

# ChangeDisc:1 Sample Service Template

For UPnP™ Version 1.0

**Status: Preliminary Design (TPD)**

**Date: 23 August 2000**

---

This sample service template is being made available to UPnP Members pursuant to Section 2.1(c)(ii) of the UPnP Membership Agreement for review and comment by Members to the UPnP Steering Committee regarding the Steering Committee's consideration of the Proposed template as a Standardized service. Pursuant to Section 3.1 of the UPnP Membership Agreement, Member has limited rights to use or reproduce the Proposed template during the comment period and only in furtherance of this review and comment. 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 PROPOSED TEMPLATES, IMPLEMENTATIONS OR IN ANY ASSOCIATED TEST SUITES. THE SAMPLE SERVICE TEMPLATE IS PROVIDED "AS IS" AND "WITH ALL FAULTS". THE UPNP FORUM MAKES NO WARRANTIES, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE PROPOSED SERVICE TEMPLATE 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.

© 2000 Contributing Members of the UPnP™ Forum. All Rights Reserved.

Author	Company
Jeffrey Schlimmer	Microsoft

## Contents

<b>1. OVERVIEW AND SCOPE.....</b>	<b>3</b>
1.1. CHANGE LOG.....	4
<b>2. SERVICE MODELING DEFINITIONS.....</b>	<b>5</b>
2.1. SERVICE TYPE.....	5
2.2. STATE VARIABLES.....	5
2.2.1. <i>TrayHasDisc</i> .....	5
2.2.2. <i>DoorIsOpen</i> .....	5
2.2.3. <i>Non-Standard State Variables Implemented by an UPnP Vendor</i> .....	5
2.2.4. <i>Relationships Between State Variables</i> .....	5
2.3. EVENTING AND MODERATION.....	6
2.3.1. <i>Event Model</i> .....	6
2.4. ACTIONS.....	6
2.4.1. <i>AddDisc</i> .....	7
2.4.2. <i>NextDisc</i> .....	7
2.4.3. <i>PrevDisc</i> .....	8
2.4.4. <i>RandomDisc</i> .....	8
2.4.5. <i>HasTrayDisc</i> .....	8
2.4.6. <i>OpenDoor</i> .....	9
2.4.7. <i>CloseDoor</i> .....	9
2.4.8. <i>ToggleDoor</i> .....	10
2.4.9. <i>IsDoorOpen</i> .....	10
2.4.10. <i>Non-Standard Actions Implemented by an UPnP Vendor</i> .....	11
2.4.11. <i>Relationships Between Actions</i> .....	11
2.4.12. <i>Common Error Codes</i> .....	11
2.5. THEORY OF OPERATION.....	11
<b>3. XML SERVICE DESCRIPTION.....</b>	<b>13</b>
<b>4. TEST.....</b>	<b>15</b>

