Feeder:1.0 **Service Template Version 1.01**

For UPnP™ Version 1.0
Status: ***Standardized DCP***
Date: ***September 11, 2002***

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.

Authors	Company
Larry Copp	Hewlett-Packard Co.

Contents

List of Tables

1. Overview and Scope

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

This service-type enables the following functions:

- *Document Feeders for Scanners, Faxes, etc.*

This service template does not address:

- *TBD*

1.1. Change Log for *Feeder:1.0*

0.12 – Removed Key value arguments and variable

0.20 – Incorporated changes from November 9, 2000 review

0.21 – Incorporated changes from November 27, 2000 review

0.22 – Incorporated changes from December 6, 2000 review – Note that these changes reflect a simplified model of feeders.

0.23 – Incorporated change from the December 11, 2000 teleconference review. This includes the following changes:

- Added explanations for allowed State values
- Reworked the Units and Max/Min Sheet size information to use a GetLimits action instead of state variables
- Changed the name of the FeederType variable to FeederMode
- Changed the description of the Load action to include a possible transition to Bottom-Loaded state.

0.24 – Changed to reflect changes from the January 11, 2001 review. This includes the following changes

- Changed the format to reflect Service Template Version 1.0
- As per the recent UPnP Imlementor's Guide, Booleans have values of 1 and 0
- Changed Allowed values of InputJustification object from right and left to inboard and outboard.
- Added Units as an argument to GetLimits action
- Clarified state requirements of the SetFeederMode action
- Added actions to get and set the InputJustification variable
- Fixed the XML (it was missing several actions added last time)

0.40 – Changed to reflect the changes from the Jan 22-23, 2001 meeting. This includes the following changes

- Changed to reflect the format of the Service Template Version 1.01
- Added A_ARG_TYPE_String state variable
- Removed InputJustification as a state variable and added it to the GetLimits action as an argument. Also removed the GetJustification and SetJustification actions.
- Removed some extra arrowheads from the state transition diagrams.
- Updated the XML to reflect the changes
- Added a description of what the XML Service Template is, and how it is used.

0.41 – Changed to reflect the February 20, 2001 teleconference review. This includes the following changes:

- Added footnotes to Table 1 to match the template
- Changed the State Transition Diagram in Figure 1 to use Eject instead of Unload term.
- Changed the state of all direction values in the SCPD to be lower case

- Windows ME seems to require that the direction specification come after the relatedStateVariable specification in the SCPD. This was done on all argument sections in the SCPD

0.50 – Fixed XML errors and updated incorrect links

0.51 – Changed to reflect the April 23, 2001 review. This includes the following changes:

- Added a JobID argument to the Load, Eject, Reset, and SetFeederMode actions
- Deleted the SetUnits, GetUnits, and GetLimits actions
- Added State variables for JobID, SheetHeight, SheetWidth, and InputJustification

0.52 – Changed to reflect the May 22, 2001 review and other inputs:

- Changed A_ARG_TYPE_JobID to JobID
- Changed A_ARG_TYPE_Boolean to EntireDocument (that was the only place it was used). Made it so that there can be a –1 or device-setting value given also.
- Deleted A_ARG_TYPE_String because it is not used.

0.53 – Changed to reflect the June 21, 2001 teleconference review

- Added a *Busy* value to State. The Load, Eject, and Reset actions are invalid during this state.
- Added a *Timeout* variable. This value does not have any actions to provide access. It is an SCPD accessible constant only. This timeout applies when in the Jammed and Busy states.
- Change to a First-Side/Second-Side duplexing model. This impacts theory of operation, and state transition diagrams.
- Changed the Jammed state to Erred, added a FailureCode variable and added the FailureCode as an *out* parameter to the GetState action.
- Added explanations of the Height and Width variables.
- Clarified the behavior of the Eject action when the bEntireDocument parameter is true.

