

Presence:1 Service

For UPnP Version 1.0

Status: Standardized DCP (SDCP)

Date: December 10, 2012

Document Version: 1.0

Service Template Version: 2.00

This Standardized DCP has been adopted as a Standardized DCP by the Steering Committee of the UPnP Forum, pursuant to Section 2.1(c)(ii) of the UPnP Forum Membership Agreement. UPnP Forum Members have rights and licenses defined by Section 3 of the UPnP Forum Membership Agreement to use and reproduce the Standardized DCP in UPnP Compliant Devices. All such use is subject to all of the provisions of the UPnP Forum Membership Agreement.

THE UPNP FORUM TAKES NO POSITION AS TO WHETHER ANY INTELLECTUAL PROPERTY RIGHTS EXIST IN THE STANDARDIZED DCPS. THE STANDARDIZED DCPS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS". THE UPNP FORUM MAKES NO WARRANTIES, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE STANDARDIZED DCPS, INCLUDING BUT NOT LIMITED TO ALL IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OF REASONABLE CARE OR WORKMANLIKE EFFORT, OR RESULTS OR OF LACK OF NEGLIGENCE.

© 2012 UPnP Forum. All Rights Reserved.

Authors ^a	Company
Mahfuzur Rahman (Chair)	Samsung Electronics
Mayuresh Patil (Editor)	Samsung Electronics
Jooyeol Lee	Samsung Electronics
Jeyoung Maeng	Samsung Electronics
Massimo Messori	Telecom Italia
Alessandro De Vincentis	Telecom Italia
Yoshiki Nishikawa	NTT
Yu Zhu	Huawei
JangWoong Park	LGE
Fabrice Fontaine	France Telecom
^a Note: The UPnP Forum in no way guarantees the accuracy or completeness of this author list and in no way implies any rights for or support from those members listed. This list is not the specifications' contributor list that is kept on the UPnP Forum's website.	

CONTENTS

1	Scope.....	3
2	Normative references	4
3	Terms, definitions, symbols and abbreviated terms.....	5
4	Notations and conventions	6
4.1	Text conventions	6
4.1.1	Data Types.....	6
4.2	Vendor-defined Extensions.....	7
5	Service Modeling Definitions.....	7
5.1	Service Type.....	7
5.2	<u>Presence</u> Service Architecture.....	7
5.3	State Variables.....	7
5.3.1	State Variable Overview.....	8
5.3.2	<u>UserPresenceInfo</u>	8
5.3.3	<u>PresenceOfContactsUpdate</u>	9
5.3.4	<u>A_ARG_TYPE UserPresenceInfo</u>	11
5.3.5	<u>A_ARG_TYPE PresenceOfContactsList</u>	13
5.3.6	<u>A_ARG_TYPE ContactId</u>	14
5.3.7	<u>A_ARG_TYPE RegistrationResult</u>	15
5.3.8	<u>A_ARG_TYPE Expires</u>	15
5.3.9	<u>Watcher</u>	15
5.3.10	<u>A_ARG_TYPE WatcherList</u>	16
5.4	Eventing and Moderation.....	17
5.4.1	Eventing of <u>UserPresenceInfo</u>	17
5.4.2	Eventing of <u>PresenceOfContactsUpdate</u>	17
5.4.3	Eventing of <u>Watcher</u>	17
5.5	Actions	18
5.5.1	<u>GetPresence()</u>	18
5.5.2	<u>UpdatePresence()</u>	19
5.5.3	<u>GetContactPresence()</u>	20
5.5.4	<u>GetPresenceOfContactsUpdate()</u>	21
5.5.5	<u>RegisterForContactPresence()</u>	22
5.5.6	<u>AuthorizePresenceReactive()</u>	23
5.5.7	<u>AuthorizePresenceProactive()</u>	24
5.5.8	Error Code Summary	25
5.6	Service Behavioral Model	25
6	XML Service Description.....	25
Annex A (normative)	XML complex type <i>peerType</i>	29
Annex B (normative)	XML Schema	33
Annex C (informative)	Theory of Operation.....	35
Annex D (informative)	Bibliography	37

When the features provided by the Presence service are associated to the PhoneManagement features, then an integrated management of the address book and the presence of contacts in the address book can be delivered.

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

[4] – IETF RFC 2119, Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, 1997. Available at: <http://www.faqs.org/rfcs/rfc2119.html>.

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

[6] – 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>.

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

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

[11] – IETF RFC 3863, Presence Information Data Format (PIDF), H. Sugano, S. Fujimoto, Fujitsu, G. Klyne, Nine by Nine, A. Bateman, VisionTech, W. Carr, Intel, J. Peterson NeuStar, August 2004. Available at: <http://www.ietf.org/rfc/rfc3863.txt>.

3.4.2

ID

Identifier

3.4.3

TC

Telephony Client

3.4.4

TelCP

Telephony Control Point

3.4.5

TS

Telephony Server

3.4.6

VoIP

Voice over IP

3.4.7

WAN

Wide Area Network

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 by [1] are printed using the arch character style.

4.1.1 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 [1]. The XML Schema namespace is used to define property data types [9].

For Boolean data types from [1], 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.

4.2 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:Presence:1

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

5.2 Presence Service Architecture

This service provides the features for a TelCP to manage the presence information that a UPnP device, either a TS or a TC acting as UPnP server for presence management, exposes in the UPnP network, in order to:

- Retrieve and update the presence status managed by the Presence service, and representing the presence information for a user.
- Get notifications of presence updates.
- Retrieve the presence information of the remote contacts managed by the Presence service.
- Get notifications of presence updates of remote contacts.

The architecture for the Presence service is shown in Figure 1.

