

---

## **Layer3Forwarding:1 Service Template Version 1.01**

**For UPnP™ Version 1.0**

**Status: Standardized DCP**

**Date: November 12, 2001**

---

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.

© 1999-2001 Contributing Members of the UPnP™ Forum. All Rights Reserved.

<b>Authors</b>	<b>Company</b>
Prakash Iyer, Ulhas Warriar	Intel Corporation
Matthew Schmitz	Cisco Systems

## Contents

<b>1. OVERVIEW AND SCOPE .....</b>	<b>3</b>
1.1. CHANGE LOG .....	3
<b>2. SERVICE MODELING DEFINITIONS .....</b>	<b>5</b>
2.1. SERVICE TYPE .....	5
2.2. STATE VARIABLES .....	5
2.2.1. <i>DefaultConnectionService</i> .....	5
2.2.2. <i>Relationships Between State Variables</i> .....	6
2.3. EVENTING AND MODERATION .....	6
2.3.1. <i>Event Model</i> .....	6
2.4. ACTIONS .....	6
2.4.1. <i>SetDefaultConnectionService</i> .....	6
2.4.2. <i>GetDefaultConnectionService</i> .....	7
2.4.3. <i>Non-Standard Actions Implemented by a UPnP Vendor</i> .....	8
2.4.4. <i>Relationships Between Actions</i> .....	8
2.4.5. <i>Common Error Codes</i> .....	8
2.5. THEORY OF OPERATION .....	8
<b>3. XML SERVICE DESCRIPTION .....</b>	<b>10</b>
<b>4. TEST .....</b>	<b>11</b>

## List of Tables

Table 1: State Variables .....	5
Table 2: Event Moderation .....	6
Table 3: Actions .....	6
Table 4: Arguments for <i>SetDefaultConnectionService</i> .....	7
Table 5: Arguments for <i>GetDefaultConnectionService</i> .....	7
Table 6: Common Error Codes .....	8

# 1. Overview and Scope

This service definition is compliant with the UPnP Device Architecture version *1.0*.

This OPTIONAL service models all layer-3 packet forwarding services applicable across all connection service instances of **urn:schemas-upnp-org:device:WANDevice**

Generic Layer 3 packet forwarding services include all IP / layer-3 routing services. However, any such services that are configurable on a per connection instance basis, will be modeled in the context of the connection service itself. Network Address Translation (NAT) and Realm Specific IP (RSIP) are examples of such services.

The service is defined to be included under the root device **urn:schemas-upnp-org:device:InternetGatewayDevice**

## 1.1. Change Log

Changes from *Layer3Forwarding:0.1*

- Updated error return codes
- Minor corrections to XML template

Changes from *Layer3Forwarding:0.2*

- Removed redundant parameter '7tuple' from AddInboundServiceMapping and DeleteInboundServiceMapping actions.
- Added 'Get' actions per Technical Committee recommendation to not use QueryStateVariable for reading state variables.

Changes from *Layer3Forwarding:0.3* (as per WC meeting on 9/6/00)

- Changed boolean values from false, true to 0,1.
- Added state variable DefaultConnection, along with Get and Set actions
- Removed Firewall related variables and actions as they do not provide adequate information
- Removed 'step' tag in XML description for Boolean variable.

Changes from *Layer3Forwarding:0.4* (as per WC meeting on 10/17/00)

- Moved all RSIP and NAT related SST variables and actions to connection service templates

Changes from *Layer3Forwarding:0.5*

- Removed "urn" from device specification for DefaultConnection
- Removed white spaces from XML template
- Changed default value for empty strings to tags with no element values

Changes from *Layer3Forwarding:0.51*

- Text clarifications in different sections
- Changed document status to template design complete

Changes from *Layer3Forwarding:0.52*

- Updated to service template v1.01
- Verified against TDC checklist v1.01
- Added error code 723

Changes from *Layer3Forwarding:0.8*

- Added XML comment tags to comments text in XML template

Changes from *Layer3Forwarding:0.81*

- Added semantic tests section.
- Textual clarifications in action descriptions and theory of operation.

