*</targetNames>
      <callStatus>Disconnected</callStatus>
      <priority>Normal</priority>
      <remoteParty>
        <peer:id>ID of the Peer</peer:id>
      </remoteParty>
      <TCList>
```

```

<TC>
  <UDN>uuid:97ef6efa-ac89-4ea2-0001-udn101</UDN>
  <mediaSessionID>media101</mediaSessionID>
</TC>
</TCList>
<mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
</mediaCapability>
</cams:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.13.7 F6

HTTP/1.1 200 OK
Content-Length: 0

D.13.8 F7

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.2.11:20011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-20001
SEQ: [sequence number]
Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <callID>call201</callID>
  <targetNames type="TelCPName">*</targetNames>
  <callStatus>Disconnected</callStatus>
  <priority>Normal</priority>
  <remoteParty>
    <peer:id>0355550101</peer:id>
  </remoteParty>
  <TCList>
    <TC>
      <UDN>uuid:97ef6efa-ac89-4ea2-0001-udn201</UDN>
      <mediaSessionID>media201</mediaSessionID>
    </TC>
  </TCList>
  <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20

```

ISO/IEC 29341-26-10:2017(E)

```
</mediaCapability>
</cams:callInfo></CallInfo>
</e:property>
</e:propertyset>
```

D.13.9 F8

HTTP/1.1 200 OK
Content-Length: 0

D.13.10 F9

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10021
SEQ: [sequence number]
Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <MediaSessionInfo><?xml version="1.0" encoding="utf-8"?>
      <?xml:mediaSessionInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
          http://www.upnp.org/schemas/phone/mms-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:mms="urn:schemas-upnp-org:phone:mms">
        <mediaSessionID>media101</mediaSessionID>
        <mediaSessionStatus>Stopped</mediaSessionStatus>
        <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.21
s=-
c=IN IP4 192.168.1.21
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
      </mediaCapability>
    </mediaSessionInfo></MediaSessionInfo>
  </e:property>
</e:propertyset>
```

D.13.11 F10

HTTP/1.1 200 OK
Content-Length: 0

D.13.12 F11

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StopMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.21:20021
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TargetMediaSessionID>media201</TargetMediaSessionID>
    </u:StopMediaSession>
  </s:Body>
</s:Envelope>
```

D.13.13 F12

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2" />
  </s:Body>
</s:Envelope>
```

D.13.14 F13

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20021

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <MediaSessionInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaSessionInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
    &lt;mediaSessionID&gt;media201&lt;/mediaSessionID&gt;
    &lt;mediaSessionStatus&gt;Stopped&lt;/mediaSessionStatus&gt;
    &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.21
s=-
c=IN IP4 192.168.2.21
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/mms:mediaSessionInfo&gt;</MediaSessionInfo>
  </e:property>
</e:propertyset>
```

D.13.15 F14

HTTP/1.1 200 OK

Content-Length: 0

D.14 Reject an incoming Call

D.14.1 Figure

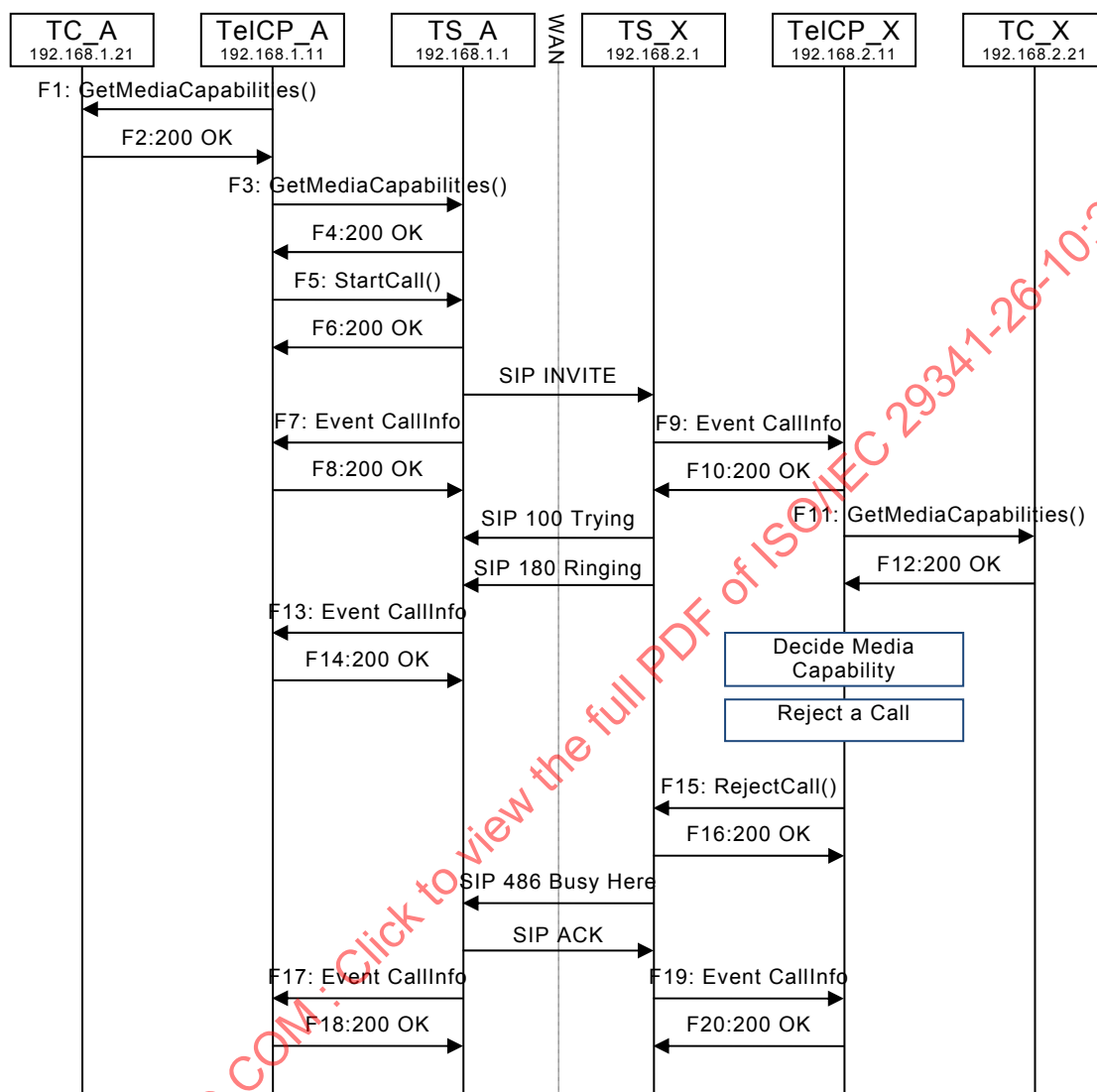


Figure D.14 — Reject an incoming Call

D.14.2 F1 to F14 are the same as “D.12 Create a Call”

D.14.3 F15

POST /urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#RejectCall"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.2.1:20001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:RejectCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName></TelCPName>
      <SecretKey></SecretKey>
      <TargetCallID>call1201</TargetCallID>
      <RejectReason>Busy</RejectReason>
    </u:RejectCall>
  </s:Body>
</s:Envelope>
  
```

```

    </u:RejectCall>
  </s:Body>
</s:Envelope>

```

D.14.4 F16

HTTP/1.1 200 OK
 Content-Length: [bytes in body]
 CONTENT-TYPE: text/xml; charset="utf-8"
 SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0
 EXT:

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:RejectCallResponse xmlns:m="urn:schemas-upnp-org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>

```

D.14.5 F17

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
 HOST: 192.168.1.11:10011
 CONTENT-TYPE: text/xml
 Content-Length: [bytes in body]
 NT: upnp:event
 NTS: upnp:propchange
 SID: uuid:c3580410-1dd1-11b2-8000-10001
 SEQ: [sequence number]
 Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
    &lt;cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
      http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
      &lt;callID&gt;call1101&lt;/callID&gt;
      &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
      &lt;callStatus reason="Busy"&gt;Disconnected&lt;/callStatus&gt;
      &lt;priority&gt;Normal&lt;/priority&gt;
      &lt;remoteParty&gt;
        &lt;peer:id&gt;0774940201&lt;/peer:id&gt;
      &lt;/remoteParty&gt;
      &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
  </e:property>
</e:propertyset>

```

D.14.6 F18

HTTP/1.1 200 OK
 Content-Length: 0

ISO/IEC 29341-26-10:2017(E)

D.14.7 F19

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20001

SEQ: [sequence number]

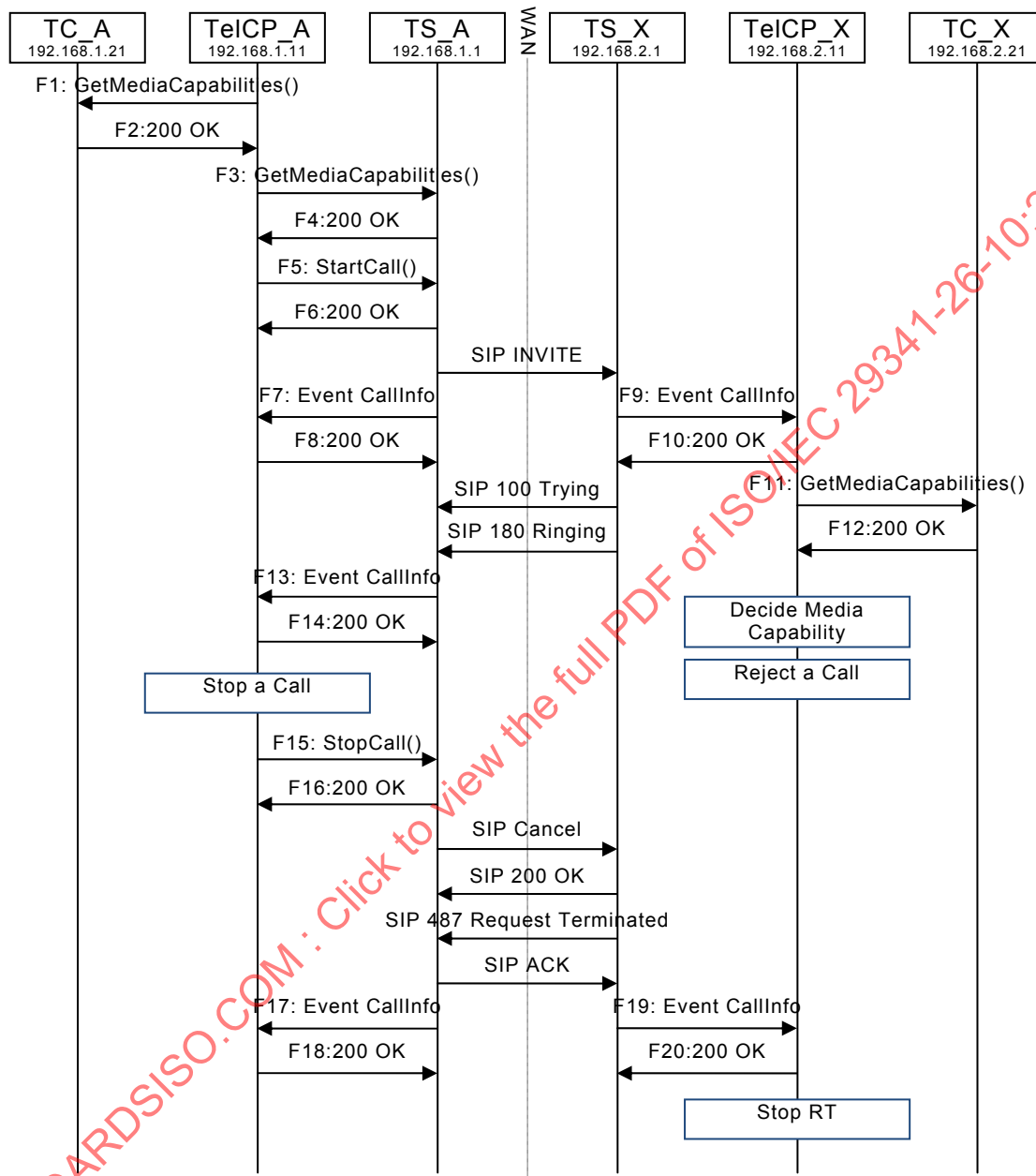
Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
      &lt;cams:callInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
          http://www.upnp.org/schemas/phone/cams-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:cams="urn:schemas-upnp-org:phone:cams"
        xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
        &lt;callID&gt;call1101&lt;/callID&gt;
        &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
        &lt;callStatus reason="Busy"&gt;Disconnected&lt;/callStatus&gt;
        &lt;priority&gt;Normal&lt;/priority&gt;
        &lt;remoteParty&gt;
          &lt;peer:id&gt;035550101&lt;/peer:id&gt;
        &lt;/remoteParty&gt;
        &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
    </e:property>
  </e:propertyset>
```

D.14.8 F20

HTTP/1.1 200 OK

Content-Length: 0

D.15 Cancel an outgoing Call**D.15.1 Figure****Figure D.15 — Cancel an outgoing Call****D.15.2 F1 to F14 are the same as “D.12 Create a Call”****D.15.3 F15**

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StopCall"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>

```

ISO/IEC 29341-26-10:2017(E)

```
<u:StopCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
  <TelCPName></TelCPName>
  <SecretKey></SecretKey>
  <CallID>call101</CallID>
</u:StopCall>
</s:Body>
</s:Envelope>
```

D.15.4 F16

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<s:Body>
<m:StopCallResponse xmlns:m="urn:schemas-upnp-org:service:CallManagement:2" />
</s:Body>
</s:Envelope>
```

D.15.5 F17

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10001

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
    &lt;callID&gt;call101&lt;/callID&gt;
    &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
    &lt;callStatus reason="Cancelled"&gt;Disconnected&lt;/callStatus&gt;
    &lt;priority&gt;Normal&lt;/priority&gt;
    &lt;remoteParty&gt;
      &lt;peer.id&gt;0774940201&lt;/peer.id&gt;
    &lt;/remoteParty&gt;
    &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
</e:property>
</e:propertyset>
```