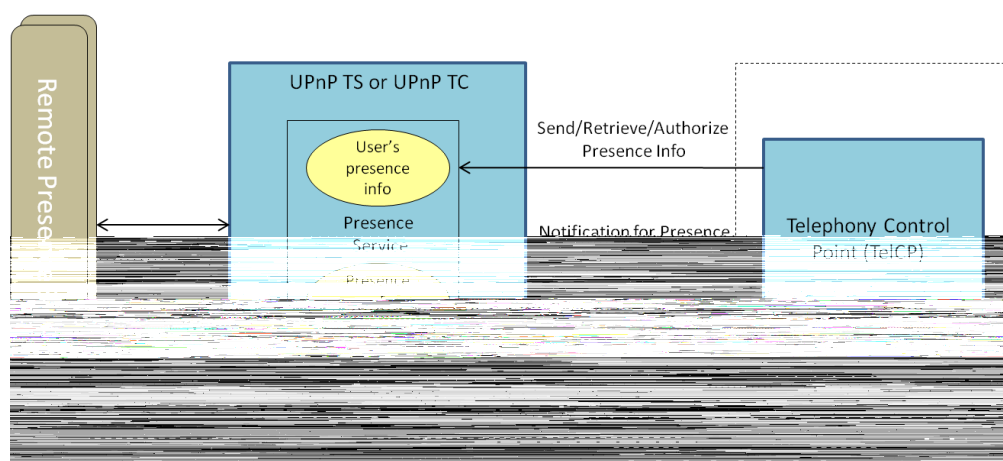


Figure 1 — Presence Service Architecture

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
<u>UserPresenceInfo</u>	<u>R</u>	<u>string</u> (XML fragment)	See 5.3.2
<u>PresenceOfContactsUpdate</u>	<u>R</u>	<u>string</u> (XML fragment)	See 5.3.3
<u>A_ARG_TYPE_UserPresenceInfo</u>	<u>R</u>	<u>string</u> (XML fragment)	See 5.3.4
<u>A_ARG_TYPE_PresenceOfContactsList</u>	<u>R</u>	<u>string</u> (XML fragment)	See 5.3.5
<u>A_ARG_TYPE_ContactId</u>	<u>R</u>	<u>string</u> (XML fragment)	See 5.3.6
<u>A_ARG_TYPE_RegistrationResult</u>	<u>R</u>	<u>string</u>	See 5.3.7
<u>A_ARG_TYPE_Expires</u>	<u>R</u>	<u>ui4</u>	See 5.3.8
<u>Watcher</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.9
<u>A_ARG_TYPE_WatcherList</u>	<u>A</u>	<u>string</u> (XML fragment)	See 5.3.10

^a [R](#) = required, [A](#) = allowed, [CR](#) = conditionally required, [CA](#) = conditionally allowed, [X](#) = Non-standard, add [-D](#) when deprecated (e.g., [R-D](#), [A-D](#)).

5.3.2 [UserPresenceInfo](#)

The format of the [UserPresenceInfo](#) state variable is an XML document. It includes the presence information (e.g., availability of the user) of the user stored in the TS. This state variable is evented to the TelCP(s) in case of any changes to the presence information of a user.

The presence information is represented in Presence Information Data Format (PIDF) format defined in [11].

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 [UserPresenceInfo](#) in the XML namespace "urn:schemas-upnp-org:phone:presence" which is located at "http://www.upnp.org/schemas/phone/Presence-v1.xsd".

5.3.2.2 Description of fields in the [UserPresenceInfo](#) structure

```
<?xml version="1.0" encoding="UTF-8"?>
<presence:UserPresenceInfo
  xsi:schemaLocation="urn:schemas-upnp-org:phone:presence
  http://www.upnp.org/schemas/phone/presence-v1.xsd"
  xmlns:Presence="urn:schemas-upnp-org:phone:presence"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:pid="urn:ietf:params:xml:ns:pidf">
  <pid:presence entity="pres:someone@example.com">
    <pid:tuple id="bs35r9">
      <pid:statuspid:basic>open</pid:basic>
      </pid:statuspid:contact priority="0.8">someone@mobilecarrier.net</pid:contactpid:note xml:lang="en">Don't Disturb Please!</pid:notepid:note xml:lang="fr">Ne derangez pas, s'il vous plait</pid:notepid:timestamp>2001-10-27T16:49:29Z</pid:timestamp

```



```

    </pid:tuple>
    <!--Any other pid tuple go here -->
  </pid:presence>
</presence:UserPresenceInfo>

```

<xml>

Required. Case Sensitive.

<UserPresenceInfo>

Required. shall include the name space declaration for the complex type <PIDF> ("urn:ietf:params:xml:ns:pidf") and the namespace declaration for the Presence service Schema ("urn:schemas-upnp-org:phone:Presence"). This namespace "urn:schemas-upnp-org:phone:Presence" defines the following elements and attributes:

<presence>

Required. includes the presence information in the format defined by [11]. This element shall include "entity" attribute to identify the contact associated with the presence information. This element includes following sub element as defined in the [11]:

<tuple>

Allowed. carries a presence information as ordered set of in values. Tuples provide a way of segmenting presence information. The <tuple> element shall contain an 'id' attribute which is used to distinguish this tuple from other tuples. Refer to [11] for more information. This element includes following subelements:

<status>

Allowed. represents the availability of a contact. Refer to [11] for more information. This element contains following subelements.

<basic>

Allowed. contains one of the following strings: "open" or "closed". This element is used to represent the availability information for the contact.

<contact>

Allowed. contains a URL of the contact address. It may have a 'priority' attribute, whose value means a relative priority of this contact address over the others. The value of the attribute shall be a decimal number between 0 and 1. Refer to [11] for more information.

<note>

Allowed. contains a string value, which is usually used for a human readable comment.

<timestamp>

Allowed. contains a string indicating the date and time of the status change of the presence tuple. Refer to [11] for more information.

This state variable is used to represent the presence information of the user stored in the TS. This specification requires to use the PIDF format to represent the basic presence information (e.g., status information as online/offline) as defined in the [11]. The IETF also have defined extensive rich set of presence information in [15] and [16] as an extension for the basic PIDF format. The Presence service can also include any extended presence information into the basic information of this state variable by exporting the particular schema.

5.3.3 PresenceOfContactsUpdate

The format of the PresenceOfContactsUpdate state variable is an XML document. It includes the presence information of the contact. This state variable is used to notify the TelCP(s) the change in the presence information for the contact. Each time one or more

contact presence information is updated, the [Presence](#) service updates this state variable with the new information only.

The presence information is represented in Presence Information Data Format (PIDF) format defined in [11].

5.3.3.1 XML Schema Definition

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for [PresenceOfContactsUpdate](#) in the XML namespace "urn:schemas-upnp-org:phone:presence" which is located at "http://www.upnp.org/schemas/phone/Presence-v1.xsd".