## List of Tables

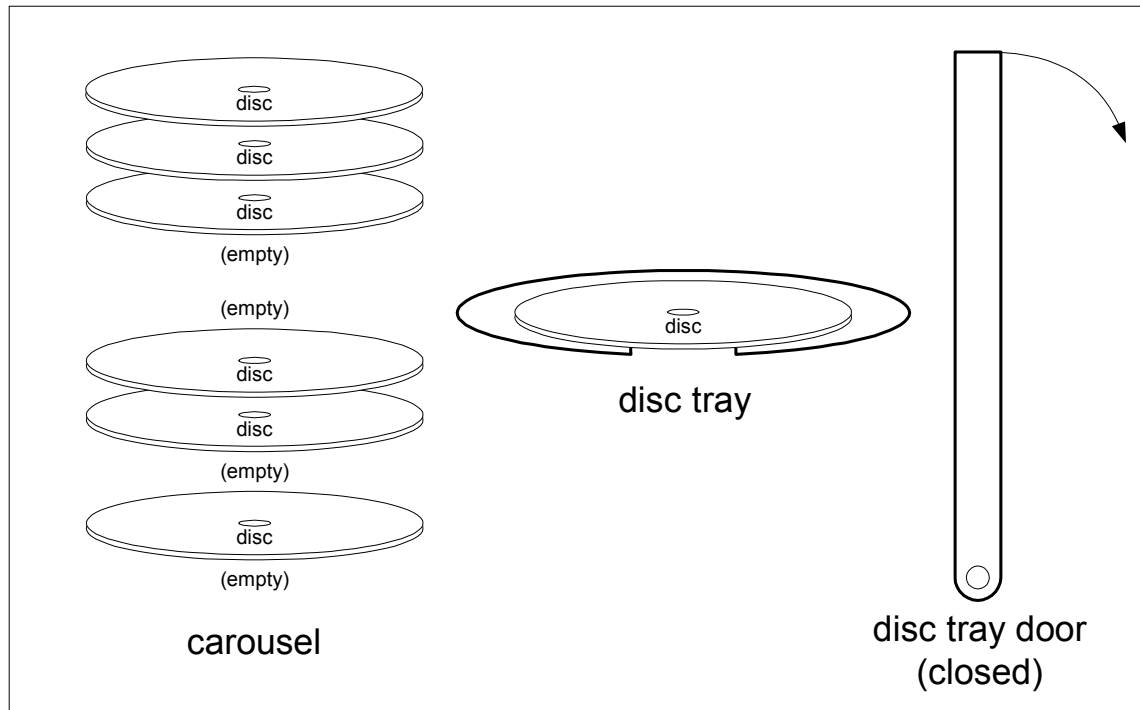
Table 1: State variables .....	5
Table 2: Event moderation .....	6
Table 3: Actions .....	6

## 1. Overview and Scope

This sample service template is compliant with the UPnP Device Architecture version 1.0. It defines a service type referred to herein as ChangeDisc:1.

ChangeDisc:1 handles shuffling CD or DVD discs between a carousel and the play mechanism. ChangeDisc:1 has a door through which the user can add discs into, and remove discs from, the changer.

To be more specific, the diagram below depicts the overall components of ChangeDisc:1



- A carousel (or magazine or cartridge) within a changer that holds a number of discs. The number may be fixed but must be at least one.
- A tray that holds at most one disc. The tray is the transport interface between the carousel and the play mechanism.
- A door that may be opened to remove the disc in the tray or to add a disc if the tray is empty.

To play a specific disc in the carousel, that disc is moved into the tray and is ready to play. (Actually playing the disc is done using another service type described elsewhere.)

When a disc is moved into the tray, the disc tray door closes (if open), the disc currently in the tray (if any) is moved back into the carousel, and the new disc is moved into the tray.

If the user opens the disc tray door, they can remove the disc in the tray (if any) or place a new disc into the tray (if empty).

Using this model, ChangeDisc:1 enables the following functions for disc changers:

- Adding (or removing) a disc into (or from) the carousel.
- Selecting the next, previous, or random disc in the carousel to be played.

ChangeDisc:1 does not enable:

- Adding discs into a changer while another disc is playing (i.e., play/exchange).
- Handling dual-sided disc media.

*This is only a sample. It does not replace the work of an UPnP Forum working committee. It is only intended to illustrate the use of device and service templates.*

## 1.1. Change Log

- [8 Aug 2000] Fleshed out v0.02. Renamed from AVDiskSel.
- [9 Aug 2000] v0.03. Added Relationships Between State Variables, Relationships Between Actions, and Event Model subsections. Cleaned up table listing actions and moved descriptions of arguments into subsections. Added RandomDisc action. Clarified that it works even if the carousel holds at most one disc.
- [14 Aug 2000] v0.04. Called out interaction with power service. Cleaned up other minor points.
- [15 Aug 2000] v0.05. Renamed to DiscChange. Changed errorCode for Door Stuck to 704 so Door Open can use 703.
- [17 Aug 2000] v0.06. Moved Relationships to Other Service Types to device template.
- [22 Aug 2000] v0.07. Rearranged order of argument sub elements. Changed TRUE/FALSE to 1/0.
- [23 Aug 2000] v0.08. Clarified that there is no effect on state by an action if an error occurs.

## 2. Service Modeling Definitions

### 2.1. Service Type

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

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

The shorthand ChangeDisc:1 is used herein to refer to this service type.