D.15.6 F18

HTTP/1.1 200 OK
Content-Length: 0

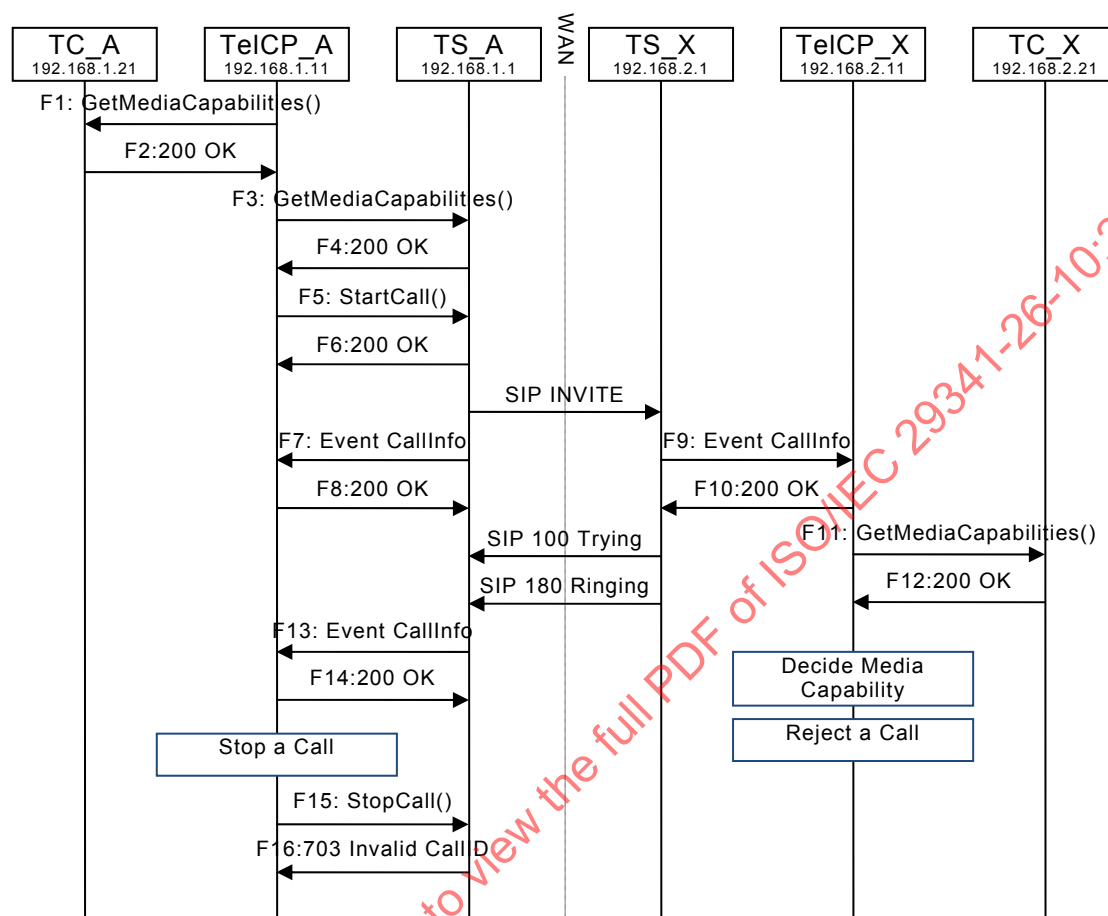
D.15.7 F19

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.2.11:20011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-20001
SEQ: [sequence number]
Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<?xml:namespace prefix="c" uri="urn:schemas-upnp-org:phone:cams" />
<?xml:namespace prefix="xsi" uri="http://www.w3.org/2001/XMLSchema-instance" />
<?xml:namespace prefix="peer" uri="urn:schemas-upnp-org:phone:peer" />
<callID><?xml:namespace prefix="c" uri="urn:schemas-upnp-org:phone:cams" />
<callID><?xml:namespace prefix="peer" uri="urn:schemas-upnp-org:phone:peer" />
<targetNames type="TelCPName" />
<callStatus reason="Cancelled" />
<priority>Normal</priority>
<remoteParty>
  <peer:id>0355550101</peer:id>
</remoteParty>
<mediaCapability format="SDP" />
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
</mediaCapability>
</c:callInfo></CallInfo>
</e:property>
</e:propertyset>
```

D.15.8 F20

HTTP/1.1 200 OK
Content-Length: 0

D.16 Cancel an outgoing Call (But the specified CallID does not exist.)**D.16.1 Figure****Figure D.16 — Cancel an outgoing Call (But the specified CallID does not exist.)****D.16.2 F1 to F14 are the same as “D.12 Create a Call”****D.16.3 F15**

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StopCall"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <SecretKey>secret101</SecretKey>
      <CallID>wrongcallid</CallID>
    </u:StopCall>
  </s:Body>
</s:Envelope>
```

D.16.4 F16

HTTP/1.1 500 Internal Server Error
 Content-Length: [bytes in body]

```

<s:Envelope
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <s:Fault>
      <faultcode>s:Client</faultcode>
      <faultstring>UPnPError</faultstring>
      <detail>
        <UPnPError xmlns="urn:schemas-upnp-org:control-1-0">
          <errorCode>703</errorCode>
          <errorDescription>Invalid CallID</errorDescription>
        </UPnPError>
      </detail>
    </s:Fault>
  </s:Body>
</s:Envelope>

```

D.17 Cancel an outgoing Call (But the StopCall() action is invoked at invalid timing.)

D.17.1 Figure

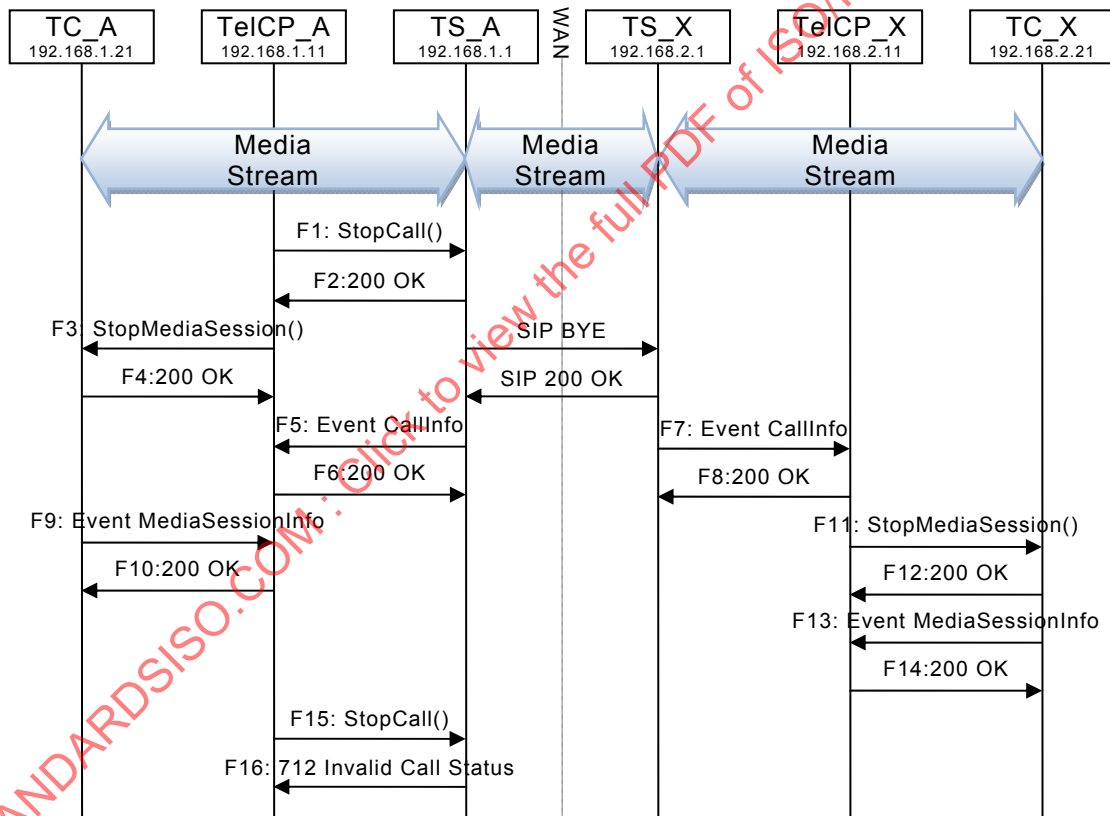


Figure D.17 — Cancel an outgoing Call (But the StopCall() action is invoked at invalid timing.)

D.17.2 F1 to F14 are the same as “D.13 Terminate a Call”

D.17.3 F15

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StopCall"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.1:10001
 Content-Length: [bytes in body]

ISO/IEC 29341-26-10:2017(E)

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName></TelCPName>
      <SecretKey></SecretKey>
      <CallID>call101</CallID>
    </u:StopCall>
  </s:Body>
</s:Envelope>
```

D.17.4 F16

HTTP/1.1 500 Internal Server Error
Content-Length: [bytes in body]

```
<s:Envelope
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <s:Fault>
      <faultcode>s:Client</faultcode>
      <faultstring>UPnPError</faultstring>
      <detail>
        <UPnPError xmlns="urn:schemas-upnp-org:control-1-0">
          <errorCode>712</errorCode>
          <errorDescription>Invalid Call Status</errorDescription>
        </UPnPError>
      </detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
```

D.18 Get MediaSessionInfo and CallInfo during the Call

D.18.1 Figure

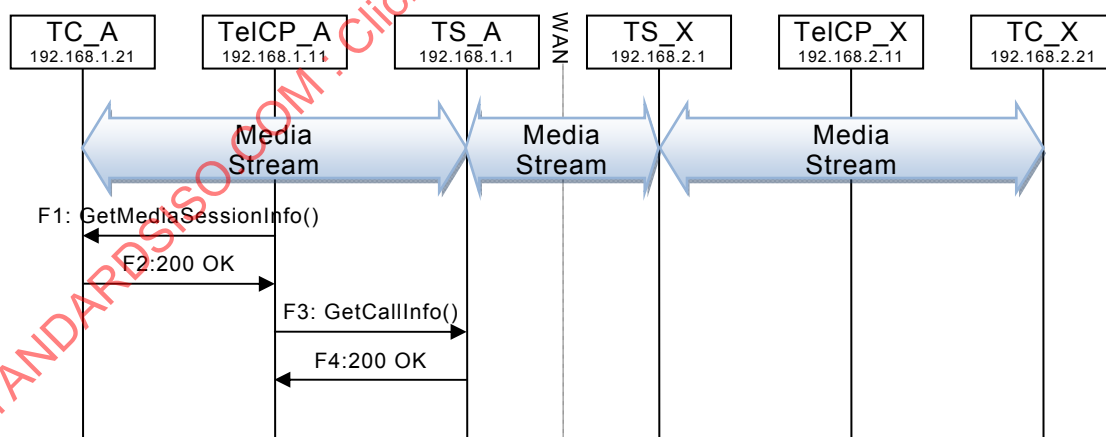


Figure D.18 — Get MediaSessionInfo and CallInfo during the Call

D.18.2 F1

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaSessionInfo"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.21:10021
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaSessionInfo xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2" />
      <TargetMediaSessionID>media101</TargetMediaSessionID>
    </u:GetMediaSessionInfo>
  </s:Body>
</s:Envelope>
```

D.18.3 F2

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaSessionInfoResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <MediaSessionInfoList>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mss:mediaSessionInfoList xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaSessionInfo&gt;
  &lt;mediaSessionID&gt;media101&lt;/mediaSessionID&gt;
  &lt;mediaSessionStatus&gt;Started&lt;/mediaSessionStatus&gt;
  &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.21
s=-
c=IN IP4 192.168.1.21
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/mediaSessionInfo&gt;
&lt;/mms:mediaSessionInfo&gt;</MediaSessionInfoList>
    </u:GetMediaSessionInfoResponse>
  </s:Body>
</s:Envelope>
```

D.18.4 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#GetCallInfo"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetCallInfo xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName></TelCPName>
      <SecretKey></SecretKey>
      <TargetCallID>call1101</TargetCallID>
    </u:GetCallInfo>
  </s:Body>
</s:Envelope>
```

ISO/IEC 29341-26-10:2017(E)

D.18.5 F4

HTTP/1.1 200 OK

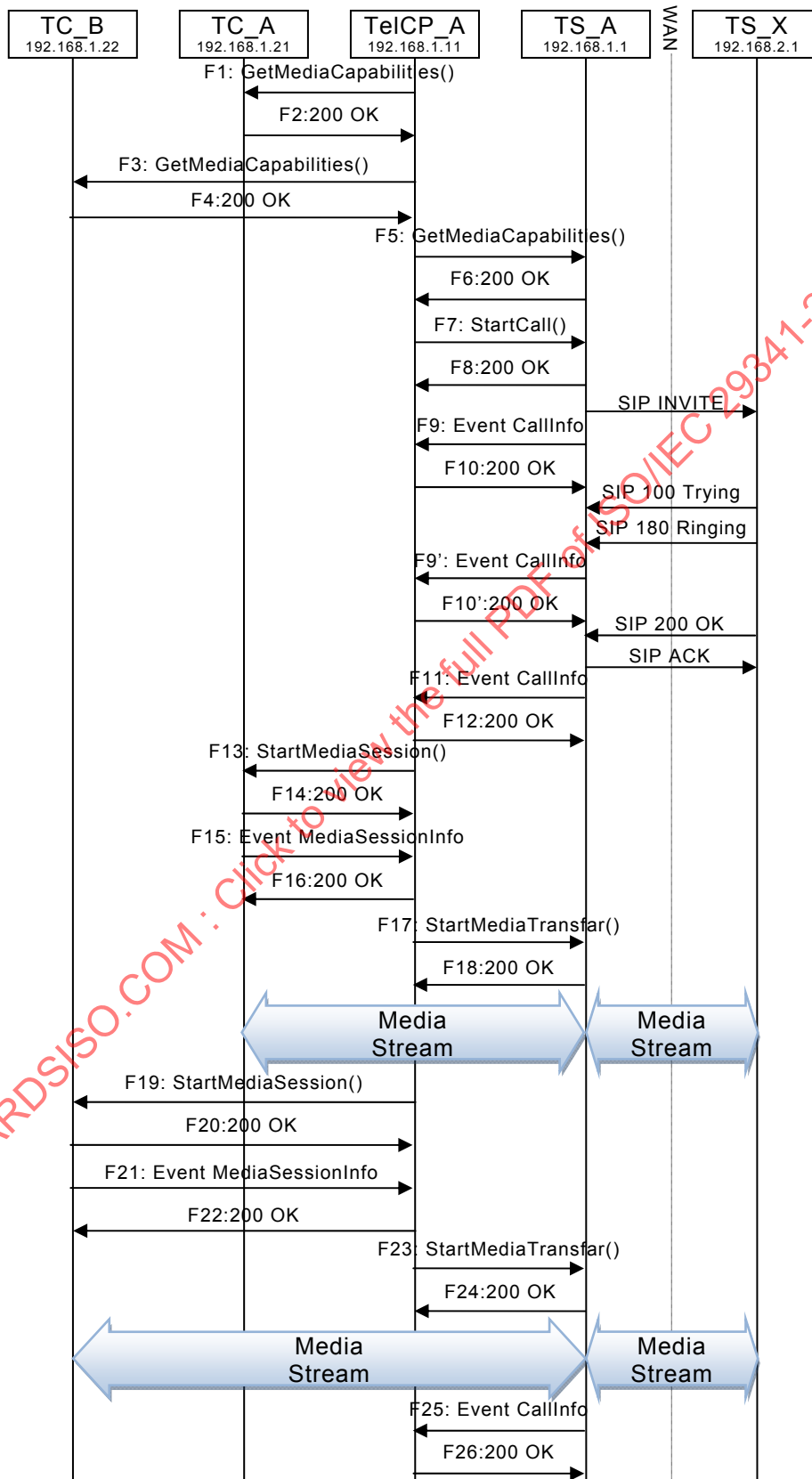
Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetCallInfoResponse xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <callInfoList>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:callInfoList
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams"
xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
  &lt;callInfo&gt;
    &lt;callID&gt;call101&lt;/callID&gt;
    &lt;TelCPNames&gt;&lt;/TelCPNames&gt;
    &lt;callStatus&gt;Talking &lt;/callStatus&gt;
    &lt;priority&gt;"Normal"&lt;/priority&gt;
    &lt;peer&gt;
      &lt;peer:id&gt;0774940201&lt;/peer:id&gt;
    &lt;/peer&gt;
    &lt;TCList&gt;
      &lt;TC&gt;
        &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn101&lt;/UDN&gt;
      &lt;/TC&gt;
    &lt;/TCList&gt;
    &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/callInfo&gt;
&lt;/cams:callInfoList&gt;</callInfoList>
</m:GetCallInfoResponse>
</s:Body>
</s:Envelope>
```

D.19 Create an outgoing Call using multiple TCs (Case 1)**D.19.1 Figure****Figure D.19 — Create an outgoing Call using multiple TCs (Case 1)**

ISO/IEC 29341-26-10:2017(E)

D.19.2 F1

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaCapabilities"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.21:10021
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo></TSMediaCapabilityInfo>
    </u:GetMediaCapabilities>
  </s:Body>
</s:Envelope>
```