5.3.3.2 Description of fields in the [PresenceOfContactsUpdate](#) structure

```
<?xml version="1.0" encoding="UTF-8"?>
<presence:PresenceOfContactsUpdate
  xsi:schemaLocation="urn:schemas-upnp-org:phone:presence
http://www.upnp.org/schemas/phone/presence-v1.xsd"
  xmlns:Presence="urn:schemas-upnp-org:phone:presence"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:pid="urn:ietf:params:xml:ns:pidf"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <remotecontact>
    <peer:id>mayur.patil@samsung.com</peer:id>
    <peer:contactInstanceId>12</peer:contactInstanceId>
    <!-- Any Other peer element if required -->
  </remotecontact>
  <pid:presence entity="pres:someone@example.com">
    <pid:tuple id="bs35r9">
      <pid:status>
        <pid:basic>open</pid:basic>
      </pid:status>
      <pid:contact priority="0.8">someone@mobilecarrier.net</pid:contact>
      <pid:note xml:lang="en">Don't Disturb Please!</pid:note>
      <pid:note xml:lang="fr">Ne derangez pas, s'il vous plait</pid:note>
      <pid:timestamp>2001-10-27T16:49:29Z</pid:timestamp>
    </pid:tuple>
    <!-- Any other pid tuple go here -->
  </pid:presence>
</presence:PresenceOfContactsUpdate>
```

<xml>

Required. Case Sensitive.

<PresenceOfContactsUpdate>

Required. shall include the name space declaration for the complex type <PIDF> ("urn:ietf:params:xml:ns:pidf") and the namespace declaration for the [Presence](#) service Schema ("urn:schemas-upnp-org:phone:Presence"). This namespace "urn:schemas-upnp-org:phone:Presence" defines the following elements and attributes:

<remoteContact>

Required. peer:PeerType, identifies the remote contact whose presence information is updated.

<presence>

Required. includes the presence information in the format defined in [11]. This element shall includes "entity" attribute to identify the contact associated with the presence information. This element includes following sub element as defined in [11]:

<tuple>

Allowed. carries a presence information as ordered set of in values. Tuples provide a way of segmenting presence information. The <tuple> element shall contain an 'id' attribute which is

used to distinguish this tuple from other tuples. Refer to [11] for more information. This element includes following subelements:

`<status>`

Allowed. represents the availability of a contact. Refer to [11] for more information. This element contains following subelements.

`<basic>`

Allowed. contains one of the following strings: "open" or "closed". This element is used to represent the availability information for the contact.

`<contact>`

Allowed. contains a URL of the contact address. It optionally has a 'priority' attribute, whose value means a relative priority of this contact address over the others. The value of the attribute shall be a decimal number between 0 and 1. Refer to [11] for more information.

`<note>`

Allowed. contains a string value, which is usually used for a human readable comment.

`<timestamp>`

Allowed. contains a string indicating the date and time of the status change of the presence tuple. Refer to [11] for more information.

This state variable is used to represent the presence information of the user stored in the TS. This specification requires to use the PIDF format to represent the basic presence information (e.g., status information as online/offline) as defined in the [11]. The IETF also have defined extensive rich set of presence information in [15] and [16] as an extension for the basic PIDF format. The Presence service can also include any extended presence information into the basic information of this state variable by exporting the particular schema.

5.3.4 A_ARG_TYPE UserPresenceInfo

The format of the A_ARG_TYPE UserPresenceInfo state variable is an XML document. It includes the presence information of the user stored in the TS.

The presence information is represented in Presence Information Data Format (PIDF) format defined in [11].

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 A_ARG_TYPE UserPresenceInfo in the XML namespace "urn:schemas-upnp-org:phone:presence" which is located at "http://www.upnp.org/schemas/phone/Presence-v1.xsd".

5.3.4.2 Description of fields in the A_ARG_TYPE UserPresenceInfo structure

```
<?xml version="1.0" encoding="UTF-8"?>
<presence:UserPresence
  xsi:schemaLocation="urn:schemas-upnp-org:phone:presence
  http://www.upnp.org/schemas/phone/presence-v1.xsd"
  xmlns:Presence="urn:schemas-upnp-org:phone:presence"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:pid="urn:ietf:params:xml:ns:pidf">
  <pid:presence entity="pres:someone@example.com">
    <pid:tuple id="adgdh">
      <pid:status>
        <pid:basic>open</pid:basic>
```

```

</pid:status>
<pid:contact priority="0.8">someone@mobilecarrier.net</pid:contact>
<pid:note xml:lang="en">Don't Disturb Please!</pid:note>
<pid:note xml:lang="fr">Ne derangez pas, s'il vous plait</pid:note>
<pid:timestamp>2001-10-27T16:49:29Z</pid:timestamp>
</pid:tuple>
<!-- Any other tuple go here -->
</pid:presence>
</presence:UserPresence>

```

<xml>

Required. Case Sensitive.

<PresenceOfContactsUpdate>

Required. shall include the name space declaration for the complex type <PIDF> ("urn:ietf:params:xml:ns:pidf") and the namespace declaration for the [Presence](#) service Schema ("urn:schemas-upnp-org:phone:Presence"). This namespace "urn:schemas-upnp-org:phone:Presence" defines the following elements and attributes:

<presence>

Required. includes the presence information in the format defined in [11]. This element shall include "entity" attribute to identify the contact associated with the presence information. This element includes following sub element as defined in [11]:

<tuple>

Allowed. carries a presence information as ordered set of in values. Tuples provide a way of segmenting presence information. The <tuple> element shall contain an 'id' attribute which is used to distinguish this tuple from other tuples. Refer to [11] for more information. This element includes following subelements:

<status>

Allowed. represents the availability of a contact. Refer to [11] for more information. This element contains following subelements.

<basic>

Allowed. contains one of the following strings: "open" or "closed". This element is used to represent the availability information for the contact.

<contact>

Allowed. contains a URL of the contact address. It optionally has a 'priority' attribute, whose value means a relative priority of this contact address over the others. The value of the attribute shall be a decimal number between 0 and 1. Refer to [11] for more information.

<note>

Allowed. contains a string value, which is usually used for a human readable comment.

<timestamp>

Allowed. contains a string indicating the date and time of the status change of the presence tuple. Refer to [11] for more information.

This state variable is used to represent the presence information of the user stored in the TS. This specification requires to use the PIDF format to represent the basic presence information (e.g., status information as online/offline) as defined in the [11]. The IETF also have defined extensive rich set of presence information in [15] and [16] as an extension for the basic PIDF format. The [Presence](#) service can also include any extended presence information into the basic information of this state variable by exporting the particular schema.

5.3.5 A_ARG_TYPE PresenceOfContactsList

The format of the A_ARG_TYPE PresenceOfContactsList state variable is an XML document. It includes the presence information of the contact.

The presence information is represented in Presence Information Data Format (PIDF) format defined in [11].

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 A_ARG_TYPE PresenceOfContactsList in the XML namespace "urn:schemas-upnp-org:phone:messaging" which is located at "http://www.upnp.org/schemas/phone/Addressbook-v1.xsd".