0.70 – Changed to reflect the input from the 31July2001 meeting in Toronto ON. This includes the following:

- The State value *jammed* was changed to *Erred* in version 0.53, however several references were not updated. These have been fixed.
- Changed the State Transition Diagrams to add a transition in the Erred state to prevent leaving when intervention is required.
- Deprecated Error Code 404
- Changed from Error Code 710 – Invalid State to 501 – Action Failed
- Added new Common Error Codes

0.71 – Changed to reflect input from team members. This includes the following:

- Removed the allowedValueList from the MorePages variable
- Changed the type of the EntireDocument variabl

- Changed the value of the InputJustification variable to be a default value not an allowed value.
- Format changes to make the xml more readable

0.80 – Updated the Testing section to reflect what we can do with the Certification Test
0.90 – Changed the version to 0.90 and the status to Design Complete
0.91 – Updated to conform to the Service Template v1.01 and the Service Template Checklist

- Changed Table numbers to make them align with the template.
- Added all non-evented variables to Table 2 – Evented Variables
- Changed all argument names for all actions by appending "In" or "Out" to the name based on the direction of the argument
- Changed nnn limit values to vendor-defined

0.92 – Changed to reflect changes suggested in the Scanner Teleconference held on 9May2002

- Changed the name of the argument in Table 15 from "StateIn" to "StateOut"

1.0 – *Document approved by Steering Committee*

- Changed version to 1.0
- Changed status to Standardized DCP
- Changed date to September 11, 2002
- Changed legal disclaimer on the first page to Appendix G of the SC Org and Process Doc.
- Changed all copyrights to "© 1999-2001 Microsoft Corporation. All rights Reserved."
- In section 2.1 "Service Type", changed the version to 1

2. Service Modeling Definitions

2.1. ServiceType

A service that is compliant with this template is identified with the following service type:

urn:schemas-upnp-org:service:*Feeder:1*

2.2. Service State Table

Table 1: State Variables

Variable Name	Req. or Opt.[1]	Data Type	Allowed Value[2]	Default Value[2]	Eng. Units
Model	*O*	*string*	*Vendor Specific*		
State	*R*	*string*	*[Unloaded | Loaded | Busy | Erred]*	*Unloaded*	*n/a*
FailureCode	*R*	*string*	*[none | Jammed | Timeout]*	*none*	*n/a*
MorePages	*O*	*boolean*		*False (0)*	*n/a*
FeederMode	*R*	*string*	*[Simplex]*	*[Simplex]*	*n/a*
SheetWidth	*R*	*ui4*			*milli-inches*
SheetHeight	*R*	*ui4*			*milli-inches*
InputJustification	*R*	*string*		*vendor specific*	*n/a*
JobID	*R*	*ui4*	*0-4294967295*	*0*	*n/a*
EntireDocument	*R*	*string*	*[device-setting|1|0]*	*True (1)*	*n/a*
Timeout	*R*	*ui4*		*vendor-defined*	*seconds*

2.2.1. Model

Gives the model number of the feeder. This variable is not required on devices that have a fixed feeder.

[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.

2.2.2. State

Describes the state of the feeder. The possible values include the following:

- *Unloaded* – The feeder does not currently have any sheets loaded.
- *Loaded* – (Simplex Mode) – A sheet is loaded and ready to operate on.
- *First-Side-Loaded* – (Duplex Mode) – A sheet is loaded with the top side ready to operate on.
- *Second-Side-Loaded* – (Duplex Mode) – A sheet is loaded with the bottom side ready to operate on.
- *Erred* – The feeder is jammed or erred. No further actions will be accepted until the state is cleared.
- *Busy* – The feeder is currently busy or being used and its state cannot be changed.

Table 1.1 allowedValueList for State

Value	*Req. or Opt.*
Unloaded	*R*
Loaded	*R*
Erred	*R*
Busy	*R*
First-Side-Loaded	*O*[3]
Second-Side-Loaded	*O*[3]

2.2.3. FailureCode

A Failure Code indicates a failure, or error, that does not occur as an immediate result of an action and cannot be reported as part of the action response. Error Codes that occur during the execution of an action are not listed as allowed values of the FailureCode unless they can occur outside of the boundary of the action.