D.19.3 F2

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <SupportedMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <t;mediaCapability format="SDP">
            o=- 0 0 IN IP4 192.168.1.21
            s=-
            c=IN IP4 192.168.1.21
            t=0 0
            m=audio 11021 RTP/AVP 0
            c=IN IP4 192.168.1.21
            a=rtpmap:0 PCMU/8000
            aptime:20
            m=video 12021 RTP/AVP 96
            c=IN IP4 192.168.1.21
            b=AS:2500
            a=rtpmap:96 MP4V-ES/90000
            a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
            a=framerate:30
          </t;mediaCapability>
        </t;mms:mediaCapabilityInfo>
      </SupportedMediaCapabilityInfo>
    </u:GetMediaCapabilitiesResponse>
  </s:Body>
</s:Envelope>
```

D.19.4 F3

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaCapabilities"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.22:10022
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2" >
      <TSMediaCapabilityInfo></TSMediaCapabilityInfo>
    </u:GetMediaCapabilities>
  </s:Body>
</s:Envelope>
```

D.19.5 F4

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <SupportedMediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.22
s=-
c=IN IP4 192.168.1.22
t=0 0
m=audio 11022 RTP/AVP 0
c=IN IP4 192.168.1.22
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/mms:mediaCapabilityInfo&gt;
&lt;/SupportedMediaCapabilityInfo>
    </u:GetMediaCapabilitiesResponse>
  </s:Body>
</s:Envelope>
```

D.19.6 F5

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#GetMediaCapabilities"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service: CallManagement:2">
      <TCMediaCapabilityInfo></TCMediaCapabilityInfo>
    </u:GetMediaCapabilities>
```

ISO/IEC 29341-26-10:2017(E)

```
</s:Body>
</s:Envelope>
```

D.19.7 F6

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:CallManagement:2">
      <SupportedMediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/cams:mediaCapabilityInfo&gt;</SupportedMediaCapabilityInfo>
    </u:GetMediaCapabilitiesResponse>
  </s:Body>
</s:Envelope>
```

D.19.8 F7

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartCall"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <SecretKey>secret101</SecretKey>
      <CalleeID>tel:0774940201</CalleeID>
      <CallPriority>Normal</CallPriority>
      <MediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;

```

```

    <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
    </mediaCapability>
    <cams:mediaCapabilityInfo></MediaCapabilityInfo>
    <CallMode>Non-Monopolize</CallMode>
  </u:StartCall>
</s:Body>
</s:Envelope>

```

D.19.9 F8

HTTP/1.1 200 OK
 Content-Length: [bytes in body]
 CONTENT-TYPE: text/xml; charset="utf-8"
 SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0
 EXT:

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartCallResponse xmlns:m="urn:schemas-upnp-org:service:CallManagement:2">
      <CallID>call101</CallID>
    </m:StartCallResponse>
  </s:Body>
</s:Envelope>

```

D.19.10 F9

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
 HOST: 192.168.1.11:10011
 CONTENT-TYPE: text/xml
 Content-Length: [bytes in body]
 NT: upnp:event
 NTS: upnp:propchange
 SID: uuid:c3580410-1dd1-11b2-8000-10001
 SEQ: [sequence number]
 Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo><?xml version="1.0" encoding="utf-8"?>
    <cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
      http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer">
      <callID>call101</callID>
      <targetNames type="TelCPName">*</targetNames>
      <callStatus>Dialing</callStatus>
      <priority>Normal</priority>
      <remoteParty>

```

ISO/IEC 29341-26-10:2017(E)

```
<peer:id>0774940201</peer:id>
</remoteParty>
<mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30</mediaCapability>
</cams:callInfo></CallInfo>
</e:property>
</e:propertyset>
```

D.19.11 F10

HTTP/1.1 200 OK
Content-Length: 0

D.19.12 F9'

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10001
SEQ: [sequence number]
Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<callInfo>
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
    <callID>call101</callID>
    <targetNames type="TelCPName">*</targetNames>
    <callStatus>Calling</callStatus>
    <priority>Normal</priority>
    <remoteParty>
      <peer:id>0774940201</peer:id>
    </remoteParty>
    <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
```

```
HTTP/1.1 200 OK
Content-Length: 0
```

D.19.14 F11

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
    &lt;callID&gt;call1101&lt;/callID&gt;
    &lt;targetNames type="TelCPName"&gt;&lt;/targetNames&gt;
    &lt;callStatus&gt;Connected&lt;/callStatus&gt;
    &lt;priority&gt;Normal&lt;/priority&gt;
    &lt;remoteParty&gt;
      &lt;peer:id&gt;0774940201&lt;/peer:id&gt;
    &lt;/remoteParty&gt;
    &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa2
a=framerate:30&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
</e:property>
</e:propertyset>
```

HTTP/1.1 200 OK
Content-Length: 0

ISO/IEC 29341-26-10:2017(E)

D.19.16 F13

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.21:10021
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <t:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
          </t:mediaCapability>
        </t:mms:mediaCapabilityInfo>
      </TSMediaCapabilityInfo>
    </u:StartMediaSession>
  </s:Body>
</s:Envelope>
```

D.19.17 F14

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <MediaSessionID>media101</MediaSessionID>
      <TCMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <t:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.21
s=-
c=IN IP4 192.168.1.21
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
          </t:mediaCapability>
        </t:mms:mediaCapabilityInfo>
      </TCMediaCapabilityInfo>
```

```

</u:StartMediaSessionResponse>
</s:Body>
</s:Envelope>

```

D.19.18 F15

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10021

SEQ: [sequence number]

Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<MediaSessionInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaSessionInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
    http://www.upnp.org/schemas/phone/mms-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
    &lt;mediaSessionID&gt;media101&lt;/mediaSessionID&gt;
    &lt;mediaSessionStatus&gt;Started&lt;/mediaSessionStatus&gt;
    &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.21
s=-
c=IN IP4 192.168.1.21
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/mms:mediaSessionInfo&gt;</MediaSessionInfo>
</e:property>
</e:propertyset>

```

D.19.19 F16

HTTP/1.1 200 OK

Content-Length: 0

D.19.20 F17

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
<TelCPName>TV@Living</TelCPName>
<SecretKey>secret101</SecretKey>
<TargetCallID>call101</TargetCallID>
<TCList>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:TCList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;

```

ISO/IEC 29341-26-10:2017(E)

```
<TC>
  <UDN>uuid:97ef6efa-ac89-4ea2-0001-udn101</UDN>
  <mediaSessionID>media101</mediaSessionID>
</TC>
</cams:TCList>
<MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
  <cams:mediaCapabilityInfo
    xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
      http://www.upnp.org/schemas/phone/cams-v2.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:cams="urn:schemas-upnp-org:phone:cams">
    <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
  </mediaCapability >
  </cams:mediaCapabilityInfo>
</MediaCapabilityInfo>
  </u:StartMediaTransfer>
</s:Body>
</s:Envelope>
```

D.19.21 F18

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
      org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.19.22 F19

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"

CONTENT-TYPE: text/xml; charset="utf-8"

HOST: 192.168.1.22:10022

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
            http://www.upnp.org/schemas/phone/mms-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=video 12001 RTP/AVP 96
```

```

c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</mms:mediaCapabilityInfo>
</TSMediaCapabilityInfo>
</u:StartMediaSession>
</s:Body>
</s:Envelope>

```

D.19.23 F20

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
<MediaSessionID>media102</MediaSessionID>
<TCMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
<t:mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
<t:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.22
s=-
c=IN IP4 192.168.1.22
t=0 0
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</mms:mediaCapabilityInfo></TCMediaCapabilityInfo>
</u:StartMediaSessionResponse>
</s:Body>
</s:Envelope>

```

D.19.24 F21

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10022

SEQ: [sequence number]

Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<MediaSessionInfo><?xml version="1.0" encoding="utf-8"?>

```

ISO/IEC 29341-26-10:2017(E)

```
<?xml version="1.0" encoding="utf-8"?>
<mms:mediaSessionInfo xmlns:mms="urn:schemas-upnp-org:phone:mms"
  xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
    http://www.upnp.org/schemas/phone/mms-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  <mediaSessionID>media102</mediaSessionID>
  <mediaSessionStatus>Started</mediaSessionStatus>
  <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.22
s=-
c=IN IP4 192.168.1.22
t=0 0
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</mms:mediaSessionInfo></MediaSessionInfo>
</e:property>
</e:propertyset>
```

D.19.25 F22

HTTP/1.1 200 OK
Content-Length: 0

D.19.26 F23

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.1:10001
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <SecretKey>secret101</SecretKey>
      <TargetCallID>call101</TargetCallID>
      <TCList><?xml version="1.0" encoding="utf-8"?>
        <cams:TCList
          xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
            http://www.upnp.org/schemas/phone/cams-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:cams">
            <TC>
              <UDN>97ef6efa-ac89-4ea2-0001-udn102</UDN>
              <mediaSessionID>media102</mediaSessionID>
            </TC>
          </cams:TCList></TCList>
        <MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
          <cams:mediaCapabilityInfo
            xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
              http://www.upnp.org/schemas/phone/cams-v2.xsd"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:cams="urn:schemas-upnp-org:phone:cams">
              <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=video 12022 RTP/AVP 96
```

```

c=IN IP4 192.168.1.22
b=AS:2500
a=rtmpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</cams:mediaCapabilityInfo></MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
</s:Envelope>

```

D.19.27 F24

HTTP/1.1 200 OK
Content-Length: [bytes in body]
CONTENT-TYPE: text/xml; charset="utf-8"
SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0
EXT:

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>

```

D.19.28 F25

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10001
SEQ: [sequence number]
Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <callID><callID101></callID>
  <targetNames type="TelCPName">*</targetNames>
  <callStatus>Talking</callStatus>
  <priority>Normal</priority>
  <remoteParty>
    <peer:id>0774940201</peer:id>
  </remoteParty>
  <TCList>
    <TC>
      <UDN>97ef6efa-ac89-4ea2-0001-udn101</UDN>
      <mediaSessionID>media101</mediaSessionID>
    </TC>
    <TC>
      <UDN>97ef6efa-ac89-4ea2-0001-udn102</UDN>
      <mediaSessionID>media102</mediaSessionID>
    </TC>
  </TCList>

```

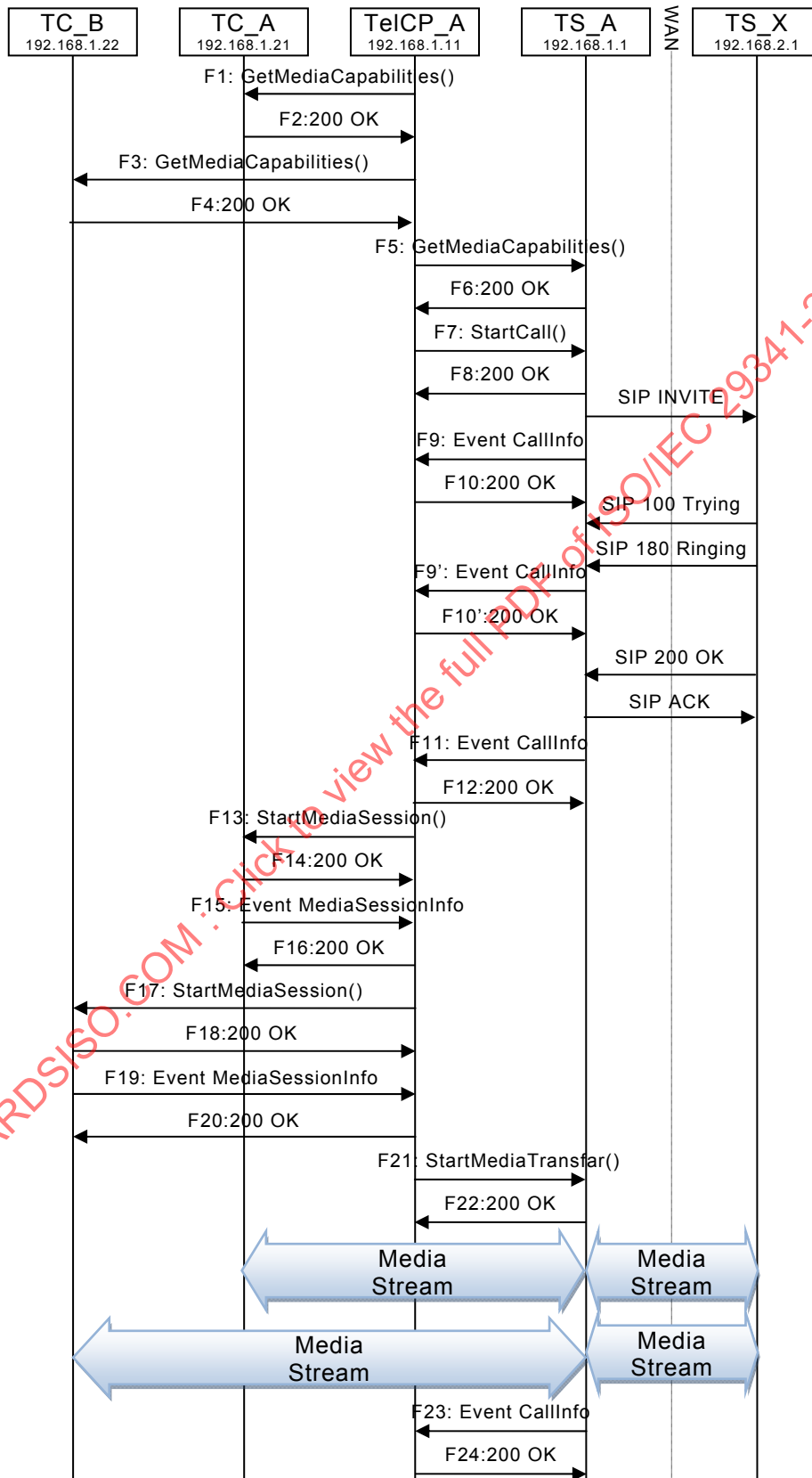
ISO/IEC 29341-26-10:2017(E)

```
<mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
<cams:callInfo></CallInfo>
</e:property>
</e:propertyset>
```

