
Contents

1	Linux SDK for UPnP Devices v1.4	2
1.1	Introduction	3
1.2	License	3
1.3	Upnp_Action_Request — Returned as part of a UPNP_CONTROL_ACTION_COMPLETE callback.	4
1.4	Upnp_State_Var_Request — Represents the request for current value of a state variable in a service state table.	6
1.5	Upnp_State_Var_Complete — Represents the reply for the current value of a state variable in an asynchronous call.	9
1.6	Upnp_Event — Returned along with a UPNP_EVENT_RECEIVED callback.	10
1.7	Upnp_Discovery — Returned in a UPNP_DISCOVERY_RESULT callback.	11
1.8	Upnp_Event_Subscribe — Returned along with a UPNP_EVENT_SUBSCRIBE_COMPLETE or UPNP_EVENT_UNSUBSCRIBE_COMPLETE callback.	14
1.9	Upnp_Subscription_Request — Returned along with a UPNP_EVENT_SUBSCRIPTION_REQUEST callback.	15
2	UpnpAddToPropertySet	17

Linux SDK for UPnP Devices v1.4

Names

1.1		Introduction	3
1.2		License	3
1.3	struct	Upnp_Action_Request	<i>Returned as part of a UPNP_CONTROL_ACTION_COMPLETE callback.</i>	4
1.4	struct	Upnp_State_Var_Request	<i>Represents the request for current value of a state variable in a service state table.</i>	6
1.5	struct	Upnp_State_Var_Complete	<i>Represents the reply for the current value of a state variable in an asynchronous call.</i>	9
1.6	struct	Upnp_Event	<i>Returned along with a UPNP_EVENT_RECEIVED call- back.</i>	10
1.7	struct	Upnp_Discovery	<i>Returned in a UPNP_DISCOVERY_RESULT callback.</i>	11
1.8	struct	Upnp_Event_Subscribe	<i>Returned along with a UPNP_EVENT_SUBSCRIBE_COMPLETE or UPNP_EVENT_UNSUBSCRIBE_COMPLETE callback.</i>	14
1.9	struct	Upnp_Subscription_Request	<i>Returned along with a UPNP_EVENT_SUBSCRIPTION_REQUEST callback.</i>	15

Linux SDK for UPnP Devices Version 1.4

Copyright (C) 2000-2003 Intel Corporation ALL RIGHTS RESERVED

Revision 1.4.1 (20170921 173014)

1.1

Introduction

This document gives a brief description of the Linux SDK for UPnP Devices API. Section 1 covers the license under which the SDK is distributed. Section 2 talks about the callback functions used in many parts of the API. Finally, section 3 details the structures and functions that comprise the API.

The Linux SDK for UPnP Devices version 1.2 supports the following platforms:

- Linux* running on an Intel Architecture processor
- Linux running on an Intel StrongARM or XScale processor

* Other brands and names are the property of their respective owners.

1.2

License

Copyright (c) 2000-2003 Intel Corporation All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL INTEL OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.3

```
struct Upnp_Action_Request
```

Returned as part of a UPNP_CONTROL_ACTION_COMPLETE callback.

Members

1.3.1	int	ErrCode	<i>The result of the operation.</i>	4
1.3.2	int	Socket	<i>The socket number of the connection to the requestor.</i>	5
1.3.3	char	ErrStr [LINE_SIZE]	<i>The error string in case of error.</i>	5
1.3.4	char	ActionName [NAME_SIZE]	<i>The Action Name.</i>	5
1.3.5	char	DevUDN [NAME_SIZE]	<i>The unique device ID.</i>	5
1.3.6	char	ServiceID [NAME_SIZE]	<i>The service ID.</i>	5
1.3.7	IXML_Document*	ActionRequest	<i>The DOM document describing the action.</i>	6
1.3.8	IXML_Document*	ActionResult	<i>The DOM document describing the result of the action.</i>	6
1.3.9	struct sockaddr_storage	CtrlPtIPAddr	<i>IP address of the control point requesting this action.</i>	6
1.3.10	IXML_Document*	SoapHeader	<i>The DOM document containing the information from the the SOAP header.</i>	6

Returned as part of a UPNP_CONTROL_ACTION_COMPLETE callback.

1.3.1

```
int ErrCode
```

The result of the operation.

The result of the operation.

1.3.2

`int Socket`

The socket number of the connection to the requestor.

The socket number of the connection to the requestor.

1.3.3

`char ErrStr [LINE_SIZE]`

The error string in case of error.

The error string in case of error.

1.3.4

`char ActionName [NAME_SIZE]`

The Action Name.

The Action Name.

1.3.5

`char DevUDN [NAME_SIZE]`

The unique device ID.

The unique device ID.

