APEX reference manual

Tom Francart

January 10, 2008

Contents

1 apex

<**apex**>

attrib <**version**> *required* **type** positiveInteger

'Apex will only load experiment files with the correct version attribute. Conversion scripts will be provided to convert to newer versions.'

<**description**> *optional* **type** string

doc 'Describes the entire experiment. Is reported in the results file.'

<**procedure**> *required* **type** procedureType (see p 77)

doc 'Describes the procedure to be used. The procedure controls the sequence of trials that will be presented. Real procedures are derived from apex:procedureType.'

<**corrector**> *required* **type** correctorType (see p 76)

doc 'Describes the corrector to be used. The corrector compares the user input and the correct answer for a certain trail. Real correctors are derived from apex:correctorType'

<**screens**> *required* **type** screensType (see p 33)

doc 'Defines one or more screens to be used elsewhere. A screen is what is visible on the computer screen at a certain instant of time.'

<**datablocks**> *required* **type** datablockstype (see p 31)

doc 'A datablock is the smallest unit of output data used in Apex. Every sound or other stimulus file to be used elsewhere, is to be defined in this section.'

<**devices**> *required* **type** devicesType (see p 24)

doc 'All output devices to be used are enumerated in this section. Before the experiment can be started, Apex tries to open them. The experiment only starts if all devices can be opened. Devices are started in alphabetical order of their id, except if a master is specified. They are stopped in reverse alphabetical order. Thus if you want them to be started or stopped in a certain order, you can change their IDs.'

<**filters**> *required* **type** filtersType (see p 17)

doc 'Filters can be placed anywhere in the output network. They can be real filters, ie blocks that process data, or generators, ie blocks that generate data without any input data.'

<**stimuli**> *required* **type** stimuliType (see p 14)

doc 'A stimulus is the whole of datablocks (for any number of devices) and parameters (fixed and variable) that is presented to the user at a certain instant of time.'

<**connections**> *optional* **type** connectionsType (see p 13)

doc 'In this section, connections can be made between datablocks, filters and devices. \"from\" is only possible for datablocks and filters, \"to\" is only possible for filters that accept input (thus no generators) and devices. Connections can only be made between elements belonging to the same device. If no connections are specified, Apex tries to make a default network by connecting all channels from _ALL_ (== all datablocks) to all channels from the datablocks\' device. Default connections are not made when filters are present.'

<**randomgenerators**> *optional* **type** randomgeneratorsType (see p 11)

doc 'Randomgenerators generate random numbers according to a certain specification and can set any parameter defined elsewhere before stimulus output. All randomgenerators are asked to set their corresponding parameters right before a stimulus is output.'

<**calibration**> *optional* **type** calibrationType (see p 9)

doc 'In this optional section, parameters for the built in calibration mechanism are defined. Calibration is the process of ensuring a relation between a digital stimulus level (dB) and the actual audio output (dBSPL). Calibration parameters are stored in profiles that are persistent between sessions. Profiles are stored in the apex/config/calibration directory and can be edited by hand if required.'

<**results**> *optional* **type** resultsType (see p 9)

doc 'Parameters related to the presentation/analysis of the result of this experiment are defined here.'

<**interactive**> *optional* **type** interactiveType (see p 8)

doc 'For every entry in this section, the value of a certain element in this experiment file will be changed to the value set by the user in a GUI window.'

<**general**> *optional* **type** generalType (see p 7)

doc 'Some general experiment parameters are defined here.'

2 generalType

<**exitafter**> *optional* **type** boolean

doc 'Exit immediately after the experiment was finished. The \"save results\" dialog box is always shown.'

<**autosave**> *optional* **type** boolean

doc 'If defined, apex will automatically select a filename and only present the user with a messagebox asking wheter to save the results yes or no'

<**showresults**> *optional* **type** boolean

doc 'Show the experiment results after transformation with the script defined in the results section in a dialog box. A dialog box will ask for confirmation before showing the results. This option will only work if an xslt script is defined and the results are successfully saved to disk.'

<**waitforstart**> *optional* **type** boolean

doc 'If true, the next trial is only presented after clicking the Start button, selecting Start from the Experiment menu or pressing F5.'

<**allowskip**> *optional* **type** boolean

doc 'If true, the F7 key can be used to skip anything that takes a certain amount of time, eg intro/outro/iti'

<**saveprocessedresults**> *optional* **type** boolean

doc 'Save the experiment results after transformation at the end of the results xml file. This option will only work if an xslt script is defined and the results are successfully saved to disk.'

<**runoutputtest**> *optional* **type** boolean

doc 'Run test to compare output with input files. Only works for WavDevice, and the apex executable must be compiled with WRITEWAVOUTPUT defined. If set, the data send to the soundcard is also written to a file, and afterwards compared with the input. This allows to check if the entire processing/buffering works properly. If the input below is not specified, the test assumes all available datablocks were played sequential. Filters and connections are not taken into account. (eg if two datablocks are defined, the test just checks if the output is identical to the two wavefiles played right after each other). To allow testing filter/loop/connections functionality, create a wavefile that mimics all the processing, and specify the path below, and the test will use it to compare against the output. See the example tests for more info.'

<**outputtestinput**> *optional* **type** anyURI

doc 'Optional: path to a file to compare the output with.'

<**scriptlibrary**> *optional* **type** anyURI

doc 'Library of ECMAScript functions. Functions defined in this library can be used from any other script for the experiment.'

3 interactiveType

<**entry**> *one or more*

attrib <**expression**> *optional* **type** string

'XPath expression defining the element to be modified. The user will be alerted if the expression yields no results. If the expression yields more than one result, every element found will be modified. The modification takes place before validation, so the resulting document will not be used unless valid.

Modifications will be reported in the results file. If expression is not present, the user input will be saved to the results file, but nothing will be modified.'

attrib <**type**> *required*

'Data type of the element to be modified. It impacts the dialog box presented to the user and prevents input of data of the wrong data type.'

attrib <**description**> *required* **type** string

'Description of this entry, will be shown to the user.'

attrib <**default**> *required* **type** string

'Default value for the replacement.'

4 resultsType

<**xsltscript**> *optional* **type** anyURI

doc 'xslt script to be used to transform the output xml file in a more human readable format. Apex looks for xslt scripts in the path defined in the main config file. It defaults to apex/scripts/xslt'

<**matlabscript**> *optional* **type** string

doc 'Matlab script to be used for processing results. Will be added as an XML processing tag to the results file'

5 calibrationType

attrib <**profile**> *required* **type** Name

<**soundlevelmeter**> *optional*

doc 'Information necessary for automatic calibration using an interface to a sound level meter.'

<**plugin**> *required* **type** anyURI

doc 'Filename of the souind level meter plugin to be used'

<**transducer**> *optional* **type** string

doc 'Name of the transducer (microphone, ...) to be used by the SLM. Will only be set if your SLM supports it.'

<**frequency_weighting**> *required*

<**time_weighting**> *required*

<**type**> *required*

<**percentile**> *optional* **type** double

doc 'Percentile of statistics on measurement. If 0 an instantaneous measurement will be made.'

<**time**> *optional* **type** double

doc 'Measurement time in seconds. If percentile is defined, time should be ¿0'

<**stimuli**> *required*

doc 'Stimuli that are used for calibration. These can be, and in some cases should be stimuli that are used in the actual experiment.'