D.19.29 F26

HTTP/1.1 200 OK
Content-Length: 0

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 29341-26-10:2017

D.20 Create an outgoing Call using multiple TCs (Case 2)**D.20.1 Figure****Figure D.20 — Create an outgoing Call using multiple TCs (Case 2)**

ISO/IEC 29341-26-10:2017(E)

D.20.2 F1 to F16 are the same as “D.19 Create an outgoing Call using multiple TCs (Case 1)”

D.20.3 F17

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.22:10022
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <t:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
          </t:mediaCapability>
        </t:mms:mediaCapabilityInfo>
      </TSMediaCapabilityInfo>
    </u:StartMediaSession>
  </s:Body>
</s:Envelope>
```

D.20.4 F18

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <MediaSessionID>media102</MediaSessionID>
      <TCMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <t:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.22
s=-
c=IN IP4 192.168.1.22
t=0 0
m=video 12022 RTP/AVP 96
```

```
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</mms:mediaCapabilityInfo></TCMediaCapabilityInfo>
</u:StartMediaSessionResponse>
</s:Body>
</s:Envelope>
```

```
NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10022
SEQ: [sequence number]
Connection: Keep-Alive
```

D.20.6 F20

D.20.7 F21

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
```

ISO/IEC 29341-26-10:2017(E)

```
<s:Body>
<u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
<TelCPName>TV@Living</TelCPName>
<SecretKey>secret101</SecretKey>
<TargetCallID>call101</TargetCallID>
<TCList><?xml version="1.0" encoding="utf-8"?>
&lt;cams:TCList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
    &lt;TC>
      &lt;UDN>uid:97ef6efa-ac89-4ea2-0001-udn101&lt;/UDN>
      &lt;mediaSessionID>media101&lt;/mediaSessionID>
    &lt;/TC>
  &lt;/cams:TCList></TCList>
<MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
&lt;cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
    &lt;mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability>
&lt;/cams:mediaCapabilityInfo></MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
</s:Envelope>
```

D.20.8 F22

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.20.9 F23

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.1.11:10011
 CONTENT-TYPE: text/xml
 Content-Length: [bytes in body]
 NT: upnp:event
 NTS: upnp:propchange
 SID: uuid:c3580410-1dd1-11b2-8000-10001
 SEQ: [sequence number]
 Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
    &lt;callID&gt;call101&lt;/callID&gt;
    &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
    &lt;callStatus&gt;Talking&lt;/callStatus&gt;
    &lt;priority&gt;Normal&lt;/priority&gt;
    &lt;remoteParty&gt;
      &lt;peer:id&gt;0774940201&lt;/peer:id&gt;
    &lt;/remoteParty&gt;
    &lt;TCList&gt;
      &lt;TC&gt;
        &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn101&lt;/UDN&gt;
        &lt;mediaSessionID&gt;media101&lt;/mediaSessionID&gt;
      &lt;/TC&gt;
      &lt;TC&gt;
        &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn102&lt;/UDN&gt;
        &lt;mediaSessionID&gt;media102&lt;/mediaSessionID&gt;
      &lt;/TC&gt;
    &lt;/TCList&gt;
    &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
</e:property>
</e:propertyset>
```

D.20.10 F24

HTTP/1.1 200 OK
 Content-Length: 0

D.21 Accept an incoming Call using multiple TCs (Case 1)

D.21.1 Figure

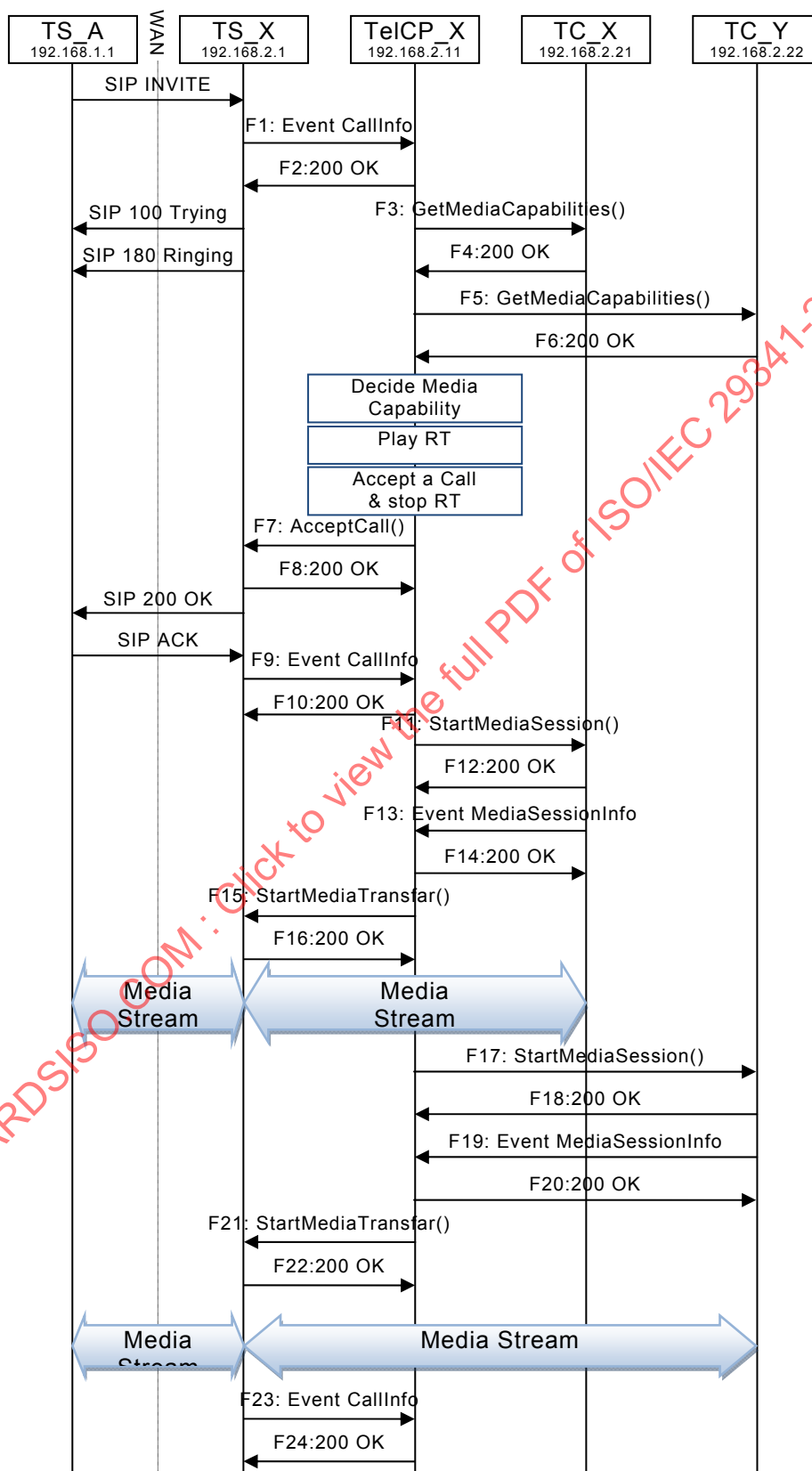


Figure D.21 — Accept an incoming Call using multiple TCs (Case 1)

D.21.2 F1

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
 HOST: 192.168.2.11:20011
 CONTENT-TYPE: text/xml
 Content-Length: [bytes in body]
 NT: upnp:event
 NTS: upnp:propchange
 SID: uuid:c3580410-1dd1-11b2-8000-20001
 SEQ: [sequence number]
 Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
      &lt;cams:callInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
          http://www.upnp.org/schemas/phone/cams-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:cams="urn:schemas-upnp-org:phone:cams"
        xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
          &lt;callID&gt;call1201&lt;/callID&gt;
          &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
          &lt;callStatus&gt;Ringing&lt;/callStatus&gt;
          &lt;priority&gt;Normal&lt;/priority&gt;
          &lt;remoteParty&gt;
            &lt;peer:id&gt;0355550101&lt;/peer:id&gt;
          &lt;/remoteParty&gt;
          &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22001 RTP/AVP 96
c=IN IP4 192.168.2.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
  </e:property>
</e:propertyset>
```

D.21.3 F2

HTTP/1.1 200 OK
 Content-Length: 0

D.21.4 F3

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaCapabilities"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.2.21:20021
 Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2" >
```

ISO/IEC 29341-26-10:2017(E)

```
<TSMediaCapabilityInfo></TSMediaCapabilityInfo>
</u:GetMediaCapabilities>
</s:Body>
</s:Envelope>
```

D.21.5 F4

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <SupportedMediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.21
s=-
c=IN IP4 192.168.2.21
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22021 RTP/AVP 96
c=IN IP4 192.168.2.21
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/mms:mediaCapabilityInfo&gt;</SupportedMediaCapabilityInfo>
    </u:GetMediaCapabilitiesResponse>
  </s:Body>
</s:Envelope>
```

D.21.6 F5

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaCapabilities"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.2.22:20022

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2" >
    <TSMediaCapabilityInfo></TSMediaCapabilityInfo>
  </u:GetMediaCapabilities>
</s:Envelope>
```

D.21.7 F6

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <SupportedMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?&gt;
        <t;mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:mms"&gt;
          <t;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.22
s=-
c=IN IP4 192.168.2.22
t=0 0
m=audio 21022 RTP/AVP 0
c=IN IP4 192.168.2.22
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
          &t;/mediaCapability &gt;
        </t;mms:mediaCapabilityInfo&gt;</SupportedMediaCapabilityInfo>
      </u:GetMediaCapabilitiesResponse>
    </s:Body>
  </s:Envelope>
```

D.21.8 F7

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#AcceptCall"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.1:20001
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:AcceptCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>phone@Bedroom</TelCPName>
      <SecretKey>secret201</SecretKey>
      <TargetCallID>call1201</TargetCallID>
      <MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?&gt;
        &t;cams:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
          <t;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
```

ISO/IEC 29341-26-10:2017(E)

```
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
<!--mediaCapability-->
<!--cams:mediaCapabilityInfo-->
</MediaCapabilityInfo>
  <CallMode>Non-Monopolize</CallMode>
</u:AcceptCall>
</s:Body>
</s:Envelope>
```

D.21.9 F8

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:AcceptCallResponse xmlns:m="urn:schemas-upnp-org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.21.10 F9

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20001

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo><!--?xml version="1.0" encoding="utf-8"?>
    <!--cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
      http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer">
      <!--callID-->call201<!--/callID-->
      <!--targetNames type="TelCPName">*-->-->targetNames-->
      <!--callStatus-->Connected<!--/callStatus-->
      <!--priority-->Normal<!--/priority-->
      <!--remoteParty-->
        <!--peer:id-->0355550101<!--/peer:id-->
      <!--/remoteParty-->
      <!--mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
```

```

a=ptime:20
m=video 22001 RTP/AVP 96
c=IN IP4 192.168.2.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
<lt;/mediaCapability>
<lt;/cams:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.21.11 F10

HTTP/1.1 200 OK
Content-Length: 0

D.21.12 F11

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.21:20021
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><lt;?xml version="1.0" encoding="utf-8"?>
        <lt;mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
            http://www.upnp.org/schemas/phone/mms-v2.xsd"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <lt;mediaCapability format="SDP">v=0
            o=- 0 0 IN IP4 192.168.2.11
            s=-
            c=IN IP4 192.168.2.11
            t=0 0
            m=audio 21001 RTP/AVP 0
            c=IN IP4 192.168.2.1
            a=rtpmap:0 PCMU/8000
            a=ptime:20
            <lt;/mediaCapability>
          <lt;/mms:mediaCapabilityInfo>
        </TSMediaCapabilityInfo>
      </u:StartMediaSession>
    </s:Body>
  </s:Envelope>

```

D.21.13 F12

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
      org:service:MediaManagement:2">
      <MediaSessionID>media201</MediaSessionID>
      <TCMediaCapabilityInfo><lt;?xml version="1.0" encoding="utf-8"?>

```

ISO/IEC 29341-26-10:2017(E)

```
<?xml version="1.0" encoding="utf-8"?>
<u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-org:service:mms"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:mms"
  <mediaCapability format="SDP" v=0
  o=- 0 0 IN IP4 192.168.2.21
  s=-
  c=IN IP4 192.168.2.21
  t=0 0
  m=audio 21021 RTP/AVP 0
  c=IN IP4 192.168.2.21
  a=rtpmap:0 PCMU/8000
  a=ptime:20
  </mediaCapability>
  </u:StartMediaSessionResponse>
</s:Body>
</s:Envelope>
```

D.21.14 F13

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20021

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <MediaSessionInfo>
      <?xml version="1.0" encoding="utf-8"?>
      <mms:mediaSessionInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
          http://www.upnp.org/schemas/phone/mms-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:mms="urn:schemas-upnp-org:phone:mms"
        <mediaSessionID media201
        </mediaSessionID
        <mediaSessionStatus Started
        </mediaSessionStatus
        <mediaCapability format="SDP" v=0
        o=- 0 0 IN IP4 192.168.2.21
        s=-
        c=IN IP4 192.168.2.21
        t=0 0
        m=audio 21021 RTP/AVP 0
        c=IN IP4 192.168.2.21
        a=rtpmap:0 PCMU/8000
        a=ptime:20
        </mediaCapability>
        </mms:mediaSessionInfo>
      </e:property>
    </e:propertyset>
```

D.21.15 F14

HTTP/1.1 200 OK

Content-Length: 0

D.21.16 F15

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.2.1:20001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
<TelCPName>phone@Bedroom</TelCPName>
<SecretKey>secret201</SecretKey>
<TargetCallID>call201</TargetCallID>
<TCList>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;t;cams:TCList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
  http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
    &lt;t;TC&gt;
      &lt;t;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn201&lt;/UDN&gt;
      &lt;t;mediaSessionID&gt;media201&lt;/mediaSessionID&gt;
    &lt;/TC&gt;
  &lt;/t;cams:TCList&gt;</TCList>
<MediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;t;cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
  http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
    &lt;t;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/t;cams:mediaCapabilityInfo&gt;</MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
</s:Envelope>
```