Table 1.2 allowedValueList for FailureCode

Value	*Req. or Opt.*
None	*R*
Jammed	*R*
Timeout	*R*
… (vendor defined)	*O*

2.2.4. MorePages

Indicates whether there are additional pages in the feeder or not.

2.2.5. FeederMode

Indicates the type of feed to use in operation with the current scan. The allowed values may be a vendor specific subset or superset of the given values. *Note: Simplex indicates one sided sheets, Duplex indicates two sided sheets.*

Table 1.3 allowedValueList for FeederMode

Value	*Req. or Opt.*
Simplex	*R*
Duplex	*O*[4]
… (vendor defined)	*O*

[3] For use with Duplex Feeders

[4] For use with a Duplex Feeder

2.2.6. SheetWidth

This value is a constant that represents the minimum and maximum values of the sheet width supported by the Feeder. These values are available in the SCPD, but there are no actions defined to provide access to the actual value. The intent of this variable is only to provide the limits of the sheet width supported by the feeder.

Table 1.4 allowedValueRange for SheetWidth

Value		*Req. or Opt.*
Minimum	*VendorUnique*	*R*
Maximum	*VendorUnique*	*R*

SheetWidth is a constant value. It provides limiting values of the paper size that is supported by the feeder.

2.2.7. SheetHeight

This value is a constant that represents the minimum and maximum values of the sheet height supported by the Feeder. These values are available in the SCPD, but there are no actions defined to provide access to the actual value. The intent of this variable is only to provide the limits of the sheet height supported by the feeder.

Table 1.5 allowedValueRange for SheetHeight

Value		*Req. or Opt.*
Minimum	*VendorUnique*	*R*
Maximum	*VendorUnique*	

SheetHeight is a constant value. It provides limiting values of the paper size that is supported by the feeder.

2.2.8. InputJustification

This value is a constant that is used to indicate the justification supported by the Feeder. This value is in the SCPD, but the actual value of the state variable is not important.

2.2.9. JobID

An argument placeholder for the JobID type arguments.

2.2.10.EntireDocument

Indicates whether the Eject action should eject one sheet or the entire document..

2.2.11.Timeout

Indicates the timeout value in seconds used to prevent deadlocks in the feeder. This value is for information only and cannot be changed.

2.3. Eventing and Moderation

Table 2: Event Moderation

Variable Name	Evented	Moderated Event?	Max Event Rate	Logical Combination	Min Delta per Event
			Determined by N Where Rate = (Event)/(N secs)		(N) * (allowed ValueRange Step)
Model	*No*				
State	*No*				
FailureCode	*No*				
MorePages	*Yes*	*No*			
FeederMode	*No*				
SheetWidth	*No*				
SheetHeight	*No*				
InputJustification	*No*				
JobID	*No*				
EntireDocument	*No*				
Timeout	*No*				

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: Action List

Name	Req. or Opt.
Load	R
Eject	R
Reset	R

Name	Req. or Opt.
GetState	R
SetFeederMode	R
GetFeederMode	R

2.4.1. *(void)* Load(JobIDIn, StateOut)

2.4.1.1. Arguments

Table 4: Arguments for Load

Argument	Direction	RelatedStateVariable
JobIDIn	*IN*	*JobID*
StateOut	*OUT*	*State*

2.4.1.2. Effect on State

Attempt to load a sheet. If successful, set the State to **Loaded, First-Side-Loaded, or Second-Side-Loaded.** The resulting state value is returned. See the state diagrams in *Figure 1* and *Figure 2* on page 14. The Load action is not valid in the Busy state. If it is used in the Busy state, then Action Failed (501) is returned and the action is not performed.

2.4.1.3. Errors

Table 5: Errors for Load

errorCode	errorDescription	Description
501	*Action Failed*	This action cannot be performed in the Busy state
711	*Jammed*	The device has jammed. Clear the paper path and execute the reset action
713	*Feeder Empty*	A load was attempted when there were no more pages available.