<**stimulus**> *one or more*

attrib <**id**> *optional* **type** IDREF

<**parameters**> *required*

doc 'Parameters that are used to calibrate the system. Generally some output gain parameters of the last stage (eg the wavdevice) are used.'

<**parameter**> *one or more*

attrib <**id**> *optional* **type** IDREF

<**targetamplitude**> *required* **type** double

doc 'Target acoustical measurement of the stimuli defined above during the experiment. Generally in dBSPL.'

<**calibrationamplitude**> *optional* **type** double

doc 'Value to calibrate at. Generally in dBSPL and higher than targetamplitude. Can be changed during calibration.'

<**mute**> *required* **type** double

doc 'Value to be used for the other parameters while calibrating this parameter. Has no effect on the actual calibration. Useful for eg muting the other channel of a headphone.'

<**min**> *optional* **type** double

doc 'Minimum parameter value that can be set by the calibrator.'

<**max**> *optional* **type** double

doc 'Maximum parameter value that can be set by the calibrator.'

6 randomgeneratorsType

<**randomgenerator**> *any*

subtype uniform (see p 13)

subtype gaussian (see p 11)

attrib <**id**> *required* **type** ID

<**min**> *required* **type** double

doc 'Minimum value of the random numbers'

<**max**> *required* **type** double

doc 'Maximum value of the random numbers'

<**type**> *required*

doc 'Data type'

<**parameter**> *required* **type** IDREF

doc 'Parameter to be set to the random value'

7 gaussian

parent randomgenerator (see p 12)

attrib <**id**> *required* **type** ID

<**min**> *required* **type** double

doc 'Minimum value of the random numbers'

<**max**> *required* **type** double

doc 'Maximum value of the random numbers'

<**type**> *required*

doc 'Data type'

<**parameter**> *required* **type** IDREF

doc 'Parameter to be set to the random value'

8 randomgenerator

subtype uniform (see p 13)

subtype gaussian (see p 11)

attrib <**id**> *required* **type** ID

<**min**> *required* **type** double

doc 'Minimum value of the random numbers'

<**max**> *required* **type** double

doc 'Maximum value of the random numbers'

<**type**> *required*

doc 'Data type'

<**parameter**> *required* **type** IDREF

doc 'Parameter to be set to the random value'

9 uniform

parent randomgenerator (see p 12)

attrib <**id**> *required* **type** ID

<**min**> *required* **type** double

doc 'Minimum value of the random numbers'

<**max**> *required* **type** double

doc 'Maximum value of the random numbers'

<**type**> *required*

doc 'Data type'

<**parameter**> *required* **type** IDREF

doc 'Parameter to be set to the random value'

10 connectionsType

<**connection**> *any*

<**from**> *required*

doc 'The channel of the filter/datablock where the connection is started from.'

attrib <**id**> *optional* **type** ID

<**id**> *required*

<**channel**> *required*

doc 'The channel to be used. Channels are zero based, ie the first channel is 0, the second 1, etc. This parameter can have an ID for later referral.'

attrib <**id**> *optional* **type** ID

<**to**> *required*

doc 'The channel of the device/filter where the connection ends.'

attrib <**id**> *optional* **type** ID

<**id**> *required* **type** IDREF

<**channel**> *required*

doc 'The channel to be used. Channels are zero based, ie the first channel is 0, the second 1, etc. This parameter can have an ID for later referral. Channel -1 means mute.'

attrib <**id**> *optional* **type** ID

11 stimuliType

<**fixed_parameters**> *optional*

doc 'Fixed parameters describe fixed properties of every stimulus. They can be used by procedures to select a certain stimulus, they can be presented on the screen or can be used for the users own reference. They have no influence whatever on the actual sound/stimulation that is sent to the device when a particular stimulus is to be output. Every fixed stimulus that is used in any stimulus should be defined here. All fixed parameters defined here, should be present in all stimuli.'

<**parameter**> *any*

attrib <**id**> *required* **type** ID

<**stimulus**> *any*

attrib <**id**> *required* **type** ID

<**description**> *optional* **type** string

doc 'Text description, for your own reference'

<**datablocks**> *required* **type** datablocksType (see p 15)

doc 'Combination of datablocks to be output'

<**variableParameters**> *optional*

doc 'These parameters will be set just before the stimulus is output'

<**parameter**> *any*

attrib <**id**> *required* **type** IDREF

<**fixedParameters**> *optional*

doc 'Fixed parameters are properties of a stimulus that can be displayed or used to select a stimulus from a list based on a certain criterion.'

<**parameter**> *any*

attrib <**id**> *required* **type** IDREF

<**b**> *any* **type** string

<**i**> *any* **type** string

<**u**> *any* **type** string

<**pluginstimuli**> *any*

parent xmlGeneratingPluginType (see p 15)

<**script**> *optional*

attrib <**source**> *required*

<**parameter**> *any*

attrib <**name**> *required* **type** Name

12 xmlGeneratingPluginType

<**script**> *optional*

attrib <**source**> *required*

<**parameter**> *any*

attrib <**name**> *required* **type** Name

13 datablocksType

optional

<**datablock**> *required*

attrib <**id**> *required* **type** Name

<**sequential**> *required*

doc 'Present all child datablocks or child combinations of datablocks sequentially. The result is guaranteed gap-less when using a single device for all children.'

one or more

<**datablock**> *optional*

attrib <**id**> *required* **type** IDREF

<**sequential**> *optional* **type** datablocksSequentialType (see p 16)

<**simultaneous**> *optional*

one or more

<**datablock**> *optional* **type** datablockRefType (see p 17)

<**sequential**> *optional* **type** datablocksSequentialType (see p 16)

<**simultaneous**> *optional* **type** datablocksSimultaneouslType (see p 16

<**simultaneous**> *required* **type** datablocksSimultaneouslType (see p 16)

doc 'Present all child datablocks or child combinations of datablocks simultaneously. The total length of this block is the length of the longest child block. Zero\'s or null stimulation will be added to shorter children if necessary (this is not guaranteed!).'

14 datablocksSimultaneouslType

one or more

<**datablock**> *optional* **type** datablockRefType (see p 17)

<**sequential**> *optional* **type** datablocksSequentialType (see p 16)

<**simultaneous**> *optional* **type** datablocksSimultaneouslType (see p 16)

15 datablocksSequentialType

one or more

<**datablock**> *optional* **type** datablockRefType (see p 17)

<**sequential**> *optional* **type** datablocksSequentialType (see p 16)

<**simultaneous**> *optional* **type** datablocksSimultaneouslType (see p 16)

16 datablockRefType

attrib <**id**> *required* **type** IDREF

17 filtersType

<**filter**> *any*

doc 'Every filter/generator derives from this base type. All filters to be used are defined here.'

subtype generator (see p 23)

subtype amplifier (see p 22)

subtype fader (see p 21)

subtype dataloop (see p 20)

subtype hrtffiltersum (see p 19)

subtype pluginfilter (see p 18)

attrib <**id**> *required* **type** ID

<**device**> *required* **type** IDREF

doc 'ID of the device to which this filter belongs. The mode of the filter (online or offline) is determined by the device.'

<**channels**> *required* **type** positiveInteger

doc 'Number of channels'

<**continuous**> *optional* **type** boolean

doc 'If continuous==true, the filter/generator keeps on running between trials (ie while the user is entering input). Only makes sense for generators.'