D.21.17 F16

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<s:Body>
<m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
</s:Body>
</s:Envelope>
```

D.21.18 F17

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.2.22:20022

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
```

ISO/IEC 29341-26-10:2017(E)

```
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <lt;mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
            http://www.upnp.org/schemas/phone/mms-v2.xsd"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <lt;mediaCapability format="SDP">v=0
            o=- 0 0 IN IP4 192.168.2.11
            s=-
            c=IN IP4 192.168.2.11
            t=0 0
            m=video 22001 RTP/AVP 96
            c=IN IP4 192.168.2.1
            b=AS:2500
            a=rtpmap:96 MP4V-ES/90000
            a=fmtp:96 profile-level-
              id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
            a=framerate:30
          </mediaCapability>
        </mms:mediaCapabilityInfo>
      </TSMediaCapabilityInfo>
    </u:StartMediaSession>
  </s:Body>
</s:Envelope>
```

D.21.19 F18

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
      org:service:MediaManagement:2">
      <MediaSessionID>media2022</MediaSessionID>
      <TCMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <lt;mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
            http://www.upnp.org/schemas/phone/mms-v2.xsd"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <lt;mediaCapability format="SDP">v=0
            o=- 0 0 IN IP4 192.168.2.22
            s=-
            c=IN IP4 192.168.2.22
            t=0 0
            m=video 22022 RTP/AVP 96
            c=IN IP4 192.168.2.22
            b=AS:2500
            a=rtpmap:96 MP4V-ES/90000
            a=fmtp:96 profile-level-
              id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
            a=framerate:30
          </mediaCapability>
        </mms:mediaCapabilityInfo>
      </TCMediaCapabilityInfo>
    </u:StartMediaSessionResponse>
  </s:Body>
</s:Envelope>
```

D.21.20 F19

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20022

SEQ: [sequence number]

Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<MediaSessionInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaSessionInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
  http://www.upnp.org/schemas/phone/mms-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
  &lt;mediaSessionID&gt;media202&lt;/mediaSessionID&gt;
  &lt;mediaSessionStatus&gt;Started&lt;/mediaSessionStatus&gt;
  &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.22
s=-
c=IN IP4 192.168.2.22
t=0 0
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/mms:mediaSessionInfo&gt;</MediaSessionInfo>
</e:property>
</e:propertyset>

```

D.21.21 F20

HTTP/1.1 200 OK

Content-Length: 0

D.21.22 F21

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.2.1:20001

Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
<TelCPName>phone@Bedroom</TelCPName>
<SecretKey>secret201</SecretKey>
<TargetCallID>call201</TargetCallID>
<TCList>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:TCList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
  http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
  &lt;TC&gt;

```

```

    <UDN>uid:97ef6efa-ac89-4ea2-0001-udn202</UDN>
    <mediaSessionID>media202</mediaSessionID>
    </TC>
    </cams:TCList></TCList>
<MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
<cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
  </mediaCapability>
  </cams:mediaCapabilityInfo></MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
</s:Envelope>

```

D.21.23 F22

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
      org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>

```

D.21.24 F23

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uid:c3580410-1dd1-11b2-8000-20001

SEQ: [sequence number]

Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo><?xml version="1.0" encoding="utf-8"?>
    <cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
        http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer">
      <callID>call201</callID>

```

```

<targetNames type="TelCPName">*</targetNames>
<callStatus>Talking</callStatus>
<priority>Normal</priority>
<remoteParty>
  <peer:id>0355550101</peer:id>
</remoteParty>
<TCList>
  <TC>
    <UDN>uid:97ef6efa-ac89-4ea2-0001-udn201</UDN>
    <mediaSessionID>media201</mediaSessionID>
  </TC>
  <TC>
    <UDN>uid:97ef6efa-ac89-4ea2-0001-udn202</UDN>
    <mediaSessionID>media202</mediaSessionID>
  </TC>
</TCList>
<mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22001 RTP/AVP 96
c=IN IP4 192.168.2.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</cans:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.21.25 F24

HTTP/1.1 200 OK
Content-Length: 0

D.22 Accept an incoming Call using multiple TCs (Case 2)

D.22.1 Figure

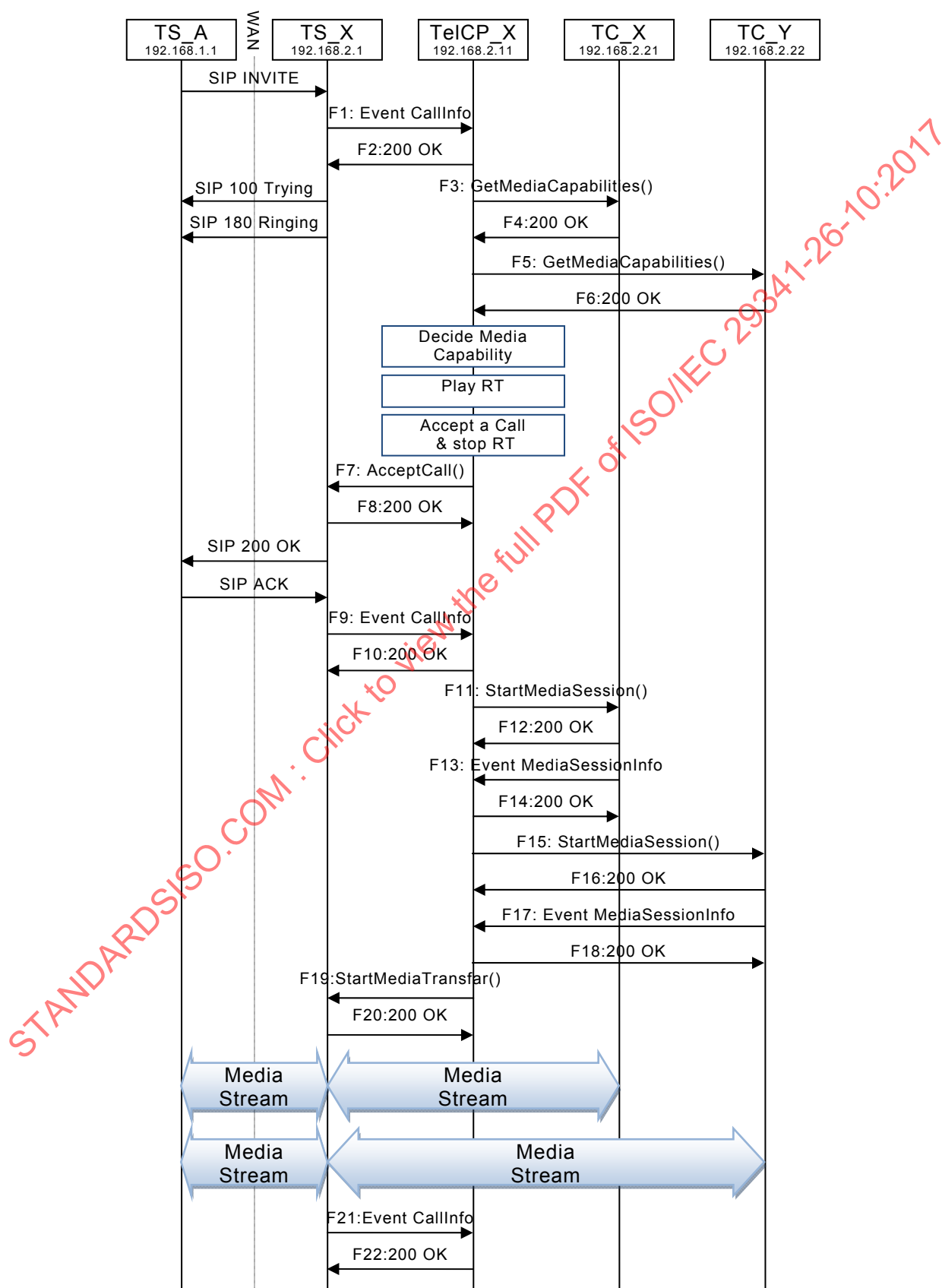


Figure D.22 — Accept an incoming Call using multiple TCs (Case 2)

D.22.2 F1 to F14 are the same as “D.21 Accept an incoming Call using multiple TCs (Case 1)”**D.22.3 F15**

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.2.22:20022
 Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
            http://www.upnp.org/schemas/phone/mms-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:mms">
            <t:mediaCapability format="SDP">v=0
              o=- 0 0 IN IP4 192.168.2.11
              s=-
              c=IN IP4 192.168.2.11
              t=0 0
              m=video 22001 RTP/AVP 96
              c=IN IP4 192.168.2.1
              b=AS:2500
              a=rtpmap:96 MP4V-ES/90000
              a=fmtp:96 profile-level-
                id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
              a=framerate:30
            </t:mediaCapability>
          </t:mms:mediaCapabilityInfo>
        </TSMediaCapabilityInfo>
      </u:StartMediaSession>
    </s:Body>
  </s:Envelope>
```

D.22.4 F16

HTTP/1.1 200 OK
 EXT:
 CONTENT-TYPE: text/xml ; charset="utf-8"
 SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
 Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
      org:service:MediaManagement:2">
      <MediaSessionID>media202</MediaSessionID>
      <TCMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
            http://www.upnp.org/schemas/phone/mms-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:mms">
            <t:mediaCapability format="SDP">v=0
              o=- 0 0 IN IP4 192.168.2.22
              s=-
              c=IN IP4 192.168.2.22
```

ISO/IEC 29341-26-10:2017(E)

```
t=0 0
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
<lt;/mediaCapability>>
<lt;/mms:mediaCapabilityInfo>></TCMediaCapabilityInfo>
</u:StartMediaSessionResponse>
</s:Body>
</s:Envelope>
```

D.22.5 F17

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20022

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <MediaSessionInfo><lt;?xml version="1.0" encoding="utf-8"?>>
      <lt;mms:mediaSessionInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
        http://www.upnp.org/schemas/phone/mms-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:mms="urn:schemas-upnp-org:phone:mms">>
        <lt;mediaSessionID>>media202<lt;/mediaSessionID>>
        <lt;mediaSessionStatus>>Started<lt;/mediaSessionStatus>>
        <lt;mediaCapability format="SDP">>v=0
o=- 0 0 IN IP4 192.168.2.22
s=-
c=IN IP4 192.168.2.22
t=0 0
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
<lt;/mediaCapability>>
<lt;/mms:mediaSessionInfo>></MediaSessionInfo>
  </e:property>
</e:propertyset>
```

D.22.6 F18

HTTP/1.1 200 OK

Content-Length: 0

D.22.7 F19

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.2.1:20001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
<TelCPName>phone@Bedroom</TelCPName>
<SecretKey>secret201</SecretKey>
<TargetCallID>call201</TargetCallID>
<TCList>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:TCList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
    &lt;TC&gt;
      &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn201&lt;/UDN&gt;
      &lt;mediaSessionID&gt;media201&lt;/mediaSessionID&gt;
    &lt;/TC&gt;
    &lt;TC&gt;
      &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn202&lt;/UDN&gt;
      &lt;mediaSessionID&gt;media202&lt;/mediaSessionID&gt;
    &lt;/TC&gt;
  &lt;/cams:TCList&gt;</TCList>
<MediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
    &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/cams:mediaCapabilityInfo&gt;</MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
</s:Envelope>

```

D.22.8 F20

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>

```

ISO/IEC 29341-26-10:2017(E)

D.22.9 F21

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20001

SEQ: [sequence number]

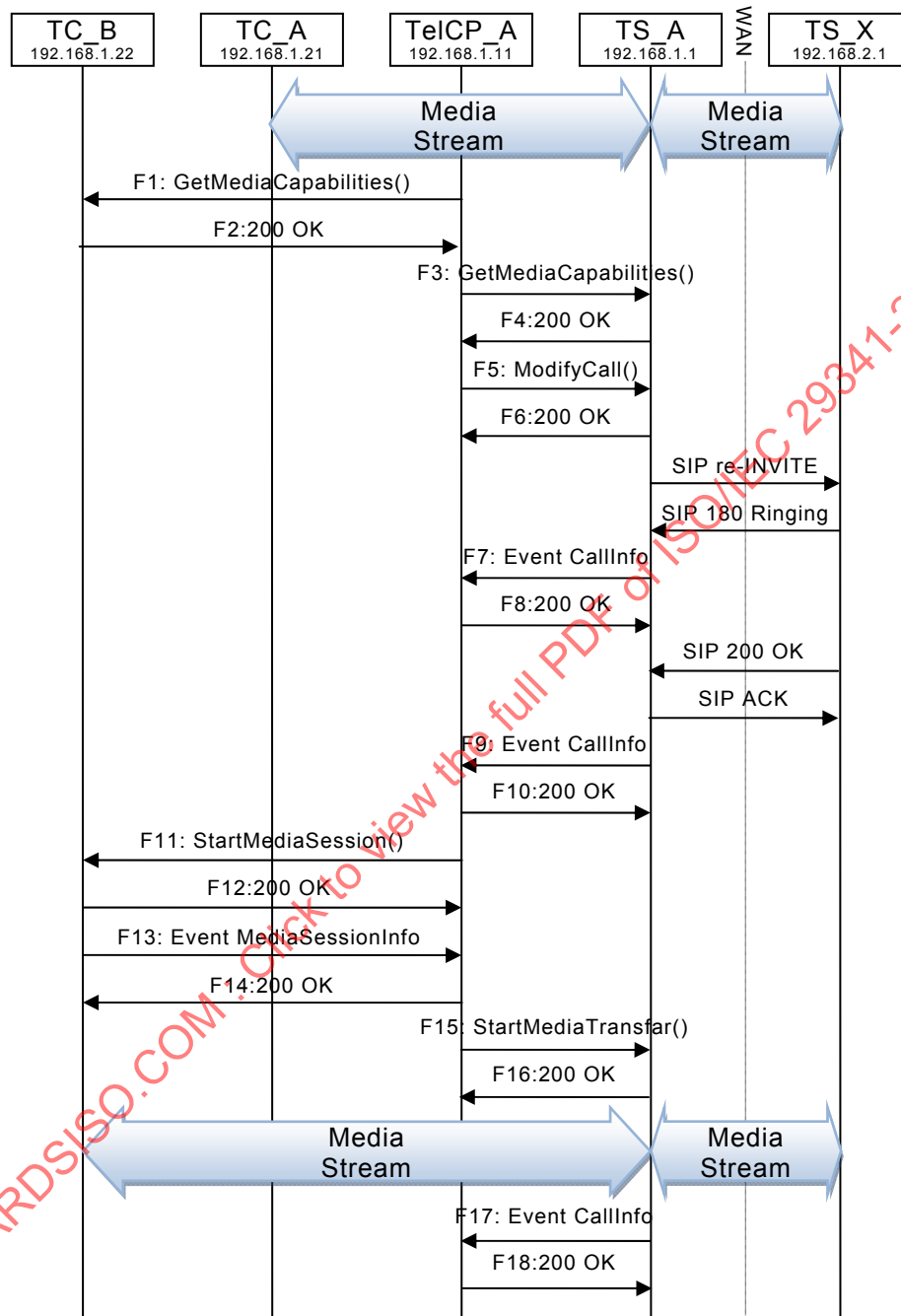
Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
      &lt;cams:callInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
          http://www.upnp.org/schemas/phone/cams-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:cams="urn:schemas-upnp-org:phone:cams"
        xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
        &lt;callID&gt;call1201&lt;/callID&gt;
        &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
        &lt;callStatus&gt;Talking&lt;/callStatus&gt;
        &lt;priority&gt;Normal&lt;/priority&gt;
        &lt;remoteParty&gt;
          &lt;peer:id&gt;0355550101&lt;/peer:id&gt;
        &lt;/remoteParty&gt;
        &lt;TCList&gt;
          &lt;TC&gt;
            &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn201&lt;/UDN&gt;
            &lt;mediaSessionID&gt;media201&lt;/mediaSessionID&gt;
            &lt;/TC&gt;
          &lt;TC&gt;
            &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn202&lt;/UDN&gt;
            &lt;mediaSessionID&gt;media202&lt;/mediaSessionID&gt;
            &lt;/TC&gt;
          &lt;/TCList&gt;
          &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22001 RTP/AVP 96
c=IN IP4 192.168.2.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/cams:callInfo&gt;</CallInfo>
  </e:property>
</e:propertyset>
```