5.3.5.2 Description of fields in the A_ARG_TYPE PresenceOfContactsList structure

```
<?xml version="1.0" encoding="UTF-8"?>
<presence:UserPresence
  xsi:schemaLocation="urn:schemas-upnp-org:phone:presence
http://www.upnp.org/schemas/phone/presence-v1.xsd"
  xmlns:Presence="urn:schemas-upnp-org:phone:presence
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:pid="urn:ietf:params:xml:ns:pidf">
  <pid:presence entity="pres:someone@example.com">
    <pid:tuple id="adgdh">
      <pid:status>
        <pid:basic>open</pid:basic>
      </pid:status>
      <pid:contact priority="0.8">someone@mobilecarrier.net</pid:contact>
      <pid:note xml:lang="en">Don't Disturb Please!</pid:note>
      <pid:note xml:lang="fr">Ne derangez pas, s'il vous plait</pid:note>
      <pid:timestamp>2001-10-27T16:49:29Z</pid:timestamp>
    </pid:tuple>
    <!-- Any other tuple go here -->
  </pid:presence>
</presence:UserPresence>
```

<xml>

Required. Case Sensitive.

<PresenceOfContactsUpdate>

Required. shall include the name space declaration for the complex type <PIDF> ("urn:ietf:params:xml:ns:pidf") and the namespace declaration for the Presence service Schema ("urn:schemas-upnp-org:phone:Presence"). This namespace "urn:schemas-upnp-org:phone:Presence" defines the following elements and attributes:

<presence>

Required. includes the presence information in the format defined in [11]. This element shall include "entity" attribute to identify the contact associated with the presence information. This element includes following sub element as defined in [11]:

<tuple>

Allowed. carries a presence information as ordered set of in values. Tuples provide a way of segmenting presence information. The <tuple> element shall contain an 'id' attribute which is used to distinguish this tuple from other tuples. Refer to [11] for more information. This element includes following subelements:

<status>

Allowed. represents the availability of a contact. Refer to [11] for more information. This element contains following subelements.

<basic>

Allowed. contains one of the following strings: "open" or "closed". This element is used to represent the availability information for the contact.

<contact>

Allowed. contains a URL of the contact address. It optionally has a 'priority' attribute, whose value means a relative priority of this contact address over the others. The value of the attribute shall be a decimal number between 0 and 1. Refer to [11] for more information.

<note>

Allowed. contains a string value, which is usually used for a human readable comment.

<timestamp>

Allowed. contains a string indicating the date and time of the status change of the presence tuple. Refer to [11] for more information.

This state variable is used to represent the presence information of the user stored in the TS. This specification requires to use the PIDF format to represent the basic presence information (e.g., status information as online/offline) as defined in the [11]. The IETF also have defined extensive rich set of presence information in [15] and [16] as an extension for the basic PIDF format. The [Presence](#) service can also include any extended presence information into the basic information of this state variable by exporting the particular schema.

5.3.6 [A ARG TYPE ContactId](#)

This state variable identifies one or more contact(s). This element is an XML structure of type [Peer](#).

5.3.6.1 Description of fields in the [A ARG TYPE ContactId](#)

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for [A ARG TYPE ContactId](#) in the XML namespace "urn:schemas-upnp-org:phone:presence" which is located at "http://www.upnp.org/schemas/phone/Presence-v1.xsd".

```
<?xml version="1.0" encoding="UTF-8"?>
<contactId
  xsi:schemaLocation="urn:schemas-upnp-org:phone:presence
http://www.upnp.org/schemas/phone/presence-v1.xsd"
  xmlns:Presence="urn:schemas-upnp-org:phone:presence"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <contactpeer:id>ID of the peer</peer:id>
    <peer:contactInstanceId>instance id of the contact</peer:contactInstanceId>
    Other peer elements if required...
  </contact>
  <!-- Any other contact information (if any) go here.-->
</contactId>
```

<xml>

Required. Case Sensitive.

<contactId>

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

<contact>

Required, peer:peerType. identifies the contact to be shared. If contact to be shared is present in the local address book then, the contactInstanceId element may be included to refer the contact from the address book otherwise the Id element will identify the contact. A TelCP can use the any other peer elements (e.g., name etc) to identify the contact.

5.3.7 A ARG TYPE RegistrationResult

This state variable contains the subscription request result. This state variable is of a type string. This state variable contains one of the following values.

- "accepted".
- "rejected".
- "pending".

5.3.8 A ARG TYPE Expires

This state variable contains the duration as an integer value.

5.3.9 Watcher

The format of the Watcher state variable is an XML document. It includes the remote party contact information who is requesting for the presence information. This state variable is evented to the TelCP(s), when TS receives an incoming request from the remote party for sharing the presence information.

5.3.9.1 Description of fields in the Watcher

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for Watcher in the XML namespace "urn:schemas-upnp-org:phone:presence" which is located at "http://www.upnp.org/schemas/phone/Presence-v1.xsd".

```
<?xml version="1.0" encoding="UTF-8"?>
<presence:Watcher
  xsi:schemaLocation="urn:schemas-upnp-org:phone:presence
  http://www.upnp.org/schemas/phone/presence-v1.xsd"
  xmlns:Presence="urn:schemas-
```

<watcher_info>

Required. peer:PeerType, identifies the remote contact requesting the presence information.

<peer:id>

Allowed. identifies the remote contact who is requesting for the presence information.

<peer:contactInstanceId>

Allowed. references the contact from the address book who is requesting for the presence information.

<presence_info>

Allowed. represents the requested presence information (or filters) by the remote party. The format attribute will specify the format in which the presence information is specified. The TS includes the presence information of the user which is requested by the watcher in the PIDF format.

5.3.10 A ARG TYPE WatcherList

The format of the A ARG TYPE WatcherList state variable is an XML document. It includes the list of Watchers currently requesting or already subscribed for the presence information.