18 pluginfilter

parent filterType (see p 18)

attrib <**id**> *required* **type** ID

<**device**> *required* **type** IDREF

doc 'ID of the device to which this filter belongs. The mode of the filter (online or offline) is determined by the device.'

<**channels**> *required* **type** positiveInteger

doc 'Number of channels'

<**continuous**> *optional* **type** boolean

doc 'If continuous==true, the filter/generator keeps on running between trials (ie while the user is entering input). Only makes sense for generators.'

<**plugin**> *required* **type** anyURI

<**parameter**> *any*

attrib <**id**> *optional* **type** ID

attrib <**name**> *required* **type** Name

attrib <**channel**> *optional* **type** nonNegativeInteger

19 filterType

subtype generator (see p 23)

subtype amplifier (see p 22)

subtype fader (see p 21)

subtype dataloop (see p 20)

subtype hrtffiltersum (see p 19)

subtype pluginfilter (see p 18)

attrib <**id**> *required* **type** ID

<**device**> *required* **type** IDREF

doc 'ID of the device to which this filter belongs. The mode of the filter (online or offline) is determined by the device.'

<**channels**> *required* **type** positiveInteger

doc 'Number of channels'

<**continuous**> *optional* **type** boolean

doc 'If continuous==true, the filter/generator keeps on running between trials (ie while the user is entering input). Only makes sense for generators.'

20 hrtffiltersum

parent filterType (see p 18)

attrib <**id**> *required* **type** ID

<**device**> *required* **type** IDREF

doc 'ID of the device to which this filter belongs. The mode of the filter (online or offline) is determined by the device.'

<**channels**> *required* **type** positiveInteger

doc 'Number of channels'

<**continuous**> *optional* **type** boolean

doc 'If continuous==true, the filter/generator keeps on running between trials (ie while the user is entering input). Only makes sense for generators.'

<**anglespeech**> *required*

attrib <**id**> *optional* **type** ID

<anglenoise> *required*

attrib <id> *optional* **type** ID

<snr> *required*

attrib <id> *optional* **type** ID

<speechfile> *required*

attrib <id> *optional* **type** ID

<noisefile> *required*

attrib <id> *optional* **type** ID

21 dataloop

parent filterType (see p 18)

attrib <id> *required* **type** ID

<device> *required* **type** IDREF

doc 'ID of the device to which this filter belongs. The mode of the filter (online or offline) is determined by the device.'

<channels> *required* **type** positiveInteger

doc 'Number of channels'

<continuous> *optional* **type** boolean

doc 'If continuous==true, the filter/generator keeps on running between trials (ie while the user is entering input). Only makes sense for generators.'

<datablock> *required* **type** IDREF

<basegain> *optional* **type** double

doc 'The total gain is basegain+gain. Basegain is useful for amplitude normalization between different blocks'

<**gain**> *required*

parent double (see p **??**)

attrib <**id**> *optional* **type** ID

<**randomjump**> *optional* **type** boolean

doc 'Jump to a random place in the datablock upon start.'

<**jump**> *optional* **type** double

doc 'Jump to a fixed place in the datablock when the generator is started. The place is specified in seconds. Take care that you do not jump further than the length of the wave file.'

22 fader

parent filterType (see p 18)

attrib <**id**> *required* **type** ID

<**device**> *required* **type** IDREF

doc 'ID of the device to which this filter belongs. The mode of the filter (online or offline) is determined by the device.'

<**channels**> *required* **type** positiveInteger

doc 'Number of channels'

<**continuous**> *optional* **type** boolean

doc 'If continuous==true, the filter/generator keeps on running between trials (ie while the user is entering input). Only makes sense for generators.'

<**length**> *required*

doc 'Fade length in mSec.'

attrib <**id**> *optional* **type** ID

<**type**> *required*

doc 'Fader type: linear or cosine shaped.'

<**direction**> *required*

doc 'Fader direction: fadein (beginning of stimulus) or fadeout(end of stimulus)'

23 amplifier

parent filterType (see p 18)

attrib <**id**> *required* **type** ID

<**device**> *required* **type** IDREF

doc 'ID of the device to which this filter belongs. The mode of the filter (online or offline) is determined by the device.'

<**channels**> *required* **type** positiveInteger

doc 'Number of channels'

<**continuous**> *optional* **type** boolean

doc 'If continuous==true, the filter/generator keeps on running between trials (ie while the user is entering input). Only makes sense for generators.'

<**basegain**> *optional* **type** gainparam (see p 23)

doc 'The total gain is basegain+gain. Basegain is useful for amplitude normalization between different blocks'

<**gain**> *one or more*

parent gainparam (see p 23)

attrib <**id**> *optional* **type** ID

attrib <**channel**> *optional* **type** nonNegativeInteger

'Used to specify a single channel to apply gain to, instead of applying it to all channels at once. For example, for an amplifier having two channels, these are set independently by using -3.00 -6.00 Channels are zero based, ie the first channel is 0, the second 1 and so on.'

24 gainparam

parent double (see p **??**)

attrib <**id**> *optional* **type** ID

25 generator

parent filterType (see p 18)

attrib <**id**> *required* **type** ID

<**device**> *required* **type** IDREF

doc 'ID of the device to which this filter belongs. The mode of the filter (online or offline) is determined by the device.'

<**channels**> *required* **type** positiveInteger

doc 'Number of channels'

<**continuous**> *optional* **type** boolean

doc 'If continuous==true, the filter/generator keeps on running between trials (ie while the user is entering input). Only makes sense for generators.'

<**type**> *required*

<**basegain**> *optional* **type** double

doc 'The total gain is basegain+gain. Basegain is useful for amplitude normalization between different blocks'

<**gain**> *required* **type** gainparam (see p 23)

doc 'The total gain is basegain+gain. Gain is a parameter that can be changed by other modules'

<**frequency**> *required*

doc 'Frequency: only makes sense for sinus, square and triangle'

attrib <**id**> *optional* **type** ID

<**phase**> *required*

doc 'Phase (rad): only makes sense for sinus, square and triangle'

attrib <**id**> *optional* **type** ID

<**pulsewidth**> *optional* **type** positiveInteger

doc 'Pulse width in samples. Currently only used for single pulse generator'

<**polarity**> *optional*

26 devicesType

<**master**> *optional* **type** IDREF

doc 'The master device is that last one to be started after all stimuli are loaded for all devices. This feature can be used when using triggers to synchronize devices.'

<**device**> *any*

doc 'All real devices derive from this base.'

subtype wavDeviceType (see p 29)

subtype dummyDeviceType (see p 29)

subtype mixerDeviceType (see p 27)

subtype pluginController (see p 27)

subtype L34DeviceType (see p 26)

subtype clarionDeviceType (see p 25)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

27 clarionDeviceType

parent deviceType (see p 25)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

<**parameter**> *one or more*

doc 'Any parameter known to Clarion. If the parameter does not exist on the device, an error will be shown.'

parent string (see p **??**)

attrib <**id**> *optional* **type** ID

attrib <**name**> *required* **type** Name

28 deviceType

subtype wavDeviceType (see p 29)

subtype dummyDeviceType (see p 29)

subtype mixerDeviceType (see p 27)

subtype pluginController (see p 27)

subtype L34DeviceType (see p 26)

subtype clarionDeviceType (see p 25)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

29 L34DeviceType

parent deviceType (see p 25)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

<**device_id**> *required* **type** nonNegativeInteger

doc 'Device number. Devices are ordered according to the assigned com-port (cfr device manager). Device 0 is a simulated device.'