2.4.2.

the documents in the current *Job* are ejected. The normal definition of *Job* is assumed to be all sheets in the feeder. However, vendors may redefine the term to fit the capabilities of the feeder (ex. If a feeder can detect separator sheets, then a job would end at the next separator sheet). If *EntireDocumentIn* is false, then one sheet will be ejected.

2.4.2.3. Errors

Table 7: Errors for Eject

errorCode	errorDescription	Description
501	*Action Failed*	The Eject action cannot be performed in the Busy state
711	*Jammed*	The device has jammed. Clear the paper path and execute the reset action

2.4.3. *(void)* Reset(JobIDIn, StateOut)

2.4.3.1. Arguments

Table 8: Arguments for Reset

Argument	Direction	relatedStateVariable
JobIDIn	*IN*	*JobID*
StateOut	*OUT*	*State*

2.4.3.2. Effect on State

Attempt to clear a Erred state. If the action is successful, the state will change to **Unloaded**. The resulting state value is returned. The Reset action is not allowed in the Busy state. If Reset is performed while in the Busy state, Action Failed (501) will be returned and the action will not be performed.

Table 9: Errors for Reset

errorCode	errorDescription	Description
501	*Action Failed*	The Reset action cannot be performed in the Busy state

2.4.4. *(void)* GetState(StateOut, MorePagesOut, FailureCodeOut)

2.4.4.1. Arguments

Table 10: Arguments for GetState

Argument	Direction	relatedStateVariable

2.4.5. *(void)* SetFeederMode(JobIDIn, FeederModeIn)

2.4.5.1. Arguments

Table 11: Arguments for SetFeederMode

Argument	Direction	relatedStateVariable
JobIDIn	*IN*	*JobID*
FeederModeIn	IN	*FeederMode*

2.4.5.2. Effect on State

This action changes the FeederModeIn value to enable or disable duplex scanning. This value must only be changed when the State value is Unloaded, otherwise the FeederModeIn value will not be changed and an *InvalidState error will be returned.*

2.4.5.3. Errors

Table 12: Errors for SetFeederMode

errorCode	errorDescription	Description
501	Action Failed	This action can only be executed when the current State value is Unloaded.

2.4.6. *(void)* GetFeederMode(FeederModeOut)

2.4.6.1. Arguments

Table 13: Arguments for GetFeederMode

Argument	Direction	RelatedStateVariable
FeederModeOut	OUT	*FeederMode*

2.4.6.2. Effect on State

This action returns the current value of the FeederMode setting. No effect on state.

2.4.7. Common Error Codes

Table 14: 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.
501	Action Failed	See UPnP Device Architecture section on Control.

errorCode	errorDescription	Description
600	Argument Value Invalid	The argument value is invalid.
601	Argument Value Out of Range	An argument value is less than the minimum or more than the maximum value of the allowedValueRange, or is not in the allowedValueList.
602	Optional Action Not Implemented	The requested action is optional and is not implemented by the device.
603	Out of Memory	The device does not have sufficient memory available to complete the action. This may be a temporary condition; the control point may choose to retry the unmodified request again later and it may succeed if memory is available.
604 *Proposed*	Human Intervention Required	The device has encountered an error condition which it cannot resolve itself and requires human intervention such as a reset or power cycle. See the device display or documentation for further guidance.

2.5. Theory of Operation

This service is intended to be used with a Scanner, FAX or other similar service and device. It has a simple state machine shown below:

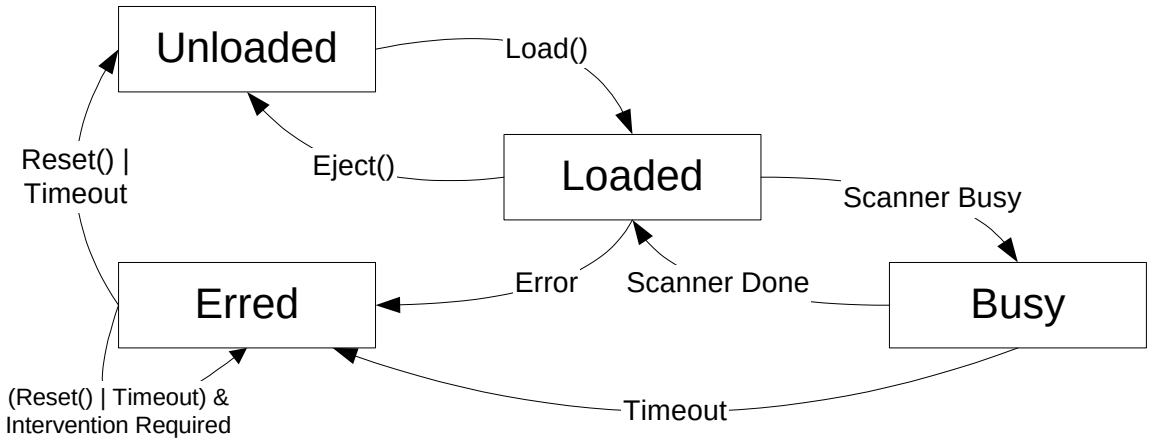

Figure 1 – Simplex Feeder State Diagram

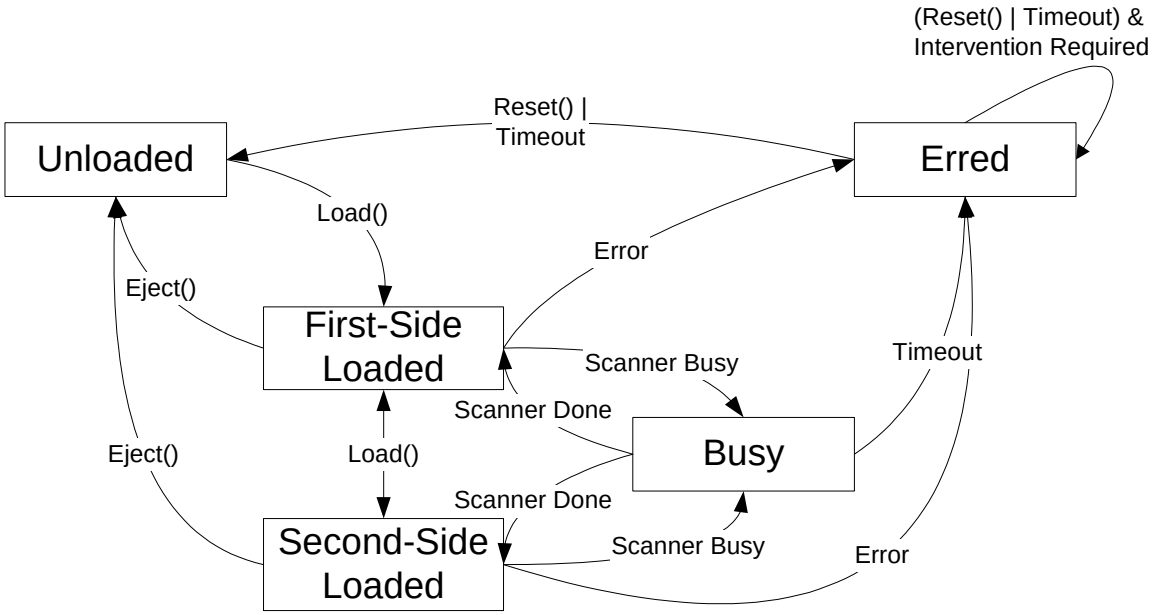

Figure 2 - Duplex Feeder State Diagram

With a Simplex FeederMode value, a load action should always load a new page and end up in the *Loaded* state. An Eject action should always unload any loaded page and end up in the *Unloaded* state. Exceptions would be when there are no pages available (State changes to *Unloaded*) or when a jam occurs (State changes to *Erred)*. With a Duplex *FeederMode* value, a load action should always move from Unloaded to First-Side-Loaded, from First-Side-Loaded to Second-Side-Loaded, or from Second-Side-Loaded to First-Side-Loaded. Exceptions would be when there are no pages available (State changes to *Unloaded*) or when a jam or other error occurs (State changes to *Erred*). The Busy state is a state where no feeder actions should be performed. Entry into and exit from the Busy state are controlled by vendor-specific interactions between the Scanner service and the Feeder service and are not specified in this document.