D.22.10 F22

HTTP/1.1 200 OK

Content-Length: 0

D.23 Add TC during a Call (Create a modification request)**D.23.1 Figure****Figure D.23 — Add TC during a Call (Create a modification request)****D.23.2 F1**

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaCapabilities"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.1.22:10022
 Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
```

ISO/IEC 29341-26-10:2017(E)

```
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2" >
      <TSMediaCapabilityInfo></TSMediaCapabilityInfo>
    </u:GetMediaCapabilities>
  </s:Body>
</s:Envelope>
```

D.23.3 F2

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <SupportedMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <?mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <?mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.22
s=-
c=IN IP4 192.168.1.22
t=0 0
m=audio 11022 RTP/AVP 0
c=IN IP4 192.168.1.22
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
        </mediaCapability>
      </mms:mediaCapabilityInfo>
    </SupportedMediaCapabilityInfo>
  </u:GetMediaCapabilitiesResponse>
</s:Body>
</s:Envelope>
```

D.23.4 F3

POST / urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#GetMediaCapabilities"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service: CallManagement:2">
      <TCMediaCapabilityInfo></TCMediaCapabilityInfo>
    </u:GetMediaCapabilities>
  </s:Body>
```

</s:Envelope>

D.23.5 F4

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:CallManagement:2">
      <SupportedMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
      <cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
      <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
      </mediaCapability>
    </cams:mediaCapabilityInfo></SupportedMediaCapabilityInfo>
    </u:GetMediaCapabilitiesResponse>
  </s:Body>
</s:Envelope>
```

D.23.6 F5

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#ModifyCall"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:ModifyCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <SecretKey>secret101</SecretKey>
      <TargetCallID>call101</TargetCallID>
    <MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
    <cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
      <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
```

ISO/IEC 29341-26-10:2017(E)

```
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
<!--mediaCapability-->
<!--cams:mediaCapabilityInfo-->
</MediaCapabilityInfo>
  </u:ModifyCall>
</s:Body>
</s:Envelope>
```

D.23.7 F6

HTTP/1.1 200 OK
Content-Length: [bytes in body]
CONTENT-TYPE: text/xml; charset="utf-8"
SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0
EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:ModifyCallResponse xmlns:m="urn:schemas-upnp-org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.23.8 F7

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10001
SEQ: [sequence number]
Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><!--xml version="1.0" encoding="utf-8"?>
  <!--cams:callInfo
    xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:cams="urn:schemas-upnp-org:phone:cams"
    xmlns:peer="urn:schemas-upnp-org:phone:peer">
      <!--callID--><callID101--></callID-->
      <!--targetNames type="TelCPName">*</targetNames-->
      <!--callStatus-->SendingModifyRequest</callStatus-->
      <!--priority-->Normal</priority-->
      <!--remoteParty-->
        <!--peer:id-->0774940201</peer:id-->
      </remoteParty-->
      <!--TCList-->
      <!--TC-->
```

```

<UDN>uuid:97ef6efa-ac89-4ea2-0001-udn101</UDN>
<mediaSessionID>media101</mediaSessionID>
</TC>
</TCList>
<mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</cams:callInfo>
</CallInfo>
</e:property>
</e:propertyset>

```

D.23.9 F8

HTTP/1.1 200 OK
Content-Length: 0

D.23.10 F9

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10001
SEQ: [sequence number]
Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<callInfo>
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
    <callID>call101</callID>
    <targetNames type="TelCPName">*</targetNames>
    <callStatus>Modified</callStatus>
    <priority>Normal</priority>
    <remoteParty>
      <peer:id>0774940201</peer:id>
    </remoteParty>
    </callInfo>
  </e:property>
</e:propertyset>
<UDN>uuid:97ef6efa-ac89-4ea2-0001-udn101</UDN>
<mediaSessionID>media101</mediaSessionID>
</TC>
</TCList>
<mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1

```

ISO/IEC 29341-26-10:2017(E)

```
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</cams:callInfo>
</CallInfo>
</e:property>
</e:propertyset>
```

D.23.11 F10

HTTP/1.1 200 OK
Content-Length: 0

D.23.12 F11

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.22:10022
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <t:mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
            http://www.upnp.org/schemas/phone/mms-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:mms">
            <t:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</mms:mediaCapabilityInfo>
</TSMediaCapabilityInfo>
      </u:StartMediaSession>
    </s:Body>
  </s:Envelope>
```

D.23.13 F12

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
<MediaSessionID>media102</MediaSessionID>
<TCMediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.22
s=-
c=IN IP4 192.168.1.22
t=0 0
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtptime:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/mms:mediaCapabilityInfo&gt;
</TCMediaCapabilityInfo>
</u:StartMediaSessionResponse>
</s:Body>
</s:Envelope>

```

D.23.14 F13

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10022

SEQ: [sequence number]

Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<MediaSessionInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaSessionInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaSessionID&gt;media102&lt;/mediaSessionID&gt;
&lt;mediaSessionStatus&gt;Started&lt;/mediaSessionStatus&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.22
s=-
c=IN IP4 192.168.1.22
t=0 0
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtptime:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21

```

ISO/IEC 29341-26-10:2017(E)

```
a=framerate:30
<lt;/mediaCapability>
<lt;/mms:mediaSessionInfo>;</MediaSessionInfo>
</e:property>
</e:propertyset>
```

D.23.15 F14

HTTP/1.1 200 OK
Content-Length: 0

D.23.16 F15

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.1:10001
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
<TelCPName>TV@Living</TelCPName>
<SecretKey>secret101</SecretKey>
<TargetCallID>call101</TargetCallID>
<TCList><lt;?xml version="1.0" encoding="utf-8"?>
<lt;cams:TCList
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
<lt;TC>
<lt;UDN>uid:97ef6efa-ac89-4ea2-0001-udn101</UDN>
<lt;mediaSessionID>media101</mediaSessionID>
<lt;/TC>
<lt;TC>
<lt;UDN>uid:97ef6efa-ac89-4ea2-0001-udn102</UDN>
<lt;mediaSessionID>media102</mediaSessionID>
<lt;/TC>
<lt;/cams:TCList></TCList>
<MediaCapabilityInfo><lt;?xml version="1.0" encoding="utf-8"?>
<lt;cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
<lt;mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
<lt;/mediaCapability>
<lt;/cams:mediaCapabilityInfo></MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
```

```
</s:Envelope>
```

D.23.17 F16

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.23.18 F17

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10001

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
  &lt;callID&gt;call101&lt;/callID&gt;
  &lt;targetNames type="TelCPName"&gt;*&lt;/targetNames&gt;
  &lt;callStatus&gt;Talking&lt;/callStatus&gt;
  &lt;priority&gt;Normal&lt;/priority&gt;
  &lt;remoteParty&gt;
    &lt;peer:id&gt;0774940201&lt;/peer:id&gt;
  &lt;/remoteParty&gt;
  &lt;TCList&gt;
    &lt;TC&gt;
      &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn101&lt;/UDN&gt;
      &lt;mediaSessionID&gt;media101&lt;/mediaSessionID&gt;
    &lt;/TC&gt;
    &lt;TC&gt;
      &lt;UDN&gt;uuid:97ef6efa-ac89-4ea2-0001-udn102&lt;/UDN&gt;
      &lt;mediaSessionID&gt;media102&lt;/mediaSessionID&gt;
    &lt;/TC&gt;
  &lt;/TCList&gt;
  &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
```

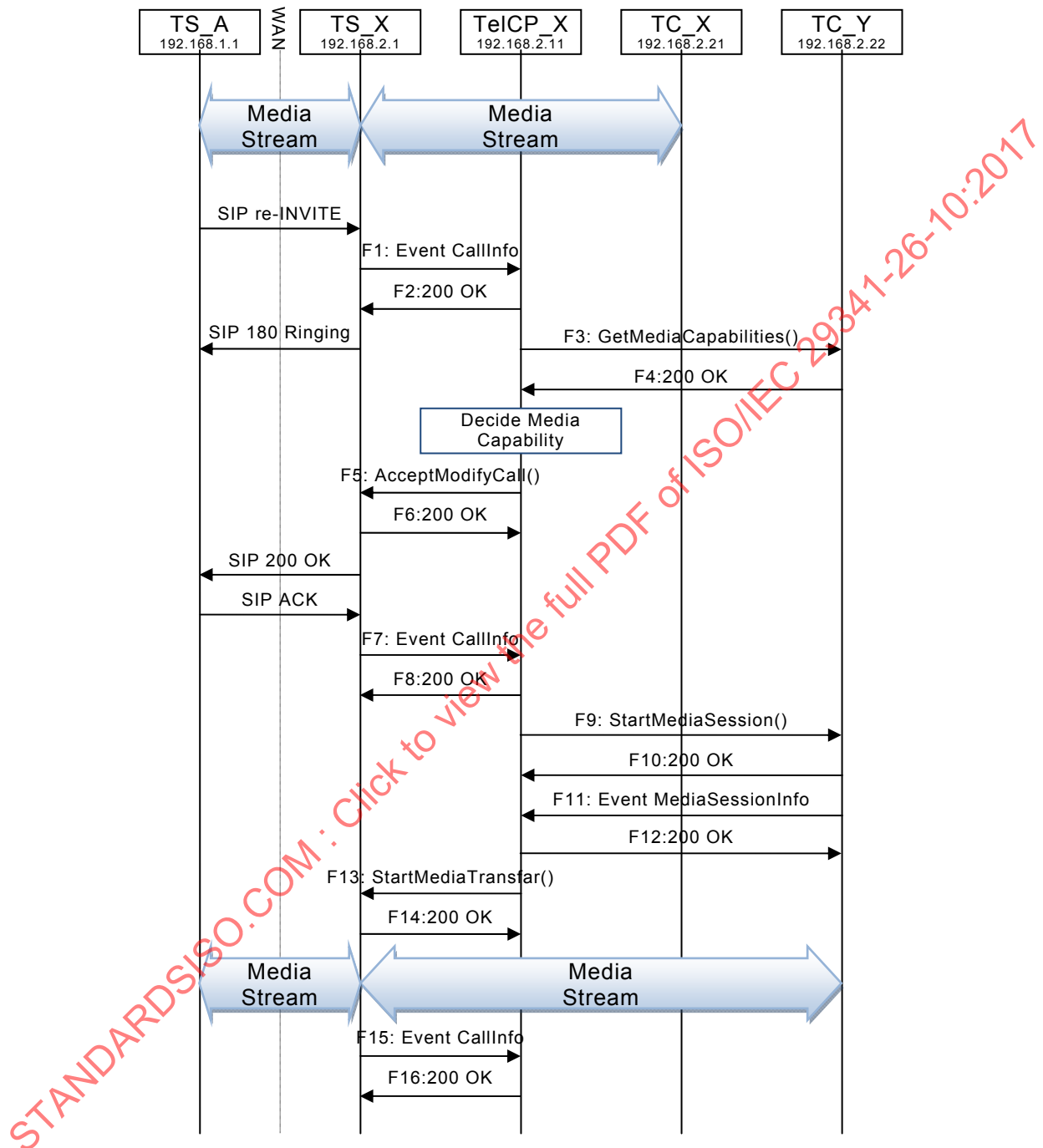
ISO/IEC 29341-26-10:2017(E)

```
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
<lt;/mediaCapability>
<lt;/cams:callInfo></CallInfo>
</e:property>
</e:propertyset>
```

D.23.19 F18

HTTP/1.1 200 OK
Content-Length: 0

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 29341-26-10:2017

D.24 Add TC during a Call (Accept a modification request)**D.24.1 Figure****Figure D.24 — Add TC during a Call (Accept a modification request)****D.24.2 F1**

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
 HOST: 192.168.2.11:20011
 CONTENT-TYPE: text/xml
 Content-Length: [bytes in body]
 NT: upnp:event
 NTS: upnp:propchange
 SID: uuid:c3580410-1dd1-11b2-8000-20001
 SEQ: [sequence number]

ISO/IEC 29341-26-10:2017(E)

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <callID><callID201>/callID>
  <targetNames type="TelCPName">*</targetNames>
  <callStatus>ReceivingModifyRequest</callStatus>
  <priority>Normal</priority>
  <remoteParty>
    <peer:id>0355550101</peer:id>
  </remoteParty>
  <TCList>
    <TC>
      <UDN>uuid:97ef6efa-ac89-4ea2-0001-udn201</UDN>
      <mediaSessionID>media201</mediaSessionID>
    </TC>
  </TCList>
</callInfo format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22001 RTP/AVP 96
c=IN IP4 192.168.2.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</callInfo></CallInfo>
</e:property>
</e:propertyset>
```

D.24.3 F2

HTTP/1.1 200 OK
Content-Length: 0

D.24.4 F3

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#GetMediaCapabilities"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.22:20022
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <u:GetMediaCapabilities xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2" >
    <TSMediaCapabilityInfo></TSMediaCapabilityInfo>
  </u:GetMediaCapabilities>
</s:Envelope>
```

