TemperatureSensor:1 Service Template

For UPnP™ Device Architecture V 1.0

Status: Standardized DCP
Date: May 13th, 2003

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 Membership Agreement. UPnP Forum Members have rights and licenses defined by Section 3 of the UPnP 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 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.

Authors	Company
Larry Stickler	Honeywell
Pete Bergstrom	Honeywell
Andrew Fiddian-Green	Siemens Building Technologies

Contents

1. OVERVIEW AND SCOPE 3

1.1. CHANGE LOG FOR: TEMPERATURESENSOR:1 3

2. SERVICE MODELING DEFINITIONS 4

2.1. SERVICETYPE 4
2.2. STATE VARIABLES 4
2.2.1. Application 5
2.2.2. CurrentTemperature 5
2.2.3. Name 5
2.2.4. Relationships Between State Variables 5
2.3. EVENTING AND MODERATION 5
2.3.1. Event Model 5
2.4. ACTIONS 6
2.4.1. SetApplication 6
2.4.2. GetApplication 7
2.4.3. GetCurrentTemperature 7
2.4.4. GetName 8
2.4.5. SetName 8
2.4.6. Non-Standard Actions Implemented by a UPnP Vendor 9
2.4.7. Relationships Between Actions 9
2.4.8. Common Error Codes 9
2.5. THEORY OF OPERATION 9

3. XML SERVICE DESCRIPTION 11

4. TEST 13

List of Tables

Table 1 State Variables 4

Table 2 AllowedValueList for Application 4

Table 3 AllowedValueRange for CurrentTemperature 5

Table 4 Eventing & Moderation 5

Table 5 Event Model 5

Table 6 Action list 6

Table 7 Arguments for SetApplication 6

Table 8 Arguments for GetApplication 7

Table 9 Arguments for GetCurrentTemperature 7

Table 10 Arguments for GetName 8

Table 11 Arguments for SetName 8

1. Overview and Scope

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

This service type enables the following functions:

- Reading of the current temperature of a temperature sensor
- Setting and reading of the intended application for this temperature sensor
- Setting and reading of the user name for this sensor

1.1. Change Log for: TemperatureSensor:1

7/26	Changes per 7/17 meeting of Home Automation and Security Working Group and conversion to 0.996 template.
8/24/00	Clean-up
8/28	Added XML, removed HighestValid and LowestValid
9/28/00	Changed units to Celsius, moved reserved application values to data type column, corrected XML
10/18/00	Changed event moderation
11/28/00	Moved to Template Design Complete, added min and max allowed values for Current temp, expanded Theory of operation.
2/14/01	Updated for Template Design Complete – Evented Applications, corrected temperature specification, corrected capitalization, improved description.
2/21/01	Moved to Template 1.1, cleaned-up XML
2/26/01	Proof read
4/2/01	Added the ability to write the Application variable.
5/31/02	Revision marks deleted; Moved to 0.9; Test chapter added.
[13 May 2003] v1.0	Converted to Approved Standard.

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:TemperatureSensor:1

2.2. State Variables

Table 1 State Variables

Variable Name	Req. or Opt.[1]	Data Type	Allowed Value [2]	Default Value [2]	Eng. Units
Application	R	string	see table	(none)	n/a
CurrentTemperature	R	i4	see table	(none)	.01 degrees Celsius
Name	O	string		Zero length string	N/a
Non-standard state variables implemented by an UPnP vendor go here.	*X*	*TBD*	*TBD*	*TBD*	*TBD*

[1] R = Required, O = Optional, X = Non-standard.

[2]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.

Table 2 AllowedValueList for Application

Value	Req. or Opt. [1]
Vendor defined as "none" *R/W -This allows a control point to establish the application type*	*O*
Vendor-defined *One value required. Reserved names are:* *Room,* *Outdoor,* *Pipe,* *AirDuct,*	*0*
Vendor-defined	*O*

[1] R = Required, O = Optional, X = Non-standard.