Changes from *Layer3Forwarding:0.9*

- Version changes for 1.0.
- Added an example to illustrate a default connection service entry

Changes from *Layer3Forwarding:0.99*

- Version updated to reflect 45-day review completion. No other changes to this draft.

Changes from *Layer3Forwarding:0.991*

- Copyright messages and document status updated.

## 2. Service Modeling Definitions

### 2.1. ServiceType

The following service type identifies a service that is compliant with this template:

urn:schemas-upnp-org:service:[Layer3Forwarding:1](#).

### 2.2. State Variables

Table 1: State Variables

Variable Name	Req. or Opt. <sup>1</sup>	Data Type	Allowed Value <sup>2</sup>	Default Value <sup>2</sup>	Eng. Units
DefaultConnectionService	R	string	Undefined	Empty string	N/A
<i>Non-standard state variables implemented by an UPnP vendor go here.</i>	<i>X</i>	<i>TBD</i>	<i>TBD</i>	<i>TBD</i>	<i>TBD</i>

<sup>1</sup> R = Required, O = Optional, X = Non-standard.

<sup>2</sup> Values listed in this column are required. To specify standard optional values or to delegate assignment of values to the vendor, you must reference a specific instance of an appropriate table below.

#### 2.2.1. DefaultConnectionService

This variable specifies a connection service instance in a **WANConnectionDevice**, one or more of which could be defined in a **WANDevice**. A comma-separated 2-tuple uniquely identifies the service:

uuid:device-UUID:WANConnectionDevice:y ,

urn:upnp-org:serviceId:serviceID

An example of a 2-tuple string follows:

“uuid:44f5824f- c57d-418c-a131-f22b34e14111:WANConnectionDevice:1,urn:upnp-org:serviceId:WANPPPConn1”

This is useful in the case of a dial-on-demand scenario where a client attempts to access Internet without explicitly initiating a connection (as would be the case for a legacy or non-UPnP networked device on the residential LAN). If multiple connections are configured, and a default connection is not set, the gateway selects a connection instance in an implementation dependent manner. **Only one instance of WAN\*Connection service per InternetGatewayDevice (across multiple WANDevice and WANConnectionDevice instances) can be set as the default connection.** If a default connection is not set, and only one WAN\*Connection instance is configured, it is automatically assumed to be the default connection. If the connection that is set as a default connection is not configured correctly on the gateway,

then a dial-on-demand / implicit connection initiation attempt will fail.. If no connections are configured on the gateway and a client initiates an implicit dial-on-demand, the attempt to access the Internet will fail.

**NOTE:** In the case of bridged, relayed or spoofed connections, there is an implicit one-to-one correspondence between a control point and an active connection. In essence, the gateway performs source address based routing. Sharing of connections is not possible in these scenarios. Consequently, if the `DefaultConnectionService` points to an **active** connection service instance of one of the types listed above, dial-on-demand for a legacy or non-UPnP client will not work.

## 2.2.2. Relationships Between State Variables

None.

## 2.3. Eventing and Moderation

Table 2: Event Moderation

Variable Name	Evented	Moderated Event	Max Event Rate <sup>1</sup>	Logical Combination	Min Delta per Event <sup>2</sup>
<code>DefaultConnectionService</code>	Yes	No	N/A	N/A	N/A
<i>Non-standard state variables implemented by an UPnP vendor go here.</i>	<i>TBD</i>	<i>TBD</i>	<i>TBD</i>	<i>TBD</i>	<i>TBD</i>

<sup>1</sup> Determined by N, where Rate = (Event)/(N secs).

<sup>2</sup> (N) \* (allowedValueRange Step).

### 2.3.1. Event Model

`DefaultConnectionService` is evented whenever its value changes. Eventing is not moderated.

## 2.4. Actions

Immediately following this table is detailed information about these actions, including short descriptions of the actions, the effects of the actions on state variables, and error codes defined by the actions.

Table 3: Actions

Name	Req. or Opt. <sup>1</sup>
<code>SetDefaultConnectionService</code>	<u>R</u>
<code>GetDefaultConnectionService</code>	<u>R</u>
<i>Non-standard actions implemented by an UPnP vendor go here.</i>	X

<sup>1</sup> R = Required, O = Optional, X = Non-standard.