5.3.10.1 Description of fields in the A ARG TYPE WatcherList

This is a string containing an XML fragment. The XML fragment in this argument shall validate against the XML schema for A ARG TYPE WatcherList in the XML namespace "urn:schemas-upnp-org:phone:presence" which is located at "http://www.upnp.org/schemas/phone/Presence-v1.xsd".

```
<?xml version="1.0" encoding="UTF-8"?>
<presence:WatcherList
  xsi:schemaLocation="urn:schemas-upnp-org:phone:presence
http://www.upnp.org/schemas/phone/presence-v1.xsd"
  xmlns:Presence="urn:schemas-upnp-org:phone:presence"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:peer="urn:schemas-upnp-org:phone:peer">
  <watcher>
    <watcher_info>
      <peer:id>ID of the peer</peer:id>
      <peer:contactInstanceId>instanceID of the
contact</peer:contactInstanceId>
    </watcher_info>
    <presence_info format="RFC4660/RFC4661/URL">
      requested presence information
    </presence_info>
  </watcher>
  <!-- Any other watcher information goes here -->
</presence:WatcherList>
```

<xml>

Required. Case Sensitive.

<WatcherList>

Required. shall include the name space declaration for the complex type <PeerType> ("urn:schemas-upnp-org:phone:peer") and the namespace declaration for the Presence service Schema ("urn:schemas-upnp-org:phone:Presence"). This namespace "urn:schemas-upnp-org:phone:Presence" defines the following elements and attributes:

<watcher>

Required. carries the Watcher information and current status of the subscription. This elements has following sub-elements.

<watcher_info>

Required. peer:PeerType, identifies the remote contact requesting the presence information.

<peer:id>

Allowed. identifies the remote contact who is requesting for the presence information.

<peer:contactInstanceId>

Allowed. references the contact from the address book who is requesting for the presence information.

<presence_info>

Allowed. represents the requested presence information (or filters) by the remote party. The format attribute will specify the format in which the filters are specified. The IETF has defined the filtering mechanism in [12], [13]. The TS export the structure defined by the corresponding RFCs.

5.4 Eventing and Moderation

Table 2 — Event Moderation

Variable Name	Evented	Moderated Event	Max Event Rate ^a (seconds)	Logical Combination	Min Delta per Event ^b
<u>UserPresenceInfo</u>	<u>YES</u>	<u>YES</u>	1		
<u>PresenceOfContactsUpdate</u>	<u>YES</u>	<u>YES</u>	1		
<u>Watcher</u>	<u>YES</u>	<u>YES</u>	1		
<u>A_ARG_TYPE_UserPresenceInfo</u>	<u>No</u>	<u>No</u>			
<u>A_ARG_TYPE_PresenceOfContactsList</u>	<u>No</u>	<u>No</u>			
<u>A_ARG_TYPE_ContactId</u>	<u>No</u>	<u>No</u>			
<u>A_ARG_TYPE_RegistrationResult</u>	<u>No</u>	<u>No</u>			
<u>A_ARG_TYPE_Expires</u>	<u>No</u>	<u>No</u>			
<u>A_ARG_TYPE_WatcherList</u>	<u>No</u>	<u>No</u>			
^a Determined by N, where Rate = (Event)/(N secs).					
^b (N) * (allowedValueRange Step).					

5.4.1 Eventing of UserPresenceInfo

This state variable is evented when a user presence information changes. However, this state variable shall not be evented more than once in 1 s. If multiple changes arrive within 1 s, then all the events will be accumulated into a single event message and the event message will be sent after 1 s.

5.4.2 Eventing of PresenceOfContactsUpdate

This state variable is evented when presence information of a particular contact changes. However, this state variable shall not be evented more than once in 1 s. If multiple change requests arrive within 1 s, then all the events will be accumulated into a single event message and the event message will be sent after 1 s.

5.4.3 Eventing of Watcher

This state variable is evented when a TS receives the request from the contact for sharing the presence information from WAN side. However, this state variable shall not be evented

more than once in 1 s. If multiple requests arrive within 1 s, then all the events will be accumulated into a single event message and the event message will be sent after 1 s.

5.5 Actions

Table 3 lists the actions of the Presence service.

The Presence service requires the implementation of all the actions necessary for delivering the complete set of functions.

Table 3 — Actions

Name	Device R/A ^a	Control Point R/A ^b
<u>GetPresence()</u>	<u>A</u>	<u>A</u>
<u>UpdatePresence()</u>	<u>R</u>	<u>R</u>
<u>GetContactPresence()</u>	<u>R</u>	<u>R</u>
<u>GetPresenceOfContactsUpdate()</u>	<u>A</u>	<u>A</u>
<u>RegisterForContactPresence()</u>	<u>R</u>	<u>R</u>
<u>AuthorizePresenceReactive()</u>	<u>A</u>	<u>A</u>
<u>AuthorizePresenceProactive()</u>	<u>A</u>	<u>A</u>
^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>CA</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>CA</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 GetPresence()

This action allows a TelCP to retrieve the user's presence information stored in the TS.

5.5.1.1 Arguments

Table 4 — Arguments for GetPresence()

Argument	Direction	relatedStateVariable
<u>UserPresence</u>	<u>OUT</u>	<u>A_ARG_TYPE_UserPresenceInfo</u>

5.5.1.2 Argument Descriptions

The output argument UserPresence returns the user's current presence information.

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 5 — Error Codes for GetPresence()**

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.

5.5.2 UpdatePresence()

This action allows a TelCP to updates the user's presence information in the TS.

5.5.2.1 Arguments**Table 6 — Arguments for UpdatePresence()**

Argument	Direction	relatedStateVariable
<u>UpdatedUserPresence</u>	<u>IN</u>	<u>A_ARG_TYPE_UserPresenceInfo</u>

5.5.2.2 Argument Descriptions

The input argument UpdatedUserPresence will include the updated presence information.

5.5.2.3 Service Requirements

The input argument will carry the updated presence information. Only the tuple elements included in the input argument will be updated in the presence information of the user, other tuple elements in the presence information will not be modified.

5.5.2.4 Control Point Requirements When Calling The Action

None.

5.5.2.5 Dependency on Device State

None.

5.5.2.6 Effect on Device State

None.

5.5.2.7 Errors

Table 7 — Error Codes for UpdatePresence()

ErrorCode	errorDescription	Description
400-499	TBD	See UPnP Device Architecture clause on Control.
500-599	TBD	See UPnP Device Architecture clause on Control.
600-699	TBD	See UPnP Device Architecture clause on Control.

5.5.3 GetContactPresence()

This action allows a TelCP to retrieve the presence information of a contact.

5.5.3.1 Arguments

Table 8 — Arguments for GetContactPresence()

Argument	Direction	relatedStateVariable
<u>TargetContact</u>	<u>IN</u>	<u>A_ARG_TYPE_ContactId</u>
<u>ContactPresence</u>	<u>OUT</u>	<u>A_ARG_TYPE_PresenceOfContactsList</u>

5.5.3.2 Argument Descriptions

The input argument TargetContact identifies the target contact.