<**implant**> *required*

doc 'Implant type. CIC4==freedom'

<**trigger**> *required*

doc 'Generate or use a trigger from the audio port. Currently the trigger button status is not checked (nicv2 doesn\'t allow this yet)'

<**volume**> *optional*

attrib <**id**> *optional* **type** ID

<**defaultmap**> *required*

doc 'Defines the default subject map to be used'

<**inline**> *required*

doc 'The map is defined below'

attrib <**id**> *optional* **type** ID

<**number_electrodes**>*required*

<**mode**> *required*

<**pulsewidth**> *required* **type** unsignedInt
<**pulsegap**> *required* **type** unsignedInt
<**period**> *required*
<**channel**> *[1..22]*
attrib <**number**> *required*
attrib <**electrode**> *required*
attrib <**threshold**> *required*
attrib <**comfort**> *required*

<**from_r126**> *required*

doc 'The user can select the map from the R126 wizard. R126 is the clinical fitting software.'

30 pluginController

parent deviceType (see p 25)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

<**plugin**> *required* **type** anyURI

<**parameter**> *any*

attrib <**id**> *optional* **type** ID
attrib <**name**> *required* **type** Name
attrib <**channel**> *optional* **type** nonNegativeInteger

31 mixerDeviceType

parent deviceType (see p 25)

subtype PA5 (see p 29)

subtype soundcardmixer (see p 28)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

<**gain**> *required* **type** channelGainType (see p 28)

32 channelGainType

parent gainparam (see p 23)

attrib <**id**> *optional* **type** ID

attrib <**channel**> *optional* **type** nonNegativeInteger

'Used to specify a single channel to apply gain to, instead of applying it to all channels at once. For example, for an amplifier having two channels, these are set independently by using -3.00 -6.00 Channels are zero based, ie the first channel is 0, the second 1 and so on.'

33 soundcardmixer

parent mixerDeviceType (see p 27)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

<**gain**> *required* **type** channelGainType (see p 28)

<**type**> *required*

34 PA5

parent mixerDeviceType (see p 27)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

<**gain**> *required* **type** channelGainType (see p 28)

35 dummyDeviceType

parent deviceType (see p 25)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

36 wavDeviceType

parent deviceType (see p 25)

attrib <**id**> *required* **type** ID

attrib <**mode**> *optional* **type** ref:deviceModesEnum

'Processing mode of this device. Impacts all associated filters.'

<**driver**> *optional*

doc 'Driver architecture to be used for sound output'

<**card**> *optional* **type** string

doc 'Name of the soundcard to be used. Cards are defined in the main config file. Every soundcard in the system can be enumerated in the main configfile. Use the ID defined there. Otherwise the \"default\" card can be used.'

<**channels**> *required* **type** positiveInteger

doc 'Number of output channels to be used. This is restricted by the selected driver, with a maximum of 2 for portaudio.'

<**gain**> *any* **type** channelGainType (see p 28)

doc 'Final gain of the device, implemented with a software mixer.'

<**samplerate**> *required*

doc 'Sample rate of the device. Warning: not all sample rates are supported by all devices and some drivers automatically convert to other sample rates. Check your sound card documentation.'

<**buffersize**> *optional*

doc 'Applies to the soundcard buffer size, not the internal apex buffer size. Larger sizes are more efficient, smaller sizes have smaller latencies. Some notes: for asio, latency is always (2 * buffersize + overhead) / samplerate. The overhead is very small, eg 16 samples for an Rme Multiface. Portaudio however always uses a number of internal buffers, mostly 4. Hence, actual latency is (2 * 4 * buffersize + overhead) / samplerate. For asio/jack: must be set to the system buffer size.'

<**blocksize**> *optional*

doc 'Applies to the block size that is used in the internal apex audio streams (for filters etc.). Larger sizes are more efficient, smaller sizes have smaller latencies.'

<**buffersize_apex**> *optional*

doc 'Size of internal apex sound buffer in seconds. This buffer ensures that no buffer underruns occur. The default value is 1s. If continuous filters are used, the system will need this amount of seconds before the next stimulus can be output. Note that the value in seconds is rounded down to the nearest multiple of 8192 samples (the buffersize used to fill the buffer): so using 1 second here, with

a samplerate of 44100, the actual amount of time will be 40960 samples / 44100 samples/sec = 0.9 seconds. If a buffersize smaller than 16384 samples is specified, a buffer of 16384 samples will be used.'

<**padzero**> *optional* **type** positiveInteger

doc 'Add the given number of empty (filled with zero\'s) buffers to the output on the end of a stream. This avoids dropping of the last N frames on some soundcards (notoriously LynxOne)'

<**setcardmixer**> *optional* **type** boolean

doc 'Set all output gains on the software mixer 0 dB to avoid quantization noise and set all input gains to -infinite dB to avoid picking up unwanted signals. The software mixer is is the mixer that is normally controlled using the windows mixer or the /dev/mixer interface. For all implementations of our knowledge this is a software mixer. Warning: not all soundcards are supported. All soundcards that support the windows mixer and the RME mixer are supported.'

<**cisim**> *optional*

doc 'Parameters for automatic CI simulation'

<**number_electrodes**>*required* **type** positiveInteger

<**pulserate**> *required* **type** integer

<**noiseband**> *one or more*

attrib <**electrode**> *optional* **type** positiveInteger

37 datablockstype

<**uri_prefix**> *optional*

doc 'The prefix for every filename used below. Only used if a filename is a relative path. This prefix is relative to the path of this experiment file.'

parent string (see p **??**)

attrib <**source**> *optional*

<**datablock**> *any*

attrib <**id**> *required* **type** ID

<**device**> *required*

doc 'The device used to play the datablock.'

<**description**> *optional* **type** string

doc 'Text description, for your own reference'

<**birth**> *optional* **type** dateTime

doc 'Time stamp of the last modification of the data file. If specified, it will be checked and an error will be shown if it does not match.'

<**checksum**> *optional*

doc 'MD5 checksum of the datablock. If check==true, apex will check whether the checksum matches and issue a warning if it doesn\'t. Is useful to ensure data integrity.'

attrib <**check**> *optional* **type** boolean

<**uri**> *required* **type** anyURI

doc 'Filename of the corresponding datafile. If it is a relative path, it will be prefixed with the uri_prefix defined above.'

<**channels**> *optional* **type** positiveInteger

doc 'Number of channels: defaults to the number of channels in a wav-file or to 1 in all other cases.'

<**loop**> *optional* **type** positiveInteger

doc 'Number of times the datablock should be looped'

<**plugindatablocks**> *any*

parent xmlGeneratingPluginType (see p 15)

<**script**> *optional*

attrib <**source**> *required*

<**parameter**> *any*

attrib <**name**> *required* **type** Name

38 screensType

<**uri_prefix**> *optional* **type** prefixType (see p 76)

doc 'Defines a prefix for the filenames of media files used in the screens below. The prefix is concatenated in the front of the filenames given below before trying to open any file. When specifying the file protocol, make sure to only use one / eg: file:/c:/tests/file.xml and NOT file://c:/tests/file.xml'

<**general**> *optional*

doc 'Some general properties of the entire Apex window are set in this section. They are applied for each particular screen defined below.'