Table 3 AllowedValueRange for CurrentTemperature

	Value	Req. or Opt. [1]
minimum	*Vendor-defined*	R
maximum	*Vendor-defined*	R
step	*Step=1 (i.e. 0.01 Celsius)*	R

[1] R = Required, O = Optional, X = Non-standard.

2.2.1. Application

This variable states the intended application of this service.

2.2.2. CurrentTemperature

This variable exposes the setpoint of a service that is controlling temperature to that setpoint.

2.2.3. Name

This optional variable may be used to capture a friendly name or location for this sensor.

2.2.4. Relationships Between State Variables

None

2.3. Eventing and Moderation

Table 4 Eventing & Moderation

Variable Name	Evented	Moderated Event	Max Event Rate[1]	Logical Combination	Min Delta per Event[2]
Name	Yes	No	none	none	On-change
Application	Yes	No	none	none	On-change
CurrentTemperature	Yes	Yes	10 sec		Per 0.2 degree Celsius change or 20 units
Non-standard state variables implemented by an UPnP vendor go here.	*TBD*	*TBD*	*TBD*	*TBD*	*TBD*

[1] Determined by N, where Rate = (Event)/(N secs).
[2] (N) * (allowedValueRange Step).

2.3.1. Event Model

Table 5 Event Model

Variable Name	UI requirements	Async Requirements	Func. Vs max rate tradeoffs	Est of Max rate	Reason not evented
Application	Needed for UI			Very Low	N/a
CurrentTemperature	Needed for UI			Very low	N/a
Name	Needed for UI			Very low	N/a

2.4. Actions

Table 6 Action list

Name	Req. or Opt. [1]
SetApplication	*O*
GetApplication	*R*
GetCurrentTemperature	*R*
GetName	O
SetName	O
Non-standard actions implemented by an UPnP vendor go here.	X

[1] R = Required, O = Optional, X = Non-standard.

2.4.1. SetApplication

Provides the Application value to a control point or other devices

2.4.1.1. Arguments

Table 7 Arguments for SetApplication

Argument	Direction	relatedStateVariable
NewApplication	*IN*	Application

2.4.1.2. Dependency on State (if any)

None

2.4.1.3. Effect on State (if any)

Changes the Application.

2.4.1.4. Errors

errorCode	errorDescription	Description
none		

2.4.2. GetApplication

Provides the Application value to a control point or other devices

2.4.2.1. Arguments

Table 8 Arguments for GetApplication

Argument	Direction	relatedStateVariable
CurrentApplication	*Out*[R]	Application

[R] Return Value

2.4.2.2. Dependency on State (if any)

Depends on Application

2.4.2.3. Effect on State (if any)

None

2.4.2.4. Errors

errorCode	errorDescription	Description
none		

2.4.3. GetCurrentTemperature

2.4.3.1. Arguments

Table 9 Arguments for GetCurrentTemperature

Argument	Direction	relatedStateVariable
CurrentTemp	*Out*[R]	CurrentTemperature

[R] Return Value

2.4.3.2. Dependency on State (if any)

Depends on the temperature.

2.4.3.3. Effect on State

None

2.4.3.4. Errors

errorCode	errorDescription	Description
none		

2.4.4. GetName

Provides the Name value to a control point or other UPnP device

2.4.4.1. Arguments

Table 10 Arguments for GetName

Table 5: Arguments for GetApplication(CurrentApplication)

Argument	Direction	relatedStateVariable
CurrentName	*Out*[R]	Name

[R] Return Value

2.4.4.2. Dependency on State (if any)

None

2.4.4.3. Effect on State

None

2.4.4.4. Errors

errorCode	errorDescription	Description
none		

2.4.5. SetName

Provides a new Name value for the Name variable.