### 2.2. State Variables

Defines two state variables:

- TrayHasDisc: Whether there is a disc in the disc tray.
- DoorIsOpen: Whether the disc tray door is open or closed.

Specifics of these state variables are listed in the table and subsections below.

**Table 1: State variables**

Variable Name	Req. or Opt. <sup>1</sup>	Data Type	Allowed Value	Default Value	Eng. Units
TrayHasDisc	R	boolean	n/a	(none)	(none)
DoorIsOpen	R	boolean	n/a	0	(none)
<i>Non-standard state variables implemented by an UPnP vendor go here.</i>	X	<i>TBD</i>	<i>TBD</i>	<i>TBD</i>	<i>TBD</i>

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

#### 2.2.1. TrayHasDisc

Indicates whether the disc tray currently contains a disc (1) or is empty (0).

#### 2.2.2. DoorIsOpen

Indicates whether the disc tray door is open and available for loading (1) or closed and ready for other operations (0). The default value (0) indicates that a disc changer should close its disc tray door when it is powered up.

#### 2.2.3. Non-Standard State Variables Implemented by an UPnP Vendor

To facilitate certification, non-standard state variables implemented by an UPnP vendor should be included in this service template. The UPnP Device Architecture lists naming requirements for non-standard state variables (cf. section on Description).

#### 2.2.4. Relationships Between State Variables

The value of TrayHasDisc is independent of the value of DoorIsOpen; all four combinations of values (both 1, one or the other 1, both 0) are possible and legal. This implies that implementations must be able to detect whether there is a disc in the tray even when the disc tray door is open.

*Relationships between standard state variable(s) defined herein and any non-standard state variable(s) is TBD.*

## 2.3. Eventing and Moderation

As the table below summarizes, ChangeDisc:1 defines unmoderated eventing for its all of its standard state variables.

**Table 2: Event moderation**

Variable Name	Evented	Moderated Event	Max Event Rate <sup>1</sup>	Logical Combination	Min Delta per Event <sup>2</sup>
TrayHasDisc	yes	no	n/a	n/a	n/a
DoorIsOpen	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

Some control points will need to be able to react to changes when the standard state variables change state. Since both standard state variables are boolean and neither is expected to change rapidly, events for neither are moderated.

However, control points do not need to subscribe to eventing to correctly control this service.

## 2.4. Actions

As the table below summarizes, ChangeDisc:1 defines actions to add a disc into the changer, select the next or previous disc to play, open and close the disc tray door, and to query for the value of the state variables listed above. Except where noted, if an action is an error, calling the action will have no effect on state. Immediately following the 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>
AddDisc	R
NextDisc	R
PrevDisc	R
RandomDisc	R
HasTrayDisc	R
OpenDoor	R
CloseDoor	R
ToggleDoor	R
IsDoorOpen	R

<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. AddDisc

Provides a means to add a disc to the changer. It unloads the disc in the disc tray (if any) and opens the disc tray door (if closed). It is an error if the carousel is full of discs.

#### 2.4.1.1. Arguments

(None.)

#### 2.4.1.2. Effect on State

Sets the TrayHasDisc state variable to 0 and sets the DoorIsOpen state variable to 1. That is,

```
ASSIGN(DoorIsOpen, 1)
ASSIGN(TrayHasDisc, 0)
```

#### 2.4.1.3. Errors

errorCode	errorDescription	Description
402	Invalid Args	See UPnP Device Architecture section on Control.
501	Action Failed	See UPnP Device Architecture section on Control.
702	Carousel Full	See Common Error Codes below.
704	Door Stuck	See Common Error Codes below.
<i>800-899</i>	<i>TBD</i>	<i>(Specified by UPnP vendor.)</i>

### 2.4.2. NextDisc

Provides sequential access to the discs in the carousel. NextDisc closes the disc tray door (if open), unloads the disc in the disc tray (if any), and, assuming a linear ordering of the discs in the carousel, loads the next disc in the carousel into the disc tray with the following exceptions:

- (a) If the disc currently in the tray is the last disc in this ordering, NextDisc loads the first disc in the carousel into the disc tray.
- (b) If there is exactly one disc in the carousel, NextDisc loads this disc into the disc tray (if it isn't already there).
- (c) It is an error if there are no discs in the carousel.