<**stopbutton**> *optional* **type** boolean

doc 'Show a red panic button that immediately stops all output when clicked.'

<**repeatbutton**> *optional* **type** boolean

doc 'Show a repeat button that will repeat the last stimulus.'

<**showpanel**> *optional* **type** boolean

doc 'Show the panel; in case of childmode: a movie if defined'

<**showmenu**> *optional* **type** boolean

doc 'Determined whether the apex main menu (containing File, Calibration, etc.) is shown'

<**fullscreen**> *optional* **type** boolean

doc 'If true, the main apex window will be shown full-screen, without window title or taskbar or whatever'

<**intertrialscreen**> *optional*

doc 'A screen to show in between trials (after feedback that is)'

attrib <**length**> *optional* **type** positiveInteger 'The period to show the screen'

<**reinforcement**> *optional*

doc 'Defines what kind of feedback about the procedure and correctness of answers is shown to the user. This is valid for the whole experiment.'

<**progressbar**> *required* **type** boolean

doc 'Show a progress bar in the right hand panel, indicating the experiment progress. Not functional for multiprocedure or trainingprocedure.'

<**feedback**> *required*

doc 'Show visual feedback according to the correctness of the last answer. Feedback is shown using an upward or downward pointing thumb in the right hand panel.'

attrib <**length**> *optional* **type** nonNegativeInteger

'Length of the feedback in ms. If feedback is false but a length is present, the specified time will be the time between two trials.'

<**feedback_on**> *optional*

doc 'If feedback is shown, it is shown as a picture in the panel, but a screen element can also be highlighted.'

<**feedback_picture_positive**> *optional* **type** string

doc 'Picture to be shown in the panel on positive feedback.'

<**feedback_picture_negative**> *optional* **type** string

doc 'Picture to be shown in the panel on negateive feedback.'

<**showcurrent**> *optional* **type** boolean

doc 'Show a border around the screen element corresponding to the currently playing stimulus.'

<**style_apex**> *optional*

doc 'Style that is to be applied to the whole of apex'

<**style**> *optional* **type** stylesheetType (see p 51)

doc 'Style that is to be applied to all screens'

<**childmode**> *optional*

doc 'Defines the elements used in child mode.'

<**intro**> *optional*

doc 'The screen to show as intro before the experiment starts'

attrib <**length**> *optional* **type** nonNegativeInteger

'Length of the movie in ms, if not specified, the system will wait for the movie to end and then continue.'

<outro> *optional*

doc 'The screen to show as outro after the experiment has finished'

attrib **<length>** *optional* **type** nonNegativeInteger

'Length of the movie in ms'

<panel> *optional* **type** anyURI

doc 'The panel reinforcement movie to use'

<defaultFont> *optional* **type** string

doc 'Name of the default font to be used for all elements of every screen. Possible names: any font found by QFontDialog'

<defaultFontsize> *optional* **type** positiveInteger

doc 'Size of the default font to be used for all elements of every screen. Unit: points as defined by the system'

<screen> *any* **type** screenType (see p 35)

doc 'Every single screen defines a screen with a certain ID to be used elsewhere in the experiment file.'

<pluginscreens> *any*

parent xmlGeneratingPluginType (see p 15)

<script> *optional*

attrib **<source>** *required*

<parameter> *any*

attrib **<name>** *required* **type** Name

39 screenType

required

attrib **<id>** *required* **type** ID

<hLayout> *optional* **type** hScreenLayoutType (see p 73)

doc 'Horizontal layout: places elements next to each other'

<**vLayout**> *optional* **type** vScreenLayoutType (see p 71)

doc 'Vertical layout: places elements above each other'

<**gridLayout**> *optional* **type** gridScreenLayoutType (see p 70)

doc 'Gridlayout: places elements in an (irregular) grid.'

<**twoPartLayout**> *optional* **type** twoPartLayoutType (see p 75)

doc 'Layout with a small upper part and larger lower part'

<**arcLayout**> *optional* **type** arcScreenLayoutType (see p 36)

doc 'Arc layout: places elements in a semi-circle. Useful for localization experiments.'

<**buttongroup**> *optional*

attrib <**id**> *required* **type** Name

<**button**> *one or more*

attrib <**id**> *required* **type** Name

<**default_answer_element**> *optional* **type** Name

doc 'Default element for getting the user input to send to the corrector.'

40 arcScreenLayoutType

required

parent screenLayoutType (see p 74)

attrib <**id**> *optional* **type** Name

attrib <**width**> *required* **type** int

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**type**> *required*

'TODO'

<**hLayout**> *optional* **type** hScreenLayoutType (see p 73)

<**vLayout**> *optional* **type** vScreenLayoutType (see p 71)

<**gridLayout**> *optional* **type** gridScreenLayoutType (see p 70)

<**arcLayout**> *optional* **type** arcScreenLayoutType (see p 36)

<**button**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional*

parent shortcutBaseType (see p 51)

attrib <**modifier**> *optional*

'Dead key to be used together with the shortcut'

attrib <**hex**> *optional* **type** boolean

'If true, parses the shortcut as a hexadecimal number. See the Key enum in qtnamespace.h for known values'

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional*

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**text**> *required* **type** string

doc 'Text to be printed on the button'

<**label**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**text**> *required* **type** string

<**answerlabel**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**parameterlabel**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**parameter**> *required*

parent IDREF (see p **??**)

attrib <**name**> *optional* **type** string

'Parameter name to show in the list'

attrib <**expression**> *optional* **type** ref:mathexpression

'Expression to be evaluated. Expressions must be in the format a*x+b, where x is the parameter value and a and b are floats'

<**textEdit**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**text**> *optional* **type** string

<**inputmask**> *optional* **type** string

doc 'Input mask to restrict what can be typed in the textbox. Defined in the Qt documentation. If inputmask==\"numbers\", only numeric input will be allowed.'

<**picture**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**path**> *required* **type** anyURI

doc 'Filename of the picture. The format can be any of: PNG, BMP, XBM, XPM, JPEG, MNG, GIF, PBM (P1 or P4), PGM (P2 or P5), and PPM (P3 or P6). If path is relative, it will be prefixed with the prefix given above.'

<**feedback**> *optional*

doc 'If used, overrides the normal feedback (using coloured borders) by drawing a different picture on screen.'

<**highlight**> *required* **type** anyURI

<**positive**> *required* **type** anyURI

<negative> *required* **type** anyURI

<**flash**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**path**> *required* **type** anyURI

doc 'Flash movie to be put on the screen.'

<**feedback**> *optional* **type** feedbackPathsType (see p 54)

doc 'If used, overrides the normal feedback (using coloured borders) by putting a different movie on screen.'

<**parameterlist**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**parameter**> *one or more* **type** showparameterType (see p 58)

<**picturelabel**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**path**> *required* **type** anyURI

doc 'Filename of the picture. The format can be any of: PNG, BMP, XBM, XPM, JPEG, MNG, GIF, PBM (P1 or P4), PGM (P2 or P5), and PPM (P3 or P6). If path is relative, it will be prefixed with the prefix given above.'