The output argument ContactPresence contains the presence information of the contact.

5.5.3.3 Service Requirements

None.

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 9 — Error Codes for GetContactPresence()

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

ErrorCode	errorDescription	Description
400-499	TBD	See UPnP Device Architecture clause on Control.
500-599	TBD	See UPnP Device Architecture clause on Control.
600-699	TBD	See UPnP Device Architecture clause on Control.
770	Invalid target contact	The target contact does not exist.

5.5.4 GetPresenceOfContactsUpdate()

This action allows a TelCP to retrieve the updated presence information for the contact(s) which was evented by the PresenceOfContactsUpdate state variable.

5.5.4.1 Arguments

Table 10 — Arguments for GetPresenceOfContactsUpdate()

Argument	Direction	relatedStateVariable
<u>ContactPresenceUpdate</u>	<u>OUT</u>	<u>A_ARG_TYPE_PresenceOfContactsList</u>

5.5.4.2 Argument Descriptions

The output argument ContactPresenceUpdate contains the presence information of the contact(s).

5.5.4.3 Service Requirements

None.

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 11 — Error Codes for GetPresenceOfContactsUpdate()

ErrorCode	errorDescription	Description
400-499	TBD	See UPnP Device Architecture clause on Control.
500-599	TBD	See UPnP Device Architecture clause on Control.
600-699	TBD	See UPnP Device Architecture clause on Control.

5.5.5 RegisterForContactPresence()

This action allows a TelCP to subscribe for the presence information of a contact. The same action can be used to unsubscribe the presence information for a contact by setting the value of the input argument Expire as "0".

5.5.5.1 Arguments

Table 12 — Arguments for RegisterForContactPresence()

Argument	Direction	relatedStateVariable
<u>Contact</u>	<u>IN</u>	<u>A_ARG_TYPE_ContactId</u>
<u>Expire</u>	<u>IN</u>	<u>A_ARG_TYPE_Expires</u>
<u>RegistrationResult</u>	<u>OUT</u>	<u>A_ARG_TYPE_RegistrationResult</u>

5.5.5.2 Argument Descriptions

The input argument Contact identifies the remote party.

The input argument Expire contains the duration for the subscription. If the value of the Expire input argument is set to "0" then the current subscription will be terminated by the TS.

The output argument RegistrationResult is used to notify the result for the subscription request.

5.5.5.3 Service Requirements

None.

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 13 — Error Codes for RegisterForContactPresence()

ErrorCode	errorDescription	Description
400-499	TBD	See UPnP Device Architecture clause on Control.
500-599	TBD	See UPnP Device Architecture clause on Control.
600-699	TBD	See UPnP Device Architecture clause on Control.
770	Invalid target contact	The target contact does not exist.

5.5.6 AuthorizePresenceReactive()

This action allows a TelCP to authorize a contact to view the presence information stored in the Telephony Server (TS). This action is invoked in response to an event sent by the TS. When the TS receives a request for the presence information from the WAN side, the TS sends out event notifications to all the subscribed control points in the home network about the incoming requests. A TelCP can then accept or reject the request by invoking this action on the TS, which is then propagated to the requester using a WAN-side protocol by the presence server.

5.5.6.1 Arguments

Table 14 — Arguments for AuthorizePresenceReactive()

Argument	Direction	relatedStateVariable
<u>Contact</u>	<u>IN</u>	<u>A_ARG_TYPE_ContactId</u>
<u>Expire</u>	<u>IN</u>	<u>A_ARG_TYPE_Expires</u>
<u>UserPresenceInfo</u>	<u>IN</u>	<u>A_ARG_TYPE_UserPresenceInfo</u>

5.5.6.2 Argument Descriptions

The input argument Contact identifies the remote peer contact that requested for the presence information of the user.

The input argument Expire contains the duration for which this authorization is valid. If the value of the input argument Expire is set to “0”, then the authorization to view the presence information of the user is denied.

The input argument UserPresenceInfo specifies the presence information to be shared with the contact.

5.5.6.3 Service Requirements

None.

5.5.6.4 Control Point Requirements When Calling The Action

None.

5.5.6.5 Dependency on Device State

None.

5.5.6.6 Effect on Device State

None.

5.5.6.7 Errors

Table 15 — Error Codes for AuthorizePresenceReactive()

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

ErrorCode	errorDescription	Description
400-499	TBD	See UPnP Device Architecture clause on Control.
500-599	TBD	See UPnP Device Architecture clause on Control.
600-699	TBD	See UPnP Device Architecture clause on Control.
770	Invalid contact	The contact does not exist

5.5.7 AuthorizePresenceProactive()

This action allows a TelCP to authorize (allow or block) a list of contacts to view the presence information of the user stored in the TS. The list of entities on the WAN side that are authorized to view the presence information of the user (TS with Presence Service) can be set up by this action even before the request for presence information is received by the TS. The TelCP invokes this action with a list of WAN contacts that are allowed to access the presence information.

5.5.7.1 Arguments

Table 16 — Arguments for AuthorizePresenceProactive()

Argument	Direction	relatedStateVariable
<u>UserPresenceInfo</u>	<u>IN</u>	<u>A_ARG_TYPE_UserPresenceInfo</u>
<u>Expire</u>	<u>IN</u>	<u>A_ARG_TYPE_Expires</u>
<u>WatcherList</u>	<u>IN</u>	<u>A_ARG_TYPE_WatcherList</u>

5.5.7.2 Argument Descriptions

The input argument UserPresenceInfo contains the presence information of the user which a watcher is authorized to view.

The input argument Expire contains the duration for which this authorization is valid. If the value of the input argument Expire is set to “0”, then the authorization to view the presence information is blocked for the contacts in the WatcherList.

The output argument WatcherList includes the list of contacts that can view the presence information of the user.

5.5.7.3 Service Requirements

When the presence service within a TS receives a request to access presence information from a contact, the TS checks its internal database to see whether the contact is allowed to access the presence information. The server can then decide to allow or deny the request to the WAN side without even consulting the devices in the home network. This is proactive authorization mechanism.

5.5.7.4 Control Point Requirements When Calling The Action

None.

5.5.7.5 Dependency on Device State

None.

5.5.7.6 Effect on Device State

None.

5.5.7.7 Errors

Table 17 — Error Codes for AuthorizePresenceProactive()

ErrorCode	errorDescription	Description
400-499	TBD	See UPnP Device Architecture clause on Control.
500-599	TBD	See UPnP Device Architecture clause on Control.
600-699	TBD	See UPnP Device Architecture clause on Control.

