Front-end Specification of XISL

Input Modalities

The attributes of <input> element are *type*, *event*, and so on. The *type* attribute specifies the type of input, and the *event* attribute specifies the input event.

Input mode is written in the <param> element contained in the <input> element.

type	event	mode
dtmf	push	digit,character
speech	recognize	command, data, speaker, emotion, …
pointing	l_click, r_click, …	
kyeboard	press	

☐ DTMF

type	*event*	*target*	*match*	*return*
"dtmf"	"push"	Specify a DTMF grammar	Specify a path to an available grammar rule	Variables to hold DTMF events

- Recognition modes

  The author can specify one of the following two modes.

  > digit mode
  >
  > character mode

Example:

```
<input   type="dtmf"   event="push"   target="grammar.gxml"
        match="/grammar/goods_select"   return="number" >
        <param name="mode">digit </param>
</input>
```

In this example, the number pushed by a user is assigned to the variable "number".

□ **Speech**

type	event	target	match	return
"speech"	"recognize"	Specify a speech grammar	Specify a path to an available grammar rule	Variables to hold speech contents

- Recognition modes

The author can specify one of the following four modes.

> command
>
> data
>
> speaker
>
> emotion

Mode can be selected (changed) in the <param> element contained in the <input> element.

Example:

```
<input  type="speech"  event="recognize"  target="grammar.gxml"
        match="/grammar/goods"  return="goods_name" >
        <param name="mode"> data </param>
</input>
```

In this example, the speech word (the name of the goods) is assigned to the variable "goods_name".

□ **Keyboard**

type	event	target	match	return
"keyboard"	"press"	Specify a displayed XML file	Specify a path to the root node of the XML file	Variables to hold pressed key

□ **Pointing**

type	event	target	match	return
"pointing"	"l_click"	Specify an XML file containing target element	Specify a path to the target element of the XML file	Three variables to hold: 1. name of the target 2. abscissa of the point 3. ordinate of the point
	"r_click"			
	"ld_click"			
	"rd_click"			
	"over"			
	"out"			
	"l_drag"			Six variables to hold: 1. name of the target 2. name of the element on the destination point 3. abscissa of start point 4. ordinate of start point 5. abscissa of destination point 6. ordinate of destination point
	"r_drag"			

Output Modalities

type	event	mode
window	open, close, …	
speech	play, stop, …	TTS, rec_speech, …
video	play, stop, …	
audio	play, stop, …	
agent	move, speak, …	

□ **Window**

| *type* | *event* | <param> | |
		name	value
"window"	"open"	"window_name"	name of the window
		"window_type"	type of the window ("normal" or "modal")
		"uri"	URI of the displayed page
		"position"	position of the window (see "move")
		"width"	width of the window
		"height"	height of the window
	"navigate"	"window_name"	name of the window
		"uri"	URI of the newly displayed page
	"close"	"window_name"	name of the window
	"maximize"		
	"minimize"		
	"restore"		
	"resize"	"window_name"	name of the window
		"width"	width of the resized window
		"height"	height of the resized window
	"move"	"window_name"	name of the window
		"position"	position of the window ("center", "left-top", "right-top", "left-bottom", "right-bottom")

☐ **Speech**

- TTS mode

type	*event*	<param>	
		name	value
"speech"	"play"	"id"	speech ID
		"mode"	"TTS"
		"speech_text"	Spoken text
		"alphabet_flag"	"true": Read as word "false": Read as characters
		"num_flag"	"true": Read as digits "false": Read as a number
		"speed"	"0" (slow) to "23" (fast)
		"volume"	"0" (quiet) to "65535" (loud)
		"speaker"	speaker type ("man", "woman", …)
		"emotion"	emotion type ("normal", "angry", …)
	"stop"	"id"	speech ID
	"pause"		

- rec_speech mode

type	*event*	<param>	
		name	value
"speech"	"play"	"id"	speech ID
		"mode"	"rec_speech"
		"uri"	URI of the speech file
		"volume"	"0" (quiet) to "65535" (loud)
	"stop"	"id"	speech ID
	"pause"		

□ **Video**

| *type* | *event* | <param> | |
		name	value
"video"	"open_file"	"id"	video ID
		"uri"	URI of the video file
	"play"	"id"	video ID
	"stop"		
	"pause"		

□ **Audio**

| *type* | *event* | <param> | |
		name	value
"audio"	"open_file"	"id"	audio ID
		"uri"	URI of the audio file
	"play"	"id"	audio ID
	"stop"		
	"pause"		

□　**Anthropomorphic Agent**

type	event	<param>		
		name		value
"agent"	"create"	"agent_name"		name of the agent
		"agent_file"		agent file to display
	"show"	"agent_name"		name of the agent
	"hide"			
	"delete"			
	"move"	"agent_name"		name of the agent
		(1)	"ax"	abscissa of the destination point
			"ay"	ordinate of the destination point
		(2)	"object"	path to the destination XML element
			"position"	move to "center", "left", or "right" of the XML element
	"action"	"agent_name"		name of the agent
		"action_name"		action name
	"balloon"	"agent_name"		name of the agent
		"balloon_text"		text to display in the balloon
	"speak"	"agent_name"		name of the agent
		"speech_text"		spoken text
	"change_voice"	"agent_name"		name of the agent
		"voice"		type of the character (e.g. "boy", "girl", "robot", …)