<**spinBox**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**value**> *optional* **type** integer

doc 'Default value'

<**min**> *optional* **type** integer

doc 'Minimum value'

<**max**> *optional* **type** integer

doc 'Maximum value'

<**step**> *optional* **type** integer

doc 'Stepsize'

<**parameter**> *optional* **type** IDREF

doc 'Parameter to be set to the value of the spinbox. This parameter will be set the NEXT trial.'

<**reset**> *optional* **type** boolean

doc 'Reset the value for every new trial'

<**checkBox**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**text**> *required* **type** string

doc 'Text to be printed right to the checkbox'

<**isChecked**> *optional* **type** integer

doc 'Determines whether checkbox is checked by default'

<**slider**> *optional*

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**orientation**> *optional* **type** string

doc 'Orientation of slider'

<**min**> *optional* **type** integer

doc 'Minimum value of slider'

<**max**> *optional* **type** integer

doc 'Maximum value of slider'

<**value**> *optional* **type** integer

doc 'Default value of slider'

<**tickInt**> *optional* **type** integer

doc 'Interval between slider ticks'

<**stepSize**> *optional* **type** integer

doc 'Step size on single arrow key press'

<**pageSize**> *optional* **type** integer

doc 'Step size of pageUp pageDown key press'

41 colorType

42 shortcutType

parent shortcutBaseType (see p 51)

attrib <**modifier**> *optional*

'Dead key to be used together with the shortcut'

attrib <**hex**> *optional* **type** boolean

'If true, parses the shortcut as a hexadecimal number. See the Key enum in qtnamespace.h for known values'

43 shortcutBaseType

44 stylesheetType

45 screenElementType

subtype screenSpinBoxType (see p 68)

subtype screenButtonType (see p 67)

subtype screenCheckBoxType (see p 65)

subtype screenSliderType (see p 64)

subtype screenParameterlistType (see p 62)

subtype screenLabelType (see p 61)

subtype screenPictureLabelType (see p 60)

subtype screenAnswerLabelType (see p 59)

subtype screenParameterLabelType (see p 57)

subtype screenTextEditType (see p 56)

subtype screenPictureType (see p 54)

subtype screenFlashType (see p 53)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

46 screenFlashType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**path**> *required* **type** anyURI

doc 'Flash movie to be put on the screen.'

<**feedback**> *optional* **type** feedbackPathsType (see p 54)

doc 'If used, overrides the normal feedback (using coloured borders) by putting a different movie on screen.'

47 feedbackPathsType

<**highlight**> *required* **type** anyURI

<**positive**> *required* **type** anyURI

<**negative**> *required* **type** anyURI

48 screenPictureType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**path**> *required* **type** anyURI

doc 'Filename of the picture. The format can be any of: PNG, BMP, XBM, XPM, JPEG, MNG, GIF, PBM (P1 or P4), PGM (P2 or P5), and PPM (P3 or P6). If path is relative, it will be prefixed with the prefix given above.'

<**feedback**> *optional* **type** feedbackPathsType (see p 54)

doc 'If used, overrides the normal feedback (using coloured borders) by drawing a different picture on screen.'

49 screenTextEditType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**text**> *optional* **type** string

<**inputmask**> *optional* **type** string

doc 'Input mask to restrict what can be typed in the textbox. Defined in the Qt documentation. If inputmask==\"numbers\", only numeric input will be allowed.'

50 screenParameterLabelType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**parameter**> *required* **type** showparameterType (see p 58)

51 showparameterType

parent IDREF (see p **??**)

attrib <**name**> *optional* **type** string

'Parameter name to show in the list'

attrib <**expression**> *optional* **type** ref:mathexpression

'Expression to be evaluated. Expressions must be in the format a*x+b, where x is the parameter value and a and b are floats'

52 screenAnswerLabelType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

53 screenPictureLabelType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**path**> *required* **type** anyURI

doc 'Filename of the picture. The format can be any of: PNG, BMP, XBM, XPM, JPEG, MNG, GIF, PBM (P1 or P4), PGM (P2 or P5), and PPM (P3 or P6). If path is relative, it will be prefixed with the prefix given above.'

54 screenLabelType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**text**> *required* **type** string

55 screenParameterlistType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**parameter**> *one or more* **type** showparameterType (see p 58)

56 screenSliderType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**orientation**> *optional* **type** string

doc 'Orientation of slider'

<**min**> *optional* **type** integer

doc 'Minimum value of slider'

<**max**> *optional* **type** integer

doc 'Maximum value of slider'

<**value**> *optional* **type** integer

doc 'Default value of slider'

<**tickInt**> *optional* **type** integer

doc 'Interval between slider ticks'

<**stepSize**> *optional* **type** integer

doc 'Step size on single arrow key press'

<**pageSize**> *optional* **type** integer

doc 'Step size of pageUp pageDown key press'

57 screenCheckBoxType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**text**> *required* **type** string

doc 'Text to be printed right to the checkbox'

<**isChecked**> *optional* **type** integer

doc 'Determines whether checkbox is checked by default'

58 screenButtonType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**text**> *required* **type** string

doc 'Text to be printed on the button'

59 screenSpinBoxType

parent screenElementType (see p 51)

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

attrib <**id**> *required* **type** Name

<**style**> *optional* **type** stylesheetType (see p 51)

<**width**> *optional* **type** nonNegativeInteger

doc 'TODO: units?'

<**height**> *optional* **type** nonNegativeInteger

<**shortcut**> *optional* **type** shortcutType (see p 51)

<**font**> *optional* **type** string

doc 'TODO: how is the font specified?'

<**fontsize**> *optional* **type** positiveInteger

doc 'Text font size, in points as defined by the system'

<**bgcolor**> *optional* **type** colorType (see p 51)

<**fgcolor**> *optional* **type** colorType (see p 51)

<**floating**> *optional* **type** boolean

doc 'Put the element in it\'s own window instead of embedding it in the current layout'

<**disabled**> *optional* **type** boolean

doc 'If set to true, the element will always be disabled (can be used to \"guide the eye\")'

<**value**> *optional* **type** integer

doc 'Default value'

<**min**> *optional* **type** integer

doc 'Minimum value'

<**max**> *optional* **type** integer

doc 'Maximum value'

<**step**> *optional* **type** integer

doc 'Stepsize'

<**parameter**> *optional* **type** IDREF

doc 'Parameter to be set to the value of the spinbox. This parameter will be set the NEXT trial.'

<**reset**> *optional* **type** boolean

doc 'Reset the value for every new trial'

60 gridScreenLayoutType

required

parent screenLayoutType (see p 74)

attrib <**id**> *optional* **type** Name

attrib <**width**> *required* **type** int

attrib <**height**> *required* **type** int

attrib <**columnstretch**> *optional* **type** ref:stretchregexp

'Stretch factor for the columns: a list of integers separated by comma\'s. If specified, there should be as much integers as columns. The width of the columns will be proportional to the numbers. E.g. if width=2 and columnstretch=\"1,2\", the second column will be twice as wide as the first. columnstretch=\"2,4\" would have the same effect.'

attrib <**rowstretch**> *optional* **type** ref:stretchregexp

'Stretch factor for the rows: a list of integers separated by comma\'s. If specified, there should be as much integers as rows. The width of the rows will be proportional to the numbers. E.g. if height=3 and rowstretch=\"1,2,1\", the second row will be twice as wide as the first and thirs. rowstretch=\"2,4,2\" would have the same effect.'