2.5.1. Enhanced Feeder Operation

The feeder described in this document is a simple feeder. It does not encompass all of the different feeder varieties that the Scanning committee envisioned. Instead, it represents a simple feeder with a simple interface to the scanner. If a vendor has a more complex feeder then they must modify this service to properly represent it. However, care must be taken to keep the same interface for the scanner service.

3. XML Service Template for ***Feeder:1.0***

The XML document below is a sample *Service Control Protocol Document* (SCPD) for a scanner feeder device. It should be modified as needed by a scanner vendor to fully describe the feeder service offered by the scanner device. The scanner device should make the modified document available at the SCPD URL given in the device descriptor. A client will perform an HTTP/GET operation on that URL to get the document. NOTE: The XML comments are for information only and should be removed from the SCPD.

```
<?xml version="1.0"?>
<scpd xmlns="urn:schemas-upnp-org:service-1-0">
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <actionList>
    <action>
      <name>Load</name>
      <argumentList>
        <argument>
          <name>JobIDIn</name>
          <relatedStateVariable>JobID</relatedStateVariable>
          <direction>in</direction>
        </argument>
        <argument>
          <name>StateOut</name>
          <relatedStateVariable>State</relatedStateVariable>
          <direction>out</direction>
        </argument>
      </argumentList>
    </action>
    <action>
      <name>Eject</name>
      <argumentList>
        <argument>
          <name>JobIDIn</name>
          <relatedStateVariable>JobID</relatedStateVariable>
          <direction>in</direction>
        </argument>
        <argument>
          <name>EntireDocumentIn</name>
          <relatedStateVariable>EntireDocument</relatedStateVariable>
          <direction>in</direction>
        </argument>
        <argument>
          <name>StateOut</name>
          <relatedStateVariable>State</relatedStateVariable>
          <direction>out</direction>
        </argument>
      </argumentList>
    </action>
    <action>
      <name>Reset</name>
      <argumentList>
        <argument>
          <name>JobIDIn</name>
```

```
        <relatedStateVariable>JobID</relatedStateVariable>
        <direction>in</direction>
      </argument>
      <argument>
        <name>StateOut</name>
        <relatedStateVariable>State</relatedStateVariable>
        <direction>out</direction>
      </argument>
    </argumentList>
  </action>
  <action>
    <name>GetState</name>
    <argumentList>
      <argument>
        <name>StateOut</name>
        <relatedStateVariable>State</relatedStateVariable>
        <direction>out</direction>
      </argument>
      <argument>
        <name>MorePagesOut</name>
        <relatedStateVariable>MorePages</relatedStateVariable>
        <direction>out</direction>
      </argument>
      <argument>
        <name>FailureCodeOut</name>
        <relatedStateVariable>FailureCode</relatedStateVariable>
        <direction>out</direction>
      </argument>
    </argumentList>
  </action>
  <!-- The SetFeederMode action is optional.It may be removed from
  this document if it is not supported by the scanner device -->
  <action>
    <name>SetFeederMode</name>
    <argumentList>
      <argument>
        <name>JobIDIn</name>
        <relatedStateVariable>JobID</relatedStateVariable>
        <direction>in</direction>
      </argument>
      <argument>
        <name>FeederModeIn</name>
        <relatedStateVariable>FeederMode</relatedStateVariable>
        <direction>in</direction>
      </argument>
    </argumentList>
  </action>
  <action>
    <name>GetFeederMode</name>
    <argumentList>
      <argument>
        <name>FeederModeOut</name>
        <relatedStateVariable>FeederMode</relatedStateVariable>
        <direction>out</direction>
      </argument>
    </argumentList>
```

```
    </action>
    Declarations for other actions added by UPnP vendor (if any) go
      here
  </actionList>
  <serviceStateTable>