### 2.4.1. SetDefaultConnectionService

Action selects a connection service instance on one of the **WANConnectionDevice** instances to be the default connection. If this is not specified, the selection of the default gateway is implementation

dependent. If the connection specified is not configured, dial-on-demand connection attempts to the Internet will fail.

#### 2.4.1.1. Arguments

**Table 4: Arguments for SetDefaultConnectionService**

Argument	Direction	relatedStateVariable
NewDefaultConnectionService	<i>IN</i>	DefaultConnectionService

#### 2.4.1.2. Dependency on State (if any)

#### 2.4.1.3. Effect on State (if any)

Sets / changes the default connection for the gateway.

#### 2.4.1.4. Errors

errorCode	errorDescription	Description
402	Invalid Args	One of following: not enough IN arguments, too many IN arguments, no IN argument by that name, one or more IN arguments are of the wrong data type. See also the UPnP™ Device Architecture.
501	Action Failed	May be returned in current state if service prevents invoking of that action.
720	InvalidDeviceUUID	The UUID of a device specified in the action arguments is invalid
721	InvalidServiceID	The Service ID of a service specified in the action arguments is invalid
723	InvalidConnServiceSelection	The selected connection service instance cannot be set as a default connection

## 2.4.2. GetDefaultConnectionService

This action retrieves the 2-tuple string representing the default connection service in the gateway.

#### 2.4.2.1. Arguments

**Table 5: Arguments for GetDefaultConnectionService**

Argument	Direction	relatedStateVariable
NewDefaultConnectionService	<i>OUT</i>	DefaultConnectionService

#### 2.4.2.2. Dependency on State (if any)

#### 2.4.2.3. Effect on State

None.

#### 2.4.2.4. Errors

errorCode	errorDescription	Description
402	Invalid Args	One of following: not enough IN arguments, too many IN arguments, no IN argument by that name, one or more IN arguments are of the wrong data type. See also the UPnP™ Device Architecture.
501	Action Failed	May be returned in current state if service prevents invoking of that action.

#### 2.4.3. Non-Standard Actions Implemented by a UPnP Vendor

To facilitate certification, non-standard actions implemented by UPnP vendors should be included in this service template. The UPnP Device Architecture lists naming requirements for non-standard actions (see the section on Description).

#### 2.4.4. Relationships Between Actions

#### 2.4.5. Common Error Codes

The following table 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 6: Common Error Codes**

errorCode	errorDescription	Description
401	Invalid Action	See UPnP Device Architecture section on Control.
402	Invalid Args	See UPnP Device Architecture section on Control.
404	Invalid Var	See UPnP Device Architecture section on Control.
501	Action Failed	See UPnP Device Architecture section on Control.
600-699	TBD	Common action errors. Defined by UPnP Forum Technical Committee.
701-799		Common action errors defined by the UPnP Forum working committees.
<i>800-899</i>	<i>TBD</i>	<i>(Specified by UPnP vendor.)</i>

## 2.5. Theory of Operation

A gateway could perform sophisticated routing functions, especially if it has multiple LAN and WAN interfaces. However, the IGD Working Committee decided to restrict the scope of the first version of gateway DCP to simpler usage scenarios. NAT functionality is configurable on a per connection basis and is consequently modeled in the appropriate connection service.

Per the current specification, the following precedence is implied in determining routes to a destination IP address on the WAN:

1. If a control point successfully invokes `RequestConnection`<sup>3</sup> on a connection service instance, subsequent IP packets from that control point **MUST** be routed on that connection instance.

<sup>3</sup> Refer to the *WAN\*\*Connection* service definition for more information on `RequestConnection`.



2. If `RequestConnection` is not called, and if `DefaultConnectionService` points to a usable connection service instance, packets from a node on the residential LAN **MUST** be routed on that connection instance.
3. If `RequestConnection` is not called, and if `DefaultConnectionService` is not specified, packets **MUST** be routed based on the reachable route for the destination IP address in the packet..