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

<**hLayout**>	*optional*	**type** hScreenLayoutType (see p 73)
<**vLayout**>	*optional*	**type** vScreenLayoutType (see p 71)
<**gridLayout**>	*optional*	**type** gridScreenLayoutType (see p 70)
<**arcLayout**>	*optional*	**type** arcScreenLayoutType (see p 36)
<**button**>	*optional*	**type** screenButtonType (see p 67)
<**label**>	*optional*	**type** screenLabelType (see p 61)
<**answerlabel**>	*optional*	**type** screenAnswerLabelType (see p 59)
<**parameterlabel**>	*optional*	**type** screenParameterLabelType (see p 57)
<**textEdit**>	*optional*	**type** screenTextEditType (see p 56)
<**picture**>	*optional*	**type** screenPictureType (see p 54)
<**flash**>	*optional*	**type** screenFlashType (see p 53)
<**parameterlist**>	*optional*	**type** screenParameterlistType (see p 62)
<**picturelabel**>	*optional*	**type** screenPictureLabelType (see p 60)
<**spinBox**>	*optional*	**type** screenSpinBoxType (see p 68)
<**checkBox**>	*optional*	**type** screenCheckBoxType (see p 65)
<**slider**>	*optional*	**type** screenSliderType (see p 64)

61 vScreenLayoutType

required

parent screenLayoutType (see p 74)

attrib	<**id**>	*optional*	**type** Name
attrib	<**height**>	*required*	**type** int
attrib	<**x**>	*optional*	**type** int
attrib	<**y**>	*optional*	**type** positiveInteger
attrib	<**row**>	*optional*	**type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

<**hLayout**>	*optional*	**type** hScreenLayoutType (see p 73)
<**vLayout**>	*optional*	**type** vScreenLayoutType (see p 71)
<**gridLayout**>	*optional*	**type** gridScreenLayoutType (see p 70)
<**arcLayout**>	*optional*	**type** arcScreenLayoutType (see p 36)
<**button**>	*optional*	**type** screenButtonType (see p 67)
<**label**>	*optional*	**type** screenLabelType (see p 61)
<**answerlabel**>	*optional*	**type** screenAnswerLabelType (see p 59)
<**parameterlabel**>	*optional*	**type** screenParameterLabelType (see p 57)
<**textEdit**>	*optional*	**type** screenTextEditType (see p 56)
<**picture**>	*optional*	**type** screenPictureType (see p 54)
<**flash**>	*optional*	**type** screenFlashType (see p 53)
<**parameterlist**>	*optional*	**type** screenParameterlistType (see p 62)
<**picturelabel**>	*optional*	**type** screenPictureLabelType (see p 60)
<**spinBox**>	*optional*	**type** screenSpinBoxType (see p 68)
<**checkBox**>	*optional*	**type** screenCheckBoxType (see p 65)
<**slider**>	*optional*	**type** screenSliderType (see p 64)

62 hScreenLayoutType

required

parent screenLayoutType (see p 74)

attrib <**id**> *optional* **type** Name

attrib <**width**> *required* **type** int

attrib <**x**> *optional* **type** int

attrib <**y**> *optional* **type** positiveInteger

attrib <**row**> *optional* **type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib <**col**> *optional* **type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

<**hLayout**> *optional* **type** hScreenLayoutType (see p 73)

<**vLayout**> *optional* **type** vScreenLayoutType (see p 71)

<**gridLayout**> *optional* **type** gridScreenLayoutType (see p 70)

<**arcLayout**> *optional* **type** arcScreenLayoutType (see p 36)

<**button**> *optional* **type** screenButtonType (see p 67)

<**label**> *optional* **type** screenLabelType (see p 61)

<**answerlabel**> *optional* **type** screenAnswerLabelType (see p 59)

<**parameterlabel**> *optional* **type** screenParameterLabelType (see p 57)

<**textEdit**> *optional* **type** screenTextEditType (see p 56)

<**picture**> *optional* **type** screenPictureType (see p 54)

<**flash**> *optional* **type** screenFlashType (see p 53)

<**parameterlist**> *optional* **type** screenParameterlistType (see p 62)

<**picturelabel**>	*optional*	**type** screenPictureLabelType (see p 60)
<**spinBox**>	*optional*	**type** screenSpinBoxType (see p 68)
<**checkBox**>	*optional*	**type** screenCheckBoxType (see p 65)
<**slider**>	*optional*	**type** screenSliderType (see p 64)

63 screenLayoutType

required

subtype hScreenLayoutType (see p 73)

subtype vScreenLayoutType (see p 71)

subtype gridScreenLayoutType (see p 70)

subtype twoPartLayoutType (see p 75)

subtype arcScreenLayoutType (see p 36)

attrib <**id**> *optional* **type** Name

<**hLayout**>	*optional*	**type** hScreenLayoutType (see p 73)
<**vLayout**>	*optional*	**type** vScreenLayoutType (see p 71)
<**gridLayout**>	*optional*	**type** gridScreenLayoutType (see p 70)
<**arcLayout**>	*optional*	**type** arcScreenLayoutType (see p 36)
<**button**>	*optional*	**type** screenButtonType (see p 67)
<**label**>	*optional*	**type** screenLabelType (see p 61)
<**answerlabel**>	*optional*	**type** screenAnswerLabelType (see p 59)
<**parameterlabel**>	*optional*	**type** screenParameterLabelType (see p 57)
<**textEdit**>	*optional*	**type** screenTextEditType (see p 56)
<**picture**>	*optional*	**type** screenPictureType (see p 54)
<**flash**>	*optional*	**type** screenFlashType (see p 53)
<**parameterlist**>	*optional*	**type** screenParameterlistType (see p 62)

<**picturelabel**>	*optional*	**type** screenPictureLabelType (see p 60)
<**spinBox**>	*optional*	**type** screenSpinBoxType (see p 68)
<**checkBox**>	*optional*	**type** screenCheckBoxType (see p 65)
<**slider**>	*optional*	**type** screenSliderType (see p 64)

64 twoPartLayoutType

required

parent screenLayoutType (see p 74)

attrib	<**id**>	*optional*	**type** Name
attrib	<**ratio**>	*required*	**type** float
attrib	<**direction**>	*required*	**type** ref:horivertType
attrib	<**x**>	*optional*	**type** int
attrib	<**y**>	*optional*	**type** positiveInteger
attrib	<**row**>	*optional*	**type** int

'Row in the grid. Is the same as y, but should not be used together with y'

attrib	<**col**>	*optional*	**type** positiveInteger

'Column in the grid. Is the same as x, but should not be used together with x'

<**hLayout**>	*optional*	**type** hScreenLayoutType (see p 73)
<**vLayout**>	*optional*	**type** vScreenLayoutType (see p 71)
<**gridLayout**>	*optional*	**type** gridScreenLayoutType (see p 70)
<**arcLayout**>	*optional*	**type** arcScreenLayoutType (see p 36)
<**button**>	*optional*	**type** screenButtonType (see p 67)
<**label**>	*optional*	**type** screenLabelType (see p 61)

<**answerlabel**>	*optional*	**type** screenAnswerLabelType (see p 59)
<**parameterlabel**>	*optional*	**type** screenParameterLabelType (see p 57)
<**textEdit**>	*optional*	**type** screenTextEditType (see p 56)
<**picture**>	*optional*	**type** screenPictureType (see p 54)
<**flash**>	*optional*	**type** screenFlashType (see p 53)
<**parameterlist**>	*optional*	**type** screenParameterlistType (see p 62)
<**picturelabel**>	*optional*	**type** screenPictureLabelType (see p 60)
<**spinBox**>	*optional*	**type** screenSpinBoxType (see p 68)
<**checkBox**>	*optional*	**type** screenCheckBoxType (see p 65)
<**slider**>	*optional*	**type** screenSliderType (see p 64)

65 prefixType

parent string (see p **??**)

attrib <**source**> *optional*

66 correctorType

subtype isequal (see p 77)

subtype cvc (see p 77)

subtype alternatives (see p 77)

67 alternatives

parent correctorType (see p 76)

<**answers**> *optional*

doc 'All possible answers are specified here. A screen element is associated with a time slot.'