5.5.8 Error Code Summary

Table 18 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 18 — 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.
770	Invalid contact	The contact does not exist.

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

5.6 Service Behavioral Model

None.

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>GetPresence</name>
      <argumentList>
        <argument>
          <name>UserPresence</name>
          <direction>OUT</direction>
          <relatedStateVariable>
            A_ARG_TYPE_UserPresenceInfo
          </relatedStateVariable>
        </argument>
      </argumentList>
    </action>
  </actionList>
</scpd>
```

```

</action>

<action>
  <name>UpdatePresence</name>
  <argumentList>
    <argument>
      <name>UpdatedUserPresence</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_UserPresenceInfo
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>GetContactPresence</name>
  <argumentList>
    <argument>
      <name>TargetContact</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_ContactId
      </relatedStateVariable>
    </argument>

    <argument>
      <name>ContactPresence</name>
      <direction>out</direction>
      <relatedStateVariable>
        A_ARG_TYPE_PresenceOfContactsList
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>GetPresenceOfContactsUpdate</name>
  <argumentList>
    <argument>
      <name>ContactPresenceUpdate</name>
      <direction>out</direction>
      <relatedStateVariable>
        A_ARG_TYPE_PresenceOfContactsList
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>RegisterForContactPresence</name>
  <argumentList>
    <argument>
      <name>Contact</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_ContactId
      </relatedStateVariable>
    </argument>

    <argument>
      <name>Expire</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_Expires
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

```

    <argument>
      <name>RegistrationResult</name>
      <direction>out</direction>
      <relatedStateVariable>
        A_ARG_TYPE_RegistrationResult
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>AuthorizePresenceReactive</name>
  <argumentList>
    <argument>
      <name>Contact</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_ContactId
      </relatedStateVariable>
    </argument>

    <argument>
      <name>Expire</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_Expires
      </relatedStateVariable>
    </argument>

    <argument>
      <name>UserPresenceInfo</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_UserPresenceInfo
      </relatedStateVariable>
    </argument>
  </argumentList>
</action>

<action>
  <name>AuthorizePresenceProactive</name>
  <argumentList>
    <argument>
      <name>UserPresenceInfo</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_UserPresenceInfo
      </relatedStateVariable>
    </argument>

    <argument>
      <name>Expire</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_Expires
      </relatedStateVariable>
    </argument>

    <argument>
      <name>WatcherList</name>
      <direction>in</direction>
      <relatedStateVariable>
        A_ARG_TYPE_WatcherList
      </relatedStateVariable>
    </argument>
  </argumentList>

```

```

    </action>
</actionList>
<serviceStateTable>
  <stateVariable sendEvents="yes">
    <name>UserPresenceInfo</name>
    <dataType>string</dataType>
  </stateVariable>

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

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

  <stateVariable sendEvents="no">
    <name>A_ARG_TYPE_WatcherList</name>
    <dataType>string</dataType>
  </stateVariable>

  <stateVariable sendEvents="no">
    <name>A_ARG_TYPE_UserPresenceInfo</name>
    <dataType>string</dataType>
  </stateVariable>

  <stateVariable sendEvents="no">
    <name>A_ARG_TYPE_PresenceOfContactsList</name>
    <dataType>string</dataType>
  </stateVariable>

  <stateVariable sendEvents="no">
    <name>A_ARG_TYPE_ContactId</name>
    <dataType>string</dataType>
  </stateVariable>

  <stateVariable sendEvents="no">
    <name>A_ARG_TYPE_RegistrationResult</name>
    <dataType>string</dataType>
  </stateVariable>

  <stateVariable sendEvents="no">
    <name>A_ARG_TYPE_Expires</name>
    <dataType>Integer</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:image>
  <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 thumbnail image data information. This element may 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 by [3]. 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 defines by [3]. 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 will 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:peer="urn:schemas-upnp-org:phone:peer"
targetNamespace="urn:schemas-upnp-org:phone:peer" elementFormDefault="qualified"
attributeFormDefault="qualified" version="1">
  <complexType name="peerType">
    <sequence>
      <element name="id" nillable="0">
        <annotation>
          <documentation>Id of the peer. The content depends on the context. For example
it can be a phone number, ad e-mail address and so on.</documentation>
        </annotation>
        <complexType>
          <simpleContent>
            <extension base="string"/>
          </simpleContent>
        </complexType>
      </element>
      <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" nillable="0">
          <annotation>
            <documentation>The Instance Identifier of a Contact in the PDM address
book.</documentation>
          </annotation>
          <complexType>
            <simpleContent>
              <extension base="unsignedInt"/>
            </simpleContent>
          </complexType>
        </element>
        <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>
    </sequence>
  </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>
```


Annex B (normative)