### 3. 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>SetDefaultConnectionService</name>
      <argumentList>
        <argument>
          <name>NewDefaultConnectionService</name>
          <direction>in</direction>
          <relatedStateVariable>DefaultConnectionService</relatedStateVariable>
        </argument>
      </argumentList>
    </action>
    <action>
      <name>GetDefaultConnectionService</name>
      <argumentList>
        <argument>
          <name>NewDefaultConnectionService</name>
          <direction>out</direction>
          <relatedStateVariable>DefaultConnectionService</relatedStateVariable>
        </argument>
      </argumentList>
    </action>
    <!-- Declarations for other actions added by UPnP vendor (if any) go
here -->
  </actionList>
  <serviceStateTable>
    <stateVariable sendEvents="yes">
      <name>DefaultConnectionService</name>
      <dataType>string</dataType>
    </stateVariable>
    <!-- Declarations for other state variables added by UPnP vendor (if
any) go here -->
  </serviceStateTable>
</scpd>

```

## 4. Test

### SetDefaultConnectionService / GetDefaultConnectionService

Test Sequence 1: To test success path

Semantic class: 2

Pre-conditions: None.

SetDefaultConnectionService

Success = 200

In-Arg	Values	State Variables	Current State	Expected State
DefaultConnectionService	A string representing a valid connection service instance per semantics defined in the service description document			
Out-Arg	Expected Value			
		Error Code (if any)	NA	NA

GetDefaultConnectionService

Success = 200

In-Arg	Values	State Variables	Current State	Expected State
Out-Arg	Expected Value			
DefaultConnectionService	String set in the preceding Set action	Error Code (if any)	NA	NA

Test Sequence 2: To test error 720

Semantic class: 2

Pre-conditions: None.

GetDefaultConnectionService

Success = 200

In-Arg	Values	State Variables	Current State	Expected State

<b>Out-Arg</b>	<b>Expected Value</b>			
DefaultConnectionService	A string representing a valid connection service instance per semantics defined in the service description document OR an empty string representing an uninitialized default connection	<b>Error Code (if any)</b>	NA	NA

SetDefaultConnectionService

Success = 200

In-Arg	Values	State Variables	Current State	Expected State
DefaultConnectionService	A string representing a connection service instance per semantics defined in the service description document but with an invalid value for device container			
<b>Out-Arg</b>	<b>Expected Value</b>			
		<b>Error Code (if any)</b>	720	NA

GetDefaultConnectionService

Success = 200

In-Arg	Values	State Variables	Current State	Expected State
<b>Out-Arg</b>	<b>Expected Value</b>			
DefaultConnectionService	String retrieved in preceding Get action	<b>Error Code (if any)</b>	NA	NA

Test Sequence 3: To test error 721

Semantic class: 2

Pre-conditions: None.

GetDefaultConnectionService

Success = 200

In-Arg	Values	State Variables	Current State	Expected State
Out-Arg	Expected Value			
DefaultConnectionService	A string representing a valid connection service instance per semantics defined in the service description document OR an empty string representing an uninitialized default connection	Error Code (if any)	NA	NA

SetDefaultConnectionService

Success = 200

In-Arg	Values	State Variables	Current State	Expected State
DefaultConnectionService	A string representing a connection service instance per semantics defined in the service description document but with an invalid value for service instance			
Out-Arg	Expected Value			
		Error Code (if any)	721	NA

GetDefaultConnectionService

Success = 200

In-Arg	Values	State Variables	Current State	Expected State
Out-Arg	Expected Value			
DefaultConnectionService	String retrieved in preceding Get action	Error Code (if any)	NA	NA

**Change History****Change Log for Version 1.0 (10-4-00)**

- Revised the Title Page to call out V1.0 of the Service Template
- Changed to be consistent with Sample Designs released to the Technical Committee
- Service State Table: Variable Descriptions removed from the table and are listed in specific sections following the table.
- Actions: Reformatted the information contained in the Action Table:
  - Added overview entry point.
  - Added an Action Summary Table to specify Required or Optional
  - Added enumerated sections to specify each actions: Arguments, Effect on State, and Errors.