#### 2.4.2.1. Arguments

(None.)

#### 2.4.2.2. Effect on State

Sets the DoorIsOpen state variable to 0 and sets the TrayHasDisc state variable to 1. That is,

```
ASSIGN(DoorIsOpen, 0)
ASSIGN(TrayHasDisc, 1)
```

**2.4.2.3. Errors**

errorCode	errorDescription	Description
402	Invalid Args	See UPnP Device Architecture section on Control.
501	Action Failed	See UPnP Device Architecture section on Control.
701	No Disc	See Common Error Codes below.
704	Door Stuck	See Common Error Codes below.
800-899	TBD	(Specified by UPnP vendor.)

**2.4.3. PrevDisc**

Behaves identically to NextDisc except that it loads the previous disc in the carousel into the disc tray and modifies Exception (a) above:

- (a) If the disc currently in the tray is the first disc in this ordering, PrevDisc loads the last disc in the carousel into the disc tray.

**2.4.3.1. Arguments**

(None.)

**2.4.3.2. Effect on State**

(Same as NextDisc.)

**2.4.3.3. Errors**

(Same as NextDisc.)

**2.4.4. RandomDisc**

Selects a disc at random from within the carousel, providing a partial implementation of a shuffle function. (Some other service must randomly select tracks/scenes/etc. on the disc.) RandomDisc closes the disc tray door (if open), unloads the disc in the disc tray (if any), and, loads a randomly selected disc in the carousel into the disc tray with Exceptions (b) and (c) above as listed for NextDisc.

**2.4.4.1. Arguments**

(None.)

**2.4.4.2. Effect on State**

(Same as NextDisc.)

**2.4.4.3. Errors**

(Same as NextDisc.)

**2.4.5. HasTrayDisc**

Queries for whether the disc tray has a disc in it or not. Operates correctly whether the disc tray door is open or not.

**2.4.5.1. Arguments**

Returns the value of the TrayHasDisc state variable.

Argument(s)	Direction	relatedStateVariable
HasDisc	OUT <sup>R</sup>	TrayHasDisc

<sup>R</sup> Return value.

**2.4.5.2. Effect on State**

(None.)

**2.4.5.3. Errors**

errorCode	errorDescription	Description
402	Invalid Args	See UPnP Device Architecture section on Control.
501	Action Failed	See UPnP Device Architecture section on Control.
800-899	TBD	(Specified by UPnP vendor.)

**2.4.6. OpenDoor**

Opens the disc tray door. Similar to eject if there is a disc in the tray.

**2.4.6.1. Arguments**

(None.)

**2.4.6.2. Effect on State**

Sets the DoorIsOpen state variable to 1. That is,

ASSIGN(DoorIsOpen, 1)

**2.4.6.3. Errors**

errorCode	errorDescription	Description
402	Invalid Args	See UPnP Device Architecture section on Control.
501	Action Failed	See UPnP Device Architecture section on Control.
704	Door Stuck	See Common Error Codes below.
800-899	TBD	(Specified by UPnP vendor.)

**2.4.7. CloseDoor**

Closes the disc tray door.

**2.4.7.1. Arguments**

(None.)

**2.4.7.2. Effect on State**

Sets the DoorIsOpen state variable to 0. That is,

ASSIGN(DoorIsOpen, 0)

**2.4.7.3. Errors**

(Same as OpenDoor.)

**2.4.8. ToggleDoor**

Opens the disc tray door if it is open and closes it otherwise. It enables single-button control of the door by simple control points that do not subscribe to eventing and do not maintain a local copy of the IsDoorOpen state variable.

**2.4.8.1. Arguments**

(None.)

**2.4.8.2. Effect on State**

Sets the DoorIsOpen state variable to the negation of the DoorIsOpen state variable. That is,

ASSIGN(DoorIsOpen, NOT DoorIsOpen)

**2.4.8.3. Errors**

(Same as OpenDoor.)

**2.4.9. IsDoorOpen**

Queries for whether the disc tray door is open or not.