1.3.6

`char ServiceID [NAME_SIZE]`

The service ID.

The service ID.

1.3.7

IXML_Document* **ActionRequest**

The DOM document describing the action.

The DOM document describing the action.

1.3.8

IXML_Document* **ActionResult**

The DOM document describing the result of the action.

The DOM document describing the result of the action.

1.3.9

struct sockaddr_storage **CtrlPtIPAddr**

IP address of the control point requesting this action.

IP address of the control point requesting this action.

1.3.10

IXML_Document* **SoapHeader**

The DOM document containing the information from the the SOAP header.

The DOM document containing the information from the the SOAP header.

1.4

struct **Upnp_State_Var_Request**

Represents the request for current value of a state variable in a service state table.

Members

1.4.1	int	ErrCode	<i>The result of the operation.</i>	7
1.4.2	int	Socket	<i>The socket number of the connection to the requestor.</i>	7
1.4.3	char	ErrStr [LINE_SIZE]	<i>The error string in case of error.</i>	7
1.4.4	char	DevUDN [NAME_SIZE]	<i>The unique device ID.</i>	8
1.4.5	char	ServiceID [NAME_SIZE]	<i>The service ID.</i>	8
1.4.6	char	StateVarName [NAME_SIZE]	<i>The name of the variable.</i>	8
1.4.7	struct sockaddr_storage	CtrlPtIPAddr	<i>IP address of sender requesting the state variable.</i>	8
1.4.8	DOMString	CurrentVal	<i>The current value of the variable.</i>	8

Represents the request for current value of a state variable in a service state table.

1.4.1

int **ErrCode**

The result of the operation.

The result of the operation.

1.4.2

int **Socket**

The socket number of the connection to the requestor.

The socket number of the connection to the requestor.

1.4.3

char **ErrStr** [LINE_SIZE]

The error string in case of error.

The error string in case of error.

1.4.4

```
char DevUDN [NAME_SIZE]
```

The unique device ID.

The unique device ID.

1.4.5

```
char ServiceID [NAME_SIZE]
```

The service ID.

The service ID.

1.4.6

```
char StateVarName [NAME_SIZE]
```

The name of the variable.

The name of the variable.

1.4.7

```
struct sockaddr_storage CtrlPtIPAddr
```

IP address of sender requesting the state variable.

IP address of sender requesting the state variable.

1.4.8

```
DOMString CurrentVal
```

The current value of the variable.

The current value of the variable. This needs to be allocated by the caller. When finished with it, the SDK frees this **DOMString**.

1.5

```
struct Upnp_State_Var_Complete
```

Represents the reply for the current value of a state variable in an asynchronous call.

Members

1.5.1	int	ErrCode	<i>The result of the operation.</i>	9
1.5.2	char	CtrlUrl [NAME_SIZE]	<i>The control URL for the service.</i>	9
1.5.3	char	StateVarName [NAME_SIZE]	<i>The name of the variable.</i>	9
1.5.4	DOMString	CurrentVal	<i>The current value of the variable or error string in case of error.</i>	10

Represents the reply for the current value of a state variable in an asynchronous call.

1.5.1

```
int ErrCode
```

The result of the operation.

The result of the operation.

1.5.2

```
char CtrlUrl [NAME_SIZE]
```

The control URL for the service.

The control URL for the service.

1.5.3

```
char StateVarName [NAME_SIZE]
```

The name of the variable.

The name of the variable.

1.5.4

DOMString **CurrentVal**

The current value of the variable or error string in case of error.

The current value of the variable or error string in case of error.

1.6

struct **Upnp_Event**

*Returned along with a **UPNP_EVENT_RECEIVED** callback.*

Members

1.6.1	Upnp_SID	Sid	<i>The subscription ID for this subscription.</i>	10
1.6.2	int	EventKey	<i>The event sequence number.</i>	10
1.6.3	IXML_Document*	ChangedVariables	<i>The DOM tree representing the changes generating the event.</i>	11

Returned along with a **UPNP_EVENT_RECEIVED** callback.

1.6.1

Upnp_SID **Sid**

The subscription ID for this subscription.

The subscription ID for this subscription.

1.6.2

int **EventKey**

The event sequence number.

The event sequence number.

1.6.3

IXML_Document* **ChangedVariables**

The DOM tree representing the changes generating the event.

The DOM tree representing the changes generating the event.

1.7

struct **Upnp_Discovery**

Returned in a UPNP_DISCOVERY_RESULT callback.