D.24.5 F4

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:GetMediaCapabilitiesResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
      <SupportedMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?&gt;
        <t;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms"&gt;
          <t;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.22
s=-
c=IN IP4 192.168.2.22
t=0 0
m=audio 21022 RTP/AVP 0
c=IN IP4 192.168.2.22
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
          &t;/mediaCapability&gt;
        &t;/mms:mediaCapabilityInfo&gt;</SupportedMediaCapabilityInfo>
      </u:GetMediaCapabilitiesResponse>
    </s:Body>
  </s:Envelope>
```

D.24.6 F5

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#AcceptModifyCall"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.2.1:20001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:AcceptModifyCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>phone@Bedroom</TelCPName>
      <SecretKey>secret201</SecretKey>
      <TargetCallID>call1201</TargetCallID>
      <MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?&gt;
        <t;cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
          <t;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
```

ISO/IEC 29341-26-10:2017(E)

```
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
<lt;/mediaCapability>>
<lt;/cams:mediaCapabilityInfo>>
</MediaCapabilityInfo>
  </u:AcceptModifyCall>
</s:Body>
</s:Envelope>
```

D.24.7 F6

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:AcceptModifyCallResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>
```

D.24.8 F7

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20001

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><lt;?xml version="1.0" encoding="utf-8"?>>
<lt;cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">>
  <lt;callID>><callID>>
  <lt;targetNames type="TelCPName">>*<lt;/targetNames>>
  <lt;callStatus>>Modified<lt;/callStatus>>
  <lt;priority>>Normal<lt;/priority>>
  <lt;remoteParty>>
    <lt;peer:id>>0355550101<lt;/peer:id>>
  <lt;/remoteParty>>
  <lt;TCList>>
    <lt;TC>>
    <lt;UDN>>uuid:97ef6efa-ac89-4ea2-0001-udn201<lt;/UDN>>
```

```

    <mediaSessionID><media201</mediaSessionID>
    </TC>
    </TCList>
    <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22001 RTP/AVP 96
c=IN IP4 192.168.2.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
    </mediaCapability>
    </cams:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.24.9 F8

HTTP/1.1 200 OK
Content-Length: 0

D.24.10 F9

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StartMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.22:20022
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaSession xmlns:u="urn:schemas-upnp-org:service:MediaManagement:2">
      <TSMediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
        <mms:mediaCapabilityInfo
          xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
            http://www.upnp.org/schemas/phone/mms-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:mms">
          <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=video 22001 RTP/AVP 96
c=IN IP4 192.168.2.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
          </mediaCapability>
        </mms:mediaCapabilityInfo>
      </TSMediaCapabilityInfo>
    </u:StartMediaSession>
  </s:Body>
</s:Envelope>

```

ISO/IEC 29341-26-10:2017(E)

D.24.11 F10

HTTP/1.1 200 OK

EXT:

CONTENT-TYPE: text/xml ; charset="utf-8"

SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2">
<MediaSessionID>media202</MediaSessionID>
<TCMediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.22
s=-
c=IN IP4 192.168.2.22
t=0 0
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
&lt;/mediaCapability&gt;
&lt;/mms:mediaCapabilityInfo&gt;</TCMediaCapabilityInfo>
</u:StartMediaSessionResponse>
</s:Body>
</s:Envelope>
```

D.24.12 F11

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20022

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<MediaSessionInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;mms:mediaSessionInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
http://www.upnp.org/schemas/phone/mms-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
&lt;mediaSessionID&gt;media202&lt;/mediaSessionID&gt;
&lt;mediaSessionStatus&gt;Started&lt;/mediaSessionStatus&gt;
&lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.22
s=-
c=IN IP4 192.168.2.22
t=0 0
m=video 22022 RTP/AVP 96
```

```

c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</mms:mediaSessionInfo></MediaSessionInfo>
</e:property>
</e:propertyset>

```

D.24.13 F12

HTTP/1.1 200 OK
Content-Length: 0

D.24.14 F13

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.1:20001
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
<TelCPName>phone@Bedroom</TelCPName>
<SecretKey>secret201</SecretKey>
<TargetCallID>call201</TargetCallID>
<TCList><?xml version="1.0" encoding="utf-8"?>
<tc:cams:TCList
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
<tc:TC>
<tc:UDN>97ef6efa-ac89-4ea2-0001-udn201</tc:UDN>
<tc:mediaSessionID>media201</tc:mediaSessionID>
</tc:TC>
<tc:TC>
<tc:UDN>97ef6efa-ac89-4ea2-0001-udn202</tc:UDN>
<tc:mediaSessionID>media202</tc:mediaSessionID>
</tc:TC>
</tc:cams:TCList></TCList>
<MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
<tc:cams:mediaCapabilityInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams">
<tc:mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22022RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21

```

ISO/IEC 29341-26-10:2017(E)

```
a=framerate:30
<lt;/mediaCapability>
<lt;/cams:mediaCapabilityInfo></MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
</s:Envelope>
```

D.24.15 F14

HTTP/1.1 200 OK
Content-Length: [bytes in body]
CONTENT-TYPE: text/xml; charset="utf-8"
SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0
EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<s:Body>
<m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
</s:Body>
</s:Envelope>
```

D.24.16 F15

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.2.11:20011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-20001
SEQ: [sequence number]
Connection: Keep-Alive

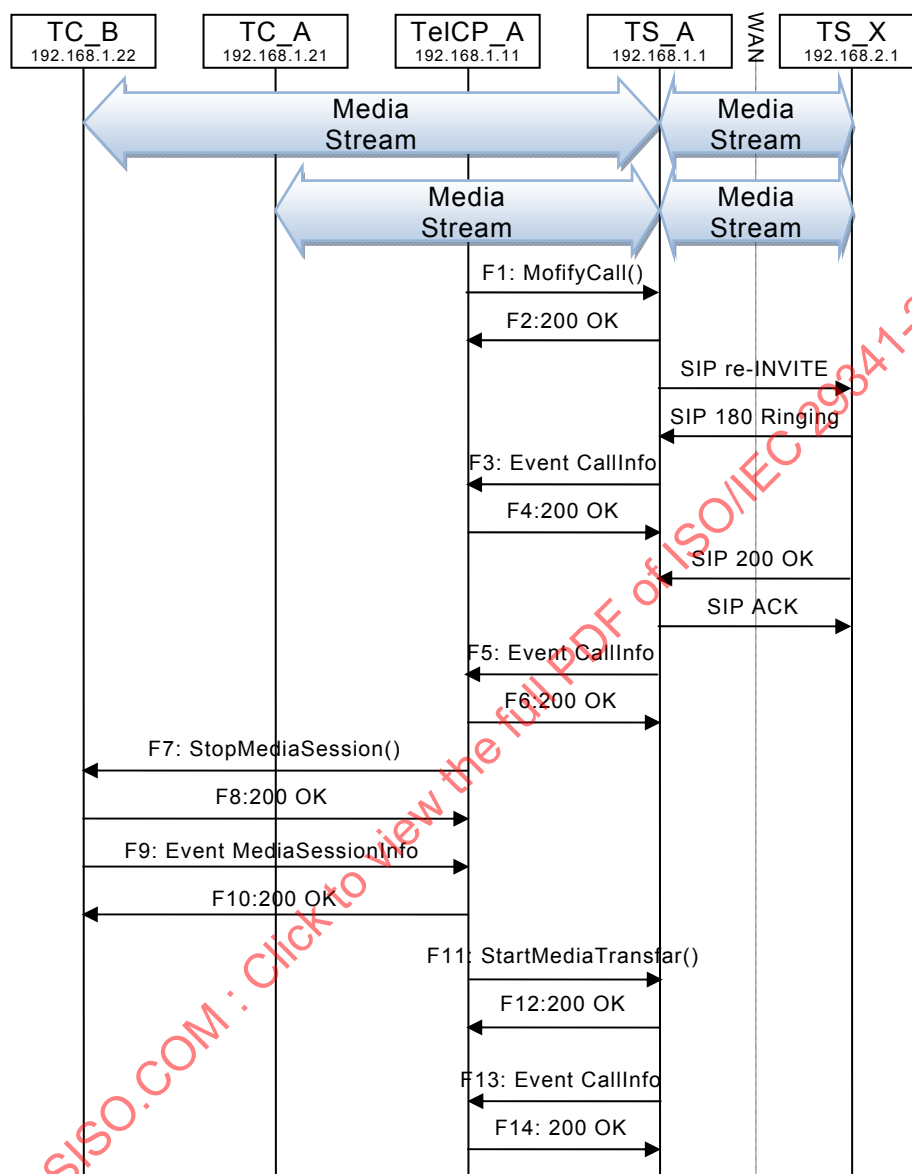
```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><lt;?xml version="1.0" encoding="utf-8"?>
<lt;cams:callInfo
xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cams="urn:schemas-upnp-org:phone:cams"
xmlns:peer="urn:schemas-upnp-org:phone:peer">
<lt;callID><lt;callID201></callID>
<lt;targetNames type="TelCPName">*</targetNames>
<lt;callStatus><lt;Talking></callStatus>
<lt;priority><lt;Normal></priority>
<lt;remoteParty>
<lt;peer:id><lt;0355550101></peer:id>
<lt;/remoteParty>
<lt;TCList>
<lt;TC>
<lt;UDN><lt;uuid:97ef6efa-ac89-4ea2-0001-udn201></UDN>
<lt;mediaSessionID><lt;media201></mediaSessionID>
<lt;/TC>
<lt;TC>
<lt;UDN><lt;uuid:97ef6efa-ac89-4ea2-0001-udn202></UDN>
<lt;mediaSessionID><lt;media202></mediaSessionID>
<lt;/TC>
<lt;/TCList>
<lt;mediaCapability format="SDP"><lt;v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
```

```
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 22001 RTP/AVP 96
c=IN IP4 192.168.2.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30
</mediaCapability>
</cams:callInfo></CallInfo>
</e:property>
</e:propertyset>
```

D.24.17 F16

HTTP/1.1 200 OK
Content-Length: 0

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 29341-26-10:2017

D.25 Remove TC during a Call (Create a modification request)**D.25.1 Figure****Figure D.25 — Remove TC during a Call (Create a modification request)****D.25.2 F1**

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#ModifyCall"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.1.1:10001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:ModifyCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <SecretKey>secret101</SecretKey>
      <TargetCallID>call101</TargetCallID>
    </u:ModifyCall>
  </s:Body>
</s:Envelope>
```

```

<MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
<cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
</mediaCapability >
</cams:mediaCapabilityInfo></MediaCapabilityInfo>
  </u:ModifyCall>
</s:Body>
</s:Envelope>

```

D.25.3 F2

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:ModifyCallResponse xmlns:m="urn:schemas-upnp-org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>

```

D.25.4 F3

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10001

SEQ: [sequence number]

Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <callID><callID101></callID>
  <targetNames type="TelCPName">*</targetNames>
  <callStatus><SendingModifyRequest></callStatus>
  <priority>Normal</priority>
  <remoteParty>
    <peer:id>0774940201</peer:id>
  </remoteParty>
  <TCList>
    <TC>

```

ISO/IEC 29341-26-10:2017(E)

```
<UDN>uuid:97ef6efa-ac89-4ea2-0001-udn101</UDN>
<mediaSessionID>media101</mediaSessionID>
</TC>
<TC>
<UDN>uuid:97ef6efa-ac89-4ea2-0001-udn102</UDN>
<mediaSessionID>media102</mediaSessionID>
</TC>
</TCList>
<mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20
m=video 12001 RTP/AVP 96
c=IN IP4 192.168.1.1
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:30</mediaCapability>
</cams:callInfo></CallInfo>
</e:property>
</e:propertyset>
```

D.25.5 F4

HTTP/1.1 200 OK
Content-Length: 0

D.25.6 F5

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10001
SEQ: [sequence number]
Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<callInfo>
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
    <callID>call101</callID>
    <targetNames type="TelCPName">*</targetNames>
    <callStatus>Modified</callStatus>
    <priority>Normal</priority>
    <remoteParty>
      <peer:id>0774940201</peer:id>
    </remoteParty>
    </callInfo>
    </TC>
    <UDN>uuid:97ef6efa-ac89-4ea2-0001-udn101</UDN>
    <mediaSessionID>media101</mediaSessionID>
    </TC>
    </TCList>
```

```

    <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1
a=rtpmap:0 PCMU/8000
a=ptime:20</mediaCapability>
</cams:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.25.7 F6

HTTP/1.1 200 OK
Content-Length: 0

D.25.8 F7

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StopMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.22:10022
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopMediaSession xmlns:u="urn:schemas-upnp-org:service: MediaManagement:2">
      <TargetMediaSessionID>media102</TargetMediaSessionID>
    </u:StopMediaSession>
  </s:Body>
</s:Envelope>

```

D.25.9 F8

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2" />
  </s:Body>
</s:Envelope>

```

D.25.10 F9

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1
HOST: 192.168.1.11:10011
CONTENT-TYPE: text/xml
Content-Length: [bytes in body]
NT: upnp:event
NTS: upnp:propchange
SID: uuid:c3580410-1dd1-11b2-8000-10022
SEQ: [sequence number]
Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <MediaSessionInfo><?xml version="1.0" encoding="utf-8"?>

```

ISO/IEC 29341-26-10:2017(E)

```
<lt;mms:mediaSessionInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
  http://www.upnp.org/schemas/phone/mms-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:mms="urn:schemas-upnp-org:phone:mms">
  <lt;mediaSessionID><media102</mediaSessionID>
  <lt;mediaSessionStatus><Stopped</mediaSessionStatus>
  <lt;mediaCapability format="SDP"><v=0
o=- 0 0 IN IP4 192.168.1.22
s=-
c=IN IP4 192.168.1.22
t=0 0
m=video 12022 RTP/AVP 96
c=IN IP4 192.168.1.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:3
</mediaCapability>
</mms:mediaSessionInfo></MediaSessionInfo>
</e:property>
</e:propertyset>
```

D.25.11 F10

HTTP/1.1 200 OK
Content-Length: 0

D.25.12 F11

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.1.1:10001
Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8">
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>TV@Living</TelCPName>
      <SecretKey>secret101</SecretKey>
      <TargetCallID>call101</TargetCallID>
      <TCList><?xml version="1.0" encoding="utf-8"><
        <lt;cams:TCList
          xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
          http://www.upnp.org/schemas/phone/cams-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:cams">
            <lt;TC>
              <lt;UDN><uuid:97ef6efa-ac89-4ea2-0001-udn101</UDN>
              <lt;mediaSessionID><media101</mediaSessionID>
              <lt;TC>
            </cams:TCList></TCList>
          <MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"><
            <lt;cams:mediaCapabilityInfo
              xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
              http://www.upnp.org/schemas/phone/cams-v2.xsd"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xmlns:cams="urn:schemas-upnp-org:phone:cams">
                <lt;mediaCapability format="SDP"><v=0
o=- 0 0 IN IP4 192.168.1.11
s=-
c=IN IP4 192.168.1.11
t=0 0
m=audio 11021 RTP/AVP 0
```