2.4.5.1. Arguments

Table 11 Arguments for SetName

Argument	Direction	relatedStateVariable
NewName	*In*	Name

2.4.5.2. Dependency on State (if any)

None

2.4.5.3. Effect on State

Changes Name.

2.4.5.4. Errors

errorCode	errorDescription	Description
none		

2.4.6. 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).

2.4.7. Relationships Between Actions

None.

2.4.8. 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.
800-899	*TBD*	*(Specified by UPnP vendor.)*

2.5. Theory of Operation

This service allows a temperature read from a temperature sensor.

Control points or other devices may set and get an application value for this service. The following applications are defined:

- Room – Indoor room temperature
- Outdoor – Outdoor temperature

- AirDuct – Temperature inside an air duct
- Pipe – surface temperature of a pipe.

Manufacturers shall establish the allowable range of temperatures using the maximum and minimum allowed values. A Control Point or other device can find these values in the XML description.

Control points or other devices may optionally establish a Name for this sensor.

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>GetApplication</name>
      <argumentList>
        <argument>
          <name>CurrentApplicationname>
          <direction>out</direction>
          <retval />
          <relatedStateVariable>Application</relatedStateVariable>
        </argument>
      </argumentList>
    </action>
  The following action is optional
<action>
    <name>SetApplication</name>
      <argumentList>
        <argument>
          <name>NewApplicationname>
          <direction>in</direction>
          <relatedStateVariable>Application</relatedStateVariable>
        </argument>
      </argumentList>
    </action>

<action>
    <name>GetCurrentTemperature</name>
      <argumentList>
        <argument>
          <name>CurrentTemp</name>
          <direction>out</direction>
          <retval />

<relatedStateVariable>CurrentTemperature</relatedStateVariable>
        </argument>
      </argumentList>
    </action>

The following action is optional
<action>
    <name>GetName</name>
      <argumentList>
        <argument>
          <name>CurrentName</name>
          <direction>out</direction>
          <retval />
          <relatedStateVariable>Name</relatedStateVariable>
        </argument>
```

```
      </argumentList>
    </action>
      The following action is optional
    <action>
    <name>SetName</name>
      <argumentList>
        <argument>
          <name>NewName</name>
          <direction>in</direction>
         <relatedStateVariable>Name</relatedStateVariable>
        </argument>
      </argumentList>
    </action>
    Declarations for other actions added by UPnP vendor (if any) go here
  </actionList>

  <serviceStateTable>
    <stateVariable sendEvents="yes">
      <name>Application</name>
      <dataType>string</dataType>
      <allowedValueList>
        <allowedValue> vender defined </allowedValue>
        Other allowed values defined by UPnP Forum working committee (if
        any) go here
      </allowedValueList>
    </stateVariable>

    <stateVariable sendEvents="yes">
      <name>CurrentTemperature</name>
      <dataType>i4</dataType>
      <allowedValueRange>
        <minimum>manufacturer defined</minimum>
        <maximum>manufacturer defined</maximum>
         <step>1</step>
       </allowedValueRange>
    </stateVariable>

  The following state variable is optional
      <stateVariable sendEvents="yes">
      <name>Name</name>
      <dataType>string</dataType>

    </stateVariable>

    Declarations for other state variables defined by UPnP Forum working
    committee(if any) go here
    Declarations for other state variables added by UPnP vendor (if any)
    go here
  </serviceStateTable>
</scpd>
```

4. Test

Testing of the UPnP functions Addressing, Discovery, Description, Control (Syntax) and Eventing are performed by the UPnP Test Tool v1.1 based on the following documents:

- UPnP Device Architecture v1.0
- The Service Definitions in chapter 2 of this document
- The XML Service Description in chapter 3 of this document
- The UPnP Test Tool service template test file: ***TemperatureSensor1.xml***
- The UPnP Test Tool service template test file: ***TemperatureSensor1.SyntaxTests.xml***

The test suite does not include tests for Control Semantics, since it is felt that such tests would not provide a higher level of interoperability.