Members

1.7.1	int	ErrCode	<i>The result code of the UpnpSearchAsync call.</i>	12
1.7.2	int	Expires	<i>The expiration time of the advertisement.</i>	12
1.7.3	char	DeviceId [LINE_SIZE]	<i>The unique device identifier.</i>	12
1.7.4	char	DeviceType [LINE_SIZE]	<i>The device type.</i>	12
1.7.5	char	ServiceType [LINE_SIZE]	<i>The service type.</i>	12
1.7.6	char	ServiceVer [LINE_SIZE]	<i>The service version.</i>	13
1.7.7	char	Location [LINE_SIZE]	<i>The URL to the UPnP description document for the device.</i>	13
1.7.8	char	Os [LINE_SIZE]	<i>The operating system the device is running.</i>	13
1.7.9	char	Date [LINE_SIZE]	<i>Date when the response was generated. .</i>	13
1.7.10	char	Ext [LINE_SIZE]	<i>Confirmation that the MAN header was understood by the device.</i>	13
1.7.11	struct sockaddr_storage	DestAddr	<i>The host address of the device responding to the search.</i>	14

Returned in a UPNP_DISCOVERY_RESULT callback.

1.7.1

`int ErrCode`

*The result code of the **UpnpSearchAsync** call.*

The result code of the **UpnpSearchAsync** call.

1.7.2

`int Expires`

The expiration time of the advertisement.

The expiration time of the advertisement.

1.7.3

`char DeviceId [LINE_SIZE]`

The unique device identifier.

The unique device identifier.

1.7.4

`char DeviceType [LINE_SIZE]`

The device type.

The device type.

1.7.5

`char ServiceType [LINE_SIZE]`

The service type.

The service type.

1.7.6

`char ServiceVer [LINE_SIZE]`

The service version.

The service version.

1.7.7

`char Location [LINE_SIZE]`

The URL to the UPnP description document for the device.

The URL to the UPnP description document for the device.

1.7.8

`char Os [LINE_SIZE]`

The operating system the device is running.

The operating system the device is running.

1.7.9

`char Date [LINE_SIZE]`

Date when the response was generated.

Date when the response was generated.

1.7.10

`char Ext [LINE_SIZE]`

Confirmation that the MAN header was understood by the device.

Confirmation that the MAN header was understood by the device.

1.7.11

```
struct sockaddr_storage DestAddr
```

The host address of the device responding to the search.

The host address of the device responding to the search.

1.8

```
struct Upnp_Event_Subscribe
```

*Returned along with a **UPNP_EVENT_SUBSCRIBE_COMPLETE** or **UPNP_EVENT_UNSUBSCRIBE_COMPLETE** callback.*

Members

1.8.1	Upnp_SID	Sid	<i>The SID for this subscription.</i>	14
1.8.2	int	ErrCode	<i>The result of the operation.</i>	15
1.8.3	char	PublisherUrl [NAME.SIZE]	<i>The event URL being subscribed to or removed from.</i>	15
1.8.4	int	TimeOut	<i>The actual subscription time (for subscriptions only).</i>	15

Returned along with a **UPNP_EVENT_SUBSCRIBE_COMPLETE** or **UPNP_EVENT_UNSUBSCRIBE_COMPLETE** callback.

1.8.1

```
Upnp_SID Sid
```

The SID for this subscription.

The SID for this subscription. For subscriptions, this only contains a valid SID if the **Upnp_EventSubscribe.result** field contains a **UPNP_E_SUCCESS** result code. For unsubscriptions, this contains the SID from which the subscription is being unsubscribed.

1.8.2

```
int ErrCode
```

The result of the operation.

The result of the operation.

1.8.3

```
char PublisherUrl [NAME_SIZE]
```

The event URL being subscribed to or removed from.

The event URL being subscribed to or removed from.

1.8.4

```
int TimeOut
```

The actual subscription time (for subscriptions only).

The actual subscription time (for subscriptions only).

1.9

```
struct Upnp_Subscription_Request
```

*Returned along with a **UPNP_EVENT_SUBSCRIPTION_REQUEST** callback.*

Members

1.9.1	char*	ServiceId	<i>The identifier for the service being subscribed to.</i>	16
1.9.2	char*	UDN	<i>Universal device name.</i>	16
1.9.3	Upnp_SID	Sid	<i>The assigned subscription ID for this subscription.</i>	16

Returned along with a **UPNP_EVENT_SUBSCRIPTION_REQUEST** callback.

1.9.1

`char* ServiceId`

The identifier for the service being subscribed to.

The identifier for the service being subscribed to.

1.9.2

`char* UDN`

Universal device name.

Universal device name.

1.9.3

`Upnp_SID Sid`

The assigned subscription ID for this subscription.

The assigned subscription ID for this subscription.

2

```
EXPORT_SPEC int UpnpAddToPropertySet (    IXML_Document**  
                                           PropSet,      const  
                                           charArgName,,  contt  
                                           char* rgVal))
```