```

c=IN IP4 192.168.1.21
a=rtpmap:0 PCMU/8000
a=ptime:20
<!--mediaCapability-->
<!--cams:mediaCapabilityInfo--></MediaCapabilityInfo>
  </u:StartMediaTransfer>
</s:Body>
</s:Envelope>

```

D.25.13 F12

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
</s:Envelope>

```

D.25.14 F13

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.1.11:10011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-10001

SEQ: [sequence number]

Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><!--?xml version="1.0" encoding="utf-8"?-->
  <!--cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
  http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
    <!--callID--><!--callID-->
    <!--targetNames type="TelCPName">*</targetNames-->
    <!--callStatus--><!--callStatus-->
    <!--priority-->Normal</priority-->
    <!--remoteParty-->
    <!--peer:id--><!--peer:id-->
  </remoteParty-->
  <!--TCList-->
  <!--TC-->
    <!--UDN-->uuid:97ef6efa-ac89-4ea2-0001-udn101</UDN-->
    <!--mediaSessionID--><!--mediaSessionID-->
    </TC-->
  </TCList-->
  <!--mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.1.1
s=-
c=IN IP4 192.168.1.1
t=0 0
m=audio 11001 RTP/AVP 0
c=IN IP4 192.168.1.1

```

ISO/IEC 29341-26-10:2017(E)

```
a=rtpmap:0 PCMU/8000
a=ptime:20
</mediaCapability>
</cans:callInfo>
</CallInfo>
</e:property>
</e:propertyset>
```

D.25.15 F14

HTTP/1.1 200 OK
Content-Length: 0

D.26 Remove TC during a Call (Accept a modification request)

D.26.1 Figure

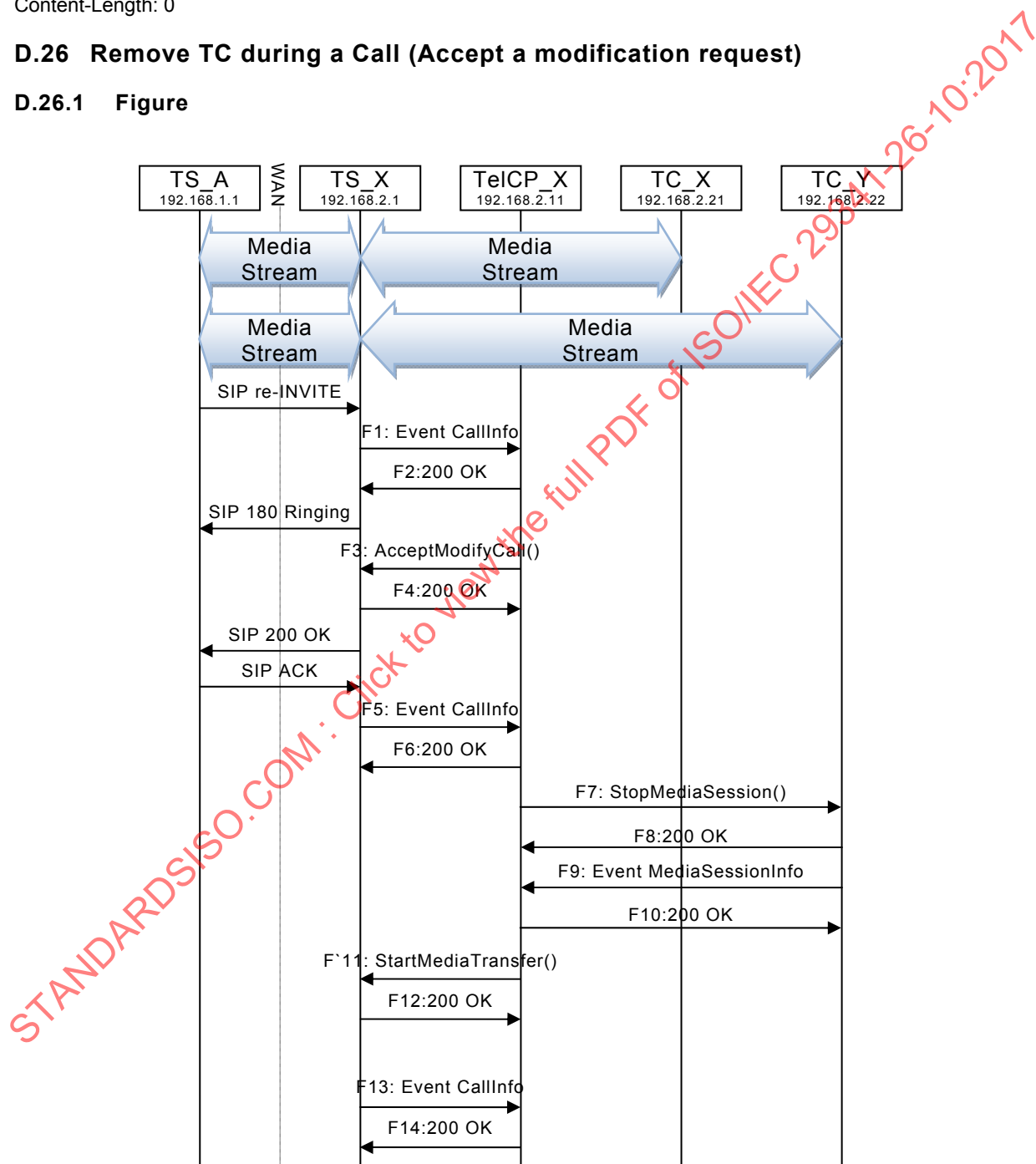


Figure D.26 — Remove TC during a Call (Accept a modification request)

D.26.2 F1

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.2.11:20011
 CONTENT-TYPE: text/xml
 Content-Length: [bytes in body]
 NT: upnp:event
 NTS: upnp:propchange
 SID: uuid:c3580410-1dd1-11b2-8000-20001
 SEQ: [sequence number]
 Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo><?xml version="1.0" encoding="utf-8"?>
      <callInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
          http://www.upnp.org/schemas/phone/cams-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:cams="urn:schemas-upnp-org:phone:cams"
        xmlns:peer="urn:schemas-upnp-org:phone:peer">
          <callID><callID201>/callID>
          <targetNames type="TelCPName">*</targetNames>
          <callStatus>ReceivingModifyRequest</callStatus>
          <priority>Normal</priority>
          <remoteParty>
            <peer:id>035550101</peer:id>
          </remoteParty>
          <TCList>
            <TC>
              <UDN>uuid:97ef6efa-ac89-4ea2-0001-udn201</UDN>
              <mediaSessionID>media201</mediaSessionID>
            </TC>
            <TC>
              <UDN>uuid:97ef6efa-ac89-4ea2-0001-udn202</UDN>
              <mediaSessionID>media202</mediaSessionID>
            </TC>
          </TCList>
          <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
          </mediaCapability>
        </callInfo>
      </CallInfo>
    </e:property>
  </e:propertyset>
```

D.26.3 F2

HTTP/1.1 200 OK
 Content-Length: 0

D.26.4 F3

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1
 SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#AcceptModifyCall"
 CONTENT-TYPE: text/xml ; charset="utf-8"
 HOST: 192.168.2.1:20001
 Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:AcceptModifyCall xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
```

ISO/IEC 29341-26-10:2017(E)

```
<TelCPName>phone@Bedroom</TelCPName>
<SecretKey>secret201</SecretKey>
<TargetCallID>call201</TargetCallID>
<MediaCapabilityInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
&lt;cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
  http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
  &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
&lt;/mediaCapability&gt;
&lt;/cams:mediaCapabilityInfo&gt;
&lt;/MediaCapabilityInfo>
  </u:AcceptModifyCall>
  </s:Body>
&lt;/s:Envelope>
```

D.26.5 F4

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <s:Body>
    <m:AcceptModifyCallResponse xmlns:m="urn:schemas-upnp-
org:service:CallManagement:2" />
  </s:Body>
&lt;/s:Envelope>
```

D.26.6 F5

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20001

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <CallInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
    &lt;cams:callInfo
      xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
      http://www.upnp.org/schemas/phone/cams-v2.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:cams="urn:schemas-upnp-org:phone:cams"
      xmlns:peer="urn:schemas-upnp-org:phone:peer"&gt;
      &lt;callID&gt;call201&lt;/callID&gt;
      &lt;targetNames type="TelCPName"&gt;phone@Bedroom&lt;/targetNames&gt;
      &lt;callStatus&gt;Modified&lt;/callStatus&gt;
      &lt;priority&gt;Normal&lt;/priority&gt;
```

```

<remoteParty>
  <peer:id>035550101</peer:id>
</remoteParty>
<TCList>
  <TC>
    <UDN>uid:97ef6efa-ac89-4ea2-0001-udn201</UDN>
    <mediaSessionID>media201</mediaSessionID>
  </TC>
  <TC>
    <UDN>uid:97ef6efa-ac89-4ea2-0001-udn202</UDN>
    <mediaSessionID>media202</mediaSessionID>
  </TC>
</TCList>
<mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.1
s=-
c=IN IP4 192.168.2.1
t=0 0
m=audio 21001 RTP/AVP 0
c=IN IP4 192.168.2.1
a=rtpmap:0 PCMU/8000
a=ptime:20
</mediaCapability>
</c:callInfo></CallInfo>
</e:property>
</e:propertyset>

```

D.26.7 F6

HTTP/1.1 200 OK
Content-Length: 0

D.26.8 F7

POST /_urn:upnp-org:serviceId:MediaManagement_control HTTP/1.1
SOAPACTION: "urn:schemas-upnp-org:service:MediaManagement:2#StopMediaSession"
CONTENT-TYPE: text/xml ; charset="utf-8"
HOST: 192.168.2.22:20022
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopMediaSession xmlns:u="urn:schemas-upnp-org:service: MediaManagement:2">
      <TargetMediaSessionID>media202</TargetMediaSessionID>
    </u:StopMediaSession>
  </s:Body>
</s:Envelope>

```

D.26.9 F8

HTTP/1.1 200 OK
EXT:
CONTENT-TYPE: text/xml ; charset="utf-8"
SERVER: Windows NT/5.0, UPnP/1.0, Intel CLR SDK/1.0
Content-Length: [bytes in body]

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StopMediaSessionResponse xmlns:u="urn:schemas-upnp-
org:service:MediaManagement:2" />
  </s:Body>
</s:Envelope>

```

ISO/IEC 29341-26-10:2017(E)

D.26.10 F9

NOTIFY /_urn:upnp-org:serviceId:MediaManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20022

SEQ: [sequence number]

Connection: Keep-Alive

```
<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
  <e:property>
    <MediaSessionInfo>&lt;?xml version="1.0" encoding="utf-8"?&gt;
      &lt;mms:mediaSessionInfo
        xsi:schemaLocation="urn:schemas-upnp-org:phone:mms
          http://www.upnp.org/schemas/phone/mms-v2.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:mms="urn:schemas-upnp-org:phone:mms"&gt;
          &lt;mediaSessionID&gt;media202&lt;/mediaSessionID&gt;
          &lt;mediaSessionStatus&gt;Stopped&lt;/mediaSessionStatus&gt;
          &lt;mediaCapability format="SDP"&gt;v=0
o=- 0 0 IN IP4 192.168.2.22
s=-
c=IN IP4 192.168.2.22
t=0 0
m=video 22022 RTP/AVP 96
c=IN IP4 192.168.2.22
b=AS:2500
a=rtpmap:96 MP4V-ES/90000
a=fmtp:96 profile-level-
id=4;config=000001b004000001b50900000100000001200086c40fa28a021e0a21
a=framerate:3
&lt;/mediaCapability&gt;
&lt;/mms:mediaSessionInfo&gt;</MediaSessionInfo>
  </e:property>
</e:propertyset>
```

D.26.11 F10

HTTP/1.1 200 OK

Content-Length: 0

D.26.12 F11

POST /_urn:upnp-org:serviceId:CallManagement_control HTTP/1.1

SOAPACTION: "urn:schemas-upnp-org:service:CallManagement:2#StartMediaTransfer"

CONTENT-TYPE: text/xml ; charset="utf-8"

HOST: 192.168.2.1:20001

Content-Length: [bytes in body]

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <u:StartMediaTransfer xmlns:u="urn:schemas-upnp-org:service:CallManagement:2">
      <TelCPName>phone@Bedroom</TelCPName>
      <SecretKey>secret201</SecretKey>
      <TargetCallID>call201</TargetCallID>
      <TCList>&lt;?xml version="1.0" encoding="utf-8"?&gt;
        &lt;cams:TCList
          xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
            http://www.upnp.org/schemas/phone/cams-v2.xsd"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:cams="urn:schemas-upnp-org:phone:cams"&gt;
            &lt;TC&gt;
```

```

<UDN>uid:97ef6efa-ac89-4ea2-0001-udn201</UDN>
<mediaSessionID>media201</mediaSessionID>
</TC>
</cams:TCList></TCList>
<MediaCapabilityInfo><?xml version="1.0" encoding="utf-8"?>
<cams:mediaCapabilityInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams">
  <mediaCapability format="SDP">v=0
o=- 0 0 IN IP4 192.168.2.11
s=-
c=IN IP4 192.168.2.11
t=0 0
m=audio 21021 RTP/AVP 0
c=IN IP4 192.168.2.21
a=rtpmap:0 PCMU/8000
a=ptime:20
</mediaCapability>
</cams:mediaCapabilityInfo></MediaCapabilityInfo>
</u:StartMediaTransfer>
</s:Body>
</s:Envelope>

```

D.26.13 F12

HTTP/1.1 200 OK

Content-Length: [bytes in body]

CONTENT-TYPE: text/xml; charset="utf-8"

SERVER: TS-HTTP/1.1 UPnP/1.0 UPnP-Device-Host/1.0

EXT:

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<s:Body>
<m:StartMediaTransferResponse xmlns:m="urn:schemas-upnp-
  org:service:CallManagement:2" />
</s:Body>
</s:Envelope>

```

D.26.14 F13

NOTIFY /_urn:upnp-org:serviceId:CallManagement HTTP/1.1

HOST: 192.168.2.11:20011

CONTENT-TYPE: text/xml

Content-Length: [bytes in body]

NT: upnp:event

NTS: upnp:propchange

SID: uuid:c3580410-1dd1-11b2-8000-20001

SEQ: [sequence number]

Connection: Keep-Alive

```

<?xml version="1.0" encoding="utf-8"?>
<e:propertyset xmlns:e="urn:schemas-upnp-org:event-1-0">
<e:property>
<CallInfo><?xml version="1.0" encoding="utf-8"?>
<cams:callInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:cams
    http://www.upnp.org/schemas/phone/cams-v2.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:cams="urn:schemas-upnp-org:phone:cams"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <callID>call201</callID>
  <targetNames type="TelCPName">phone@Bedroom</targetNames>
  <callStatus>Talking</callStatus>
  <priority>Normal</priority>

```