**2.4.9.1. Arguments**

Returns the value of the DoorIsOpen state variable.

Argument(s)	Direction	relatedStateVariable
IsOpen	OUT <sup>R</sup>	DoorIsOpen

<sup>R</sup> Return value.

**2.4.9.2. Effect on State**

(None.)

**2.4.9.3. Errors**

errorCode	errorDescription	Description
402	Invalid Args	See UPnP Device Architecture section on Control.
501	Action Failed	See UPnP Device Architecture section on Control.
704	Door Stuck	See Common Error Codes below.

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

#### 2.4.10. Non-Standard Actions Implemented by an UPnP Vendor

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

#### 2.4.11. Relationships Between Actions

The actions defined herein may be called in any order.

*Relationships between standard action(s) defined herein and any non-standard action(s) is TBD.*

#### 2.4.12. 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.

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	No Disc	There are no discs in the carousel.
702	Carousel Full	There is no room for additional discs in the carousel.
704	Door Stuck	Can't open (or close) the disc tray door.
800-899	TBD	(Specified by UPnP vendor.)

### 2.5. Theory of Operation

To allow a user to load a disc into the changer, a control point invokes AddDisc. A control point does not need to explicitly open the disc tray door because AddDisc does this automatically.

```
// Invoke AddDisc
//
```

If the user puts a disc into the disc tray, when the disc tray door closes, this disc will be ready to play. When another disc is selected, this disc will be moved back into the carousel.

To allow the user to remove a disc from the changer, a control point must select that disc and open the disc tray door.

```
// Select the disc
//   While current disc is not in the disc tray
//     Try the next one
//       Invoke NextDisc
// Open the disc tray door
//   Invoke OpenDoor
//
```

To allow the user to play a disc in the changer, a control point must select that disc. A control point does not need to explicitly close the disc tray door because even if the disc is already selected, selecting a disc will automatically close the door.

```
// Select the disc
//   While current disc is not in the disc tray
//     Try the next one
//       Invoke NextDisc
//
```

(Actually playing the disc is done through another service type and is not described here.)

### 3. XML Service Description

```

<?xml version="1.0"?>
<scpd xmlns="urn:schemas-upnp-org:service-1-0">
  <specVersion> <!-- UPnP version 1.0 -->
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <actionList>
    <action>
      <name>AddDisc</name>
    </action>
    <action>
      <name>NextDisc</name>
    </action>
    <action>
      <name>PrevDisc</name>
    </action>
    <action>
      <name>RandomDisc</name>
    </action>
    <action>
      <name>HasTrayDisc</name>
      <argumentList>
        <argument>
          <name>HasDisc</name>
          <relatedStateVariable>TrayHasDisc</relatedStateVariable>
          <direction>out</direction>
        </argument>
      </argumentList>
    </action>
    <action>
      <name>OpenDoor</name>
    </action>
    <action>
      <name>CloseDoor</name>
    </action>
    <action>
      <name>ToggleDoor</name>
    </action>
    <action>
      <name>IsDoorOpen</name>
      <argumentList>
        <argument>
          <name>IsOpen</name>
          <relatedStateVariable>DoorIsOpen</relatedStateVariable>
          <direction>out</direction>
        </argument>
      </argumentList>
    </action>
    <!-- Declarations for other actions implemented by an -->
    <!--   UPnP vendor (if any) go here. -->
  </actionList>
  <serviceStateTable>
    <stateVariable sendEvents="yes">
      <name>DoorIsOpen</name> <!-- whether disc tray is open -->
      <dataType>boolean</dataType> <!-- 1 if open -->
    </stateVariable>
  </serviceStateTable>
</scpd>

```

```
<defaultValue>0</defaultValue>
</stateVariable>
<stateVariable sendEvents="yes">
  <name>TrayHasDisc</name> <!-- is there a disc in the tray -->
  <dataType>boolean</dataType> <!-- 1 if a disc -->
</stateVariable>
<!-- Declarations for other state variables implemented by an -->
<!-- UPNP vendor (if any) go here. -->
</serviceStateTable>
</scpd>
```

## **4. Test**

TBD.