    <!-- The Model variable is optional.It may be removed if not
      supported by the device -->
    <stateVariable sendEvents="no">
     <name>Model</name>
     <dataType>string</dataType>
    </stateVariable>
    <stateVariable sendEvents="no">
      <name>State</name>
      <dataType>string</dataType>
      <defaultValue>Unloaded</defaultValue>
      <allowedValueList>
        <allowedValue>Unloaded</allowedValue>
        <allowedValue>Loaded</allowedValue>
        <allowedValue>Busy</allowedValue>
        <allowedValue>Erred</allowedValue>
        <!-- Optional Values for scanners with duplexers
        <allowedValue>First-Side-Loaded</allowedValue>
        <allowedValue>Second-Side-Loaded</allowedValue>
        -->
      </allowedValueList>
    </stateVariable>
    <stateVariable sendEvents="no">
      <name>FailureCode</name>
      <dataType>string</dataType>
      <defaultValue>None</defaultValue>
      <allowedValueList>
        <allowedValue>None</allowedValue>
        <allowedValue>Jammed</allowedValue>
        <allowedValue>Timeout</allowedValue>
      </allowedValueList>
    </stateVariable>
    <!-- The MorePages variable is optional.It may be removed if not
      supported by the device -->
    <stateVariable sendEvents="yes">
      <name>MorePages</name>
      <dataType>boolean</dataType>
      <defaultValue>0</defaultValue>
    </stateVariable>
    <stateVariable sendEvents="no">
      <name>FeederMode</name>
      <dataType>string</dataType>
      <defaultValue>Simplex</defaultValue>
      <allowedValueList>
        <allowedValue>Simplex</allowedValue>
        <!-- Optional Value
        <allowedValue>Duplex</allowedValue>
        -->
        <!-- Additional Vendor Unique values may be added here -->
      </allowedValueList>
    </stateVariable>
    <stateVariable sendEvents="no">
      <name>JobID</name>
```

```
        <dataType>ui4</dataType>
        <defaultValue>0</defaultValue>
        <allowedValueRange>
          <minimum>0</minimum>
          <maximum>vendor-defined</maximum>
          <step>1</step>
        </allowedValueRange>
      </stateVariable>
      <!--SheetWidth is a constant value that is used to define the
        limits of the sheet width in milli-inches -->
      <stateVariable sendEvents="no">
        <name>SheetWidth</name>
        <dataType>ui4</dataType>
        <allowedValueRange>
          <minimum>0</minimum><!--Vendor defined range value -->
          <maximum>vendor-defined</maximum><!-- Vendor defined range
            value -->
          <step>1</step>
        </allowedValueRange>
      </stateVariable>
      <!--SheetHeight is a constant value that is used to define the
        limits of the sheet width in milli-inches -->
      <stateVariable sendEvents="no">
        <name>SheetHeight</name>
        <dataType>ui4</dataType>
        <allowedValueRange>
          <minimum>0</minimum><!--Vendor defined range value -->
          <maximum>vendor-defined</maximum><!-- Vendor defined range
            value -->
          <step>1</step>
        </allowedValueRange>
      </stateVariable>
      <stateVariable sendEvents="no">
        <name>InputJustification</name>
        <dataType>string</dataType>
        <defaultValue>vendor-defined</defaultValue>  <!--Vendor defined
          constant value -->
      </stateVariable>
      <stateVariable sendEvents="no">
        <name>EntireDocument</name>
        <dataType>string</dataType>
        <defaultValue>1</defaultValue>
        <allowedValueList>
          <allowedValue>1</allowedValue><!-- True -->
          <allowedValue>0</allowedValue><!-- False -->
          <allowedValue>device-setting</allowedValue></allowedValueList>
      </stateVariable>
      <stateVariable sendEvents="no">
        <name>Timeout</name>
        <dataType>ui4</dataType>
        <defaultValue>vendor-defined</defaultValue><!--Vendor Specific
          value -->
      </stateVariable>
      Declarations for other state variables added by UPnP vendor (if
        any) go here
    </serviceStateTable>
</scpd>
```

4. Testing

4.1. Syntax Tests

All actions are tested using the UIC Certification Test tool.