XML Schema

Annex B provides the global XML Schema for syntactical validation of all the XML fragments used in the [Presence](#) service.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:presence="urn:schemas-upnp-org:phone:Presence"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:pid="urn:ietf:params:xml:ns:pidf"
  xmlns:peer="urn:schemas-upnp-org:phone:peer" targetNamespace="urn:schemas-upnp-
  org:phone:Presence" elementFormDefault="unqualified" attributeFormDefault="unqualified">
  <xs:import namespace="urn:ietf:params:xml:ns:pidf"
  schemaLocation="http://tools.ietf.org/rfc/rfc3863.txt"/>
  <xs:import namespace="urn:schemas-upnp-org:phone:peer"
  schemaLocation="http://www.upnp.org/schemas/phone/peer-v2.xsd"/>
  <xs:element name="UserPresenceInfo">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="presence" type="pid:presence"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="PresenceOfContactsUpdate">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="remotecontact" type="peer:peerType"/>
        <xs:element name="presence" type="pid:presence"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="UserPresence">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="presence" type="pid:presence"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="PresenceOfContactsList">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="presence" type="presence:presence" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ContactId">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="contact" type="peer:peerType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="RegistrationResult">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="accepted"/>
        <xs:enumeration value="rejected"/>
        <xs:enumeration value="pending"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="Expires" type="xs:integer"/>
  <xs:element name="Watcher">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="watcher_info" type="peer:peerType" minOccurs="0"/>
        <xs:element name="presence_info">
          <xs:complexType>
            <xs:attribute name="format" type="xs:string"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
</xs:element>
<xs:element name="WatcherList">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element ref="presence:Watcher" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:complexType name="presence">
  <xs:complexContent>
    <xs:extension base="pid:presence">
      <xs:sequence>
        <xs:element name="remotecontact" type="peer:peerType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="presence_info">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="format" type="xs:string"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
</xs:schema>
```

Annex C (informative)

Theory of Operation

C.1 Registration and Notification for Presence

The Telephony Server provides a registration action for a TelCP to subscribe for presence information of a contact in the WAN side. The [UpdatePresence\(\)](#) action allows a TelCP to publish the user's presence in the Telephony Server.

When a Telephony Server receives a request for presence information it may respond to it immediately (proactive response) or may attempt to contact control points which have registered with the Telephony Server (reactive response). The Telephony Server initiates a request for reactive control point responses by sending the [Watcher](#) event.

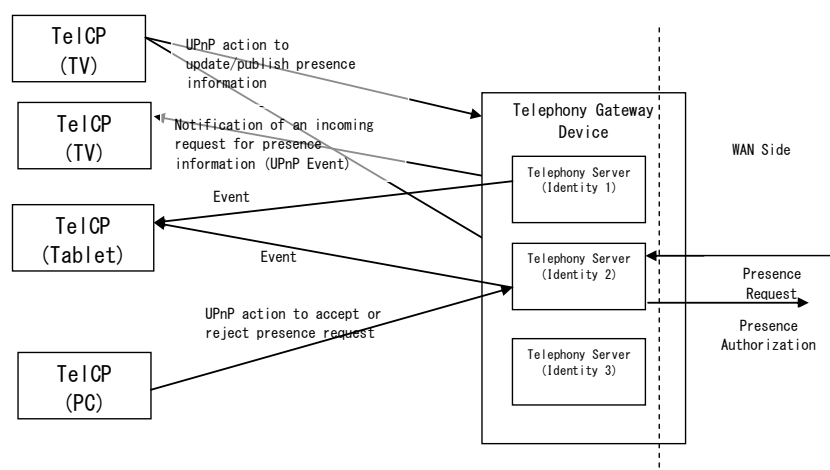


Figure C.1 — Presence registration and notification

C.2 Reactive Authorization of a Request for Presence Information

The Telephony Control Point (TelCP) publishes or updates the presence information of the Telephony Server by invoking an UPnP action on the server. When the Telephony Server receives a request for the presence information from the WAN side, the Telephony server sends out event notifications to all the subscribed TelCP in the home network about the incoming request. A TelCP then can accept or reject the request by invoking [AuthorizePresenceReactive\(\)](#) action on the Telephony Server which is then propagated to the requester using WAN side protocol by the Telephony Server. This generic scenario of accepting or rejecting a presence request is shown in Figure C.2.

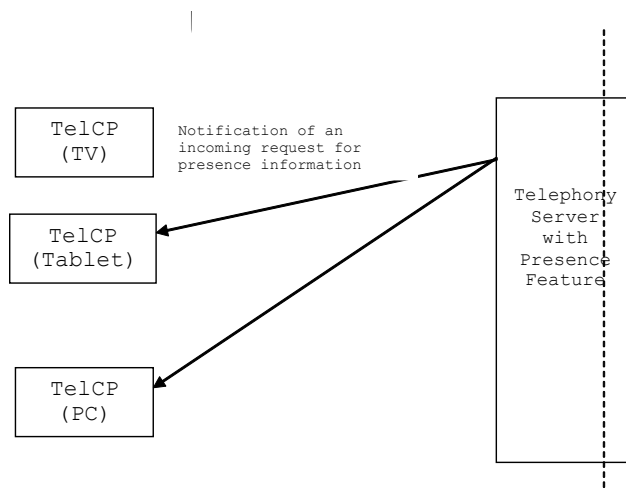


Figure C.2 — Reactive Authorization of a request for Presence information

C.3 Pro-active Authorization for Presence Information

The list of entities in the WAN side that are authorized to view the presence information of the Telephony Server is set by an UPnP action even before the request for presence information is received. In this scenario, a TelCP invokes the [*AuthorizePresenceProactive\(\)*](#) with a list of WAN entities (identified by contact which includes phone number or SIP URL) that are allowed to access Telephony Server's presence information. When the Telephony Server receives a request to access presence information; the server checks its internal database to see whether the entity is allowed to access the presence information. The server then decides to allow/deny the request to the WAN side without consulting the TelCP(s) in the home network.

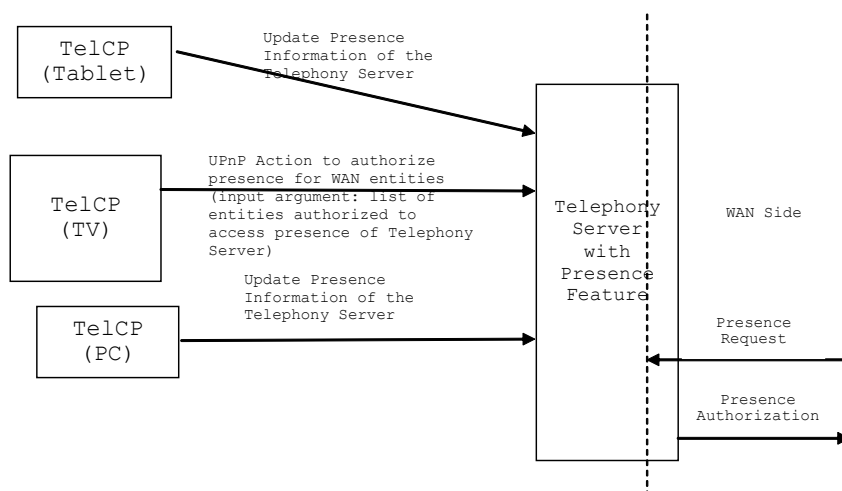


Figure C.3 — Proactive Authorization for Presence information

A TelCP can subscribe to multiple Telephony Servers. Each Telephony Server has its own identity and presence information. The Telephony Server(s) presence information is the same as user's presence information stored in the TS. A TelCP can update presence information of multiple Telephony Servers.

Annex D (informative)

Bibliography

The following documents, in whole or in part, may be useful for understanding this document but they are not essential 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.

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

[15] – IETF RFC 4479, A Data Model for Presence, J. Rosenberg, Cisco Systems, July 2006. Available at: <http://www.ietf.org/rfc/rfc4479.txt>.

[16] – IETF RFC 4480, RPID: Rich Presence Extensions to the Presence Information Data Format (PIDF), H. Schulzrinne, Columbia U., V. Gurbani, Lucent, P. Kyzivat, J. Rosenberg, Cisco, July 2006. Available at: <http://www.ietf.org/rfc/rfc4480.txt>.