<**answer**> *one or more*

attrib <**number**> *optional* **type** positiveInteger
'Time slot of the stimulus'

attrib <**value**> *optional* **type** Name
'Name of the screen element corresponding to the time slot'

68 cvc

parent correctorType (see p 76)

<**language**> *required*

69 isequal

parent correctorType (see p 76)

70 procedureType

subtype adaptiveProcedureType (see p 95)

subtype constantProcedureType (see p 93)

subtype trainingProcedureType (see p 90)

subtype multiProcedureType (see p 90)

subtype pluginProcedureType (see p 78)

attrib <**id**> *optional* **type** ID

'The ID is optional, except for child procedures of a multiprocedure. IDs are reported in the results file.'

71 pluginProcedureType

parent procedureType (see p 77)

attrib <**id**> *optional* **type** ID

'The ID is optional, except for child procedures of a multiprocedure. IDs are reported in the results file.'

<**parameters**> *required*

parent procedureParametersType (see p 81)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials). Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

<**script**> *required*

doc 'Name of the script to be used as plugin procedure TODO'

<**adjust_parameter**> *optional* **type** IDREF

doc 'Parameter to be modified by the procedure'

<**parameter**> *any*

attrib <**name**> *required* **type** Name

<**trials**> *required*

<**trial**> *any*

attrib <**id**> *required* **type** ID

<**answer**> *any*

doc 'The correct answer for this trial. Can be the name of a screen element, a string, a number or mixed xml data.'

<**key**> *any* **type** string

doc 'Keyword, to be used by the corrector'

<**skey**> *any* **type** string

doc 'Part of a split keyword, to be used by the corrector.'

<**answer_element**> *optional* **type** IDREF

doc 'Screen element that contains the user input to be used by the corrector.'

<**screen**> *required*

doc 'Screen to be shown'

attrib <**id**> *required* **type** Name

<**stimulus**> *one or more*

doc 'One or more stimuli. An adaptive procedure using a fixed parameter will select one of these stimuli. Other procedures select a random stimulus if more than one is present. Notice that it is not guaranteed that all stimuli will be presented an equal number of times, in the case a random stimulus is selected here.'

attrib <**id**> *required* **type** Name

<**standard**> *any* **type** stimulusRefType (see p 80)

doc 'The standard to be used when procedure/choices¿1. The standard will be presented on every time-slot except for one when the stimulus is presented. If more than one standard is defined, a random standard will be selected for each presentation. If e.g. 3 standards are defined and choices=4, a possible sequence would be standard1 stimulus1 standard2 standard1'

72 stimulusRefType

attrib <**id**> *required* **type** Name

73 procedureParametersType

subtype adaptiveProcedureParametersType (see p 87)

subtype constantProcedureParametersType (see p 85)

subtype trainingProcedureParametersType (see p 84)

subtype pluginProcedureParametersType (see p 82)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials). Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

74 pluginProcedureParametersType

parent procedureParametersType (see p 81)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials). Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

<**script**> *required*

doc 'Name of the script to be used as plugin procedure TODO'

<**adjust_parameter**> *optional* **type** IDREF

doc 'Parameter to be modified by the procedure'

<**parameter**> *any*

attrib <**name**> *required* **type** Name

75 trainingProcedureParametersType

parent procedureParametersType (see p 81)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials). Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

76 constantProcedureParametersType

parent procedureParametersType (see p 81)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials). Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

77 adaptiveProcedureParametersType

parent procedureParametersType (see p 81)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials). Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

<**nUp**> *required* **type** positiveInteger

doc 'Number of times the user has to give a correct answer before the parameter is adapted'

<**nDown**> *required* **type** positiveInteger

doc 'Number of times the user has to give a wrong answer before the parameter is adapted'

<**adapt_parameter**> *one or more* **type** IDREF

doc 'Parameter to be adapted. Can be a fixed, variable or general parameter. If more than one parameter is specified, all parameters will be set to the same value. Only the first parameter given can be a fixed parameter.'

<**start_value**> *required* **type** string

doc 'Start value of the parameter'

<**stop_after_type**> *required*

doc 'Criterion to be used to stop the procedure. The procedure will be stopped after stop_after instances of this event.'

<**stop_after**> *required* **type** positiveInteger

doc 'The procedure ends after stop_after events of type stop_after_type'

<**min_value**> *optional* **type** double

doc 'Minimal value of the parameter. If the procedure tries to go below this value, the parameter is saturated and saturation is reported on screen and in the results file.'

<**max_value**> *optional* **type** double

doc 'Maximal value of the parameter. If the procedure tries to go above this value, the parameter is saturated and saturation is reported on screen and in the results file.'

<**rev_for_mean**> *optional* **type** positiveInteger

doc 'Number of reversals to be taken for mean value. Currently only used in xslt scripts to convert/analyse the resuls file.'

<**larger_is_easier**> *required* **type** boolean

doc 'Larger values of the parameter are easier than smaller values'

<**repeat_first_until_correct**> *optional* **type** boolean

doc 'Repeat the first trial untill the answer is correct.'

<**stepsizes**> *required*

doc 'Defines the stepsizes to be used to adapt the parameter'

<**change_after**> *optional*

doc 'Change the stepsize after a certain number of trials or a certain number of reversals'

<**stepsize**> *one or more*

doc 'The procedure uses a step of size \"size\" after \"begin\" events of type \"change_after\" have occured'

attrib <**begin**> *required* **type** nonNegativeInteger

attrib <**size**> *required* **type** double

78 multiProcedureType

parent procedureType (see p 77)

attrib <**id**> *optional* **type** ID

'The ID is optional, except for child procedures of a multiprocedure. IDs are reported in the results file.'

<**parameters**> *required*

<**order**> *required*

doc 'Order of interleaving the child procedures.'

<**procedure**> *one or more* **type** procedureType (see p 77)

doc 'All child procedures MUST have an ID'

79 trainingProcedureType

parent procedureType (see p 77)

attrib <**id**> *optional* **type** ID

'The ID is optional, except for child procedures of a multiprocedure. IDs are reported in the results file.'

<**parameters**> *required*

parent procedureParametersType (see p 81)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials). Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

<**trials**> *required* **type** trialsType (see p 92)

80 trialsType

<**trial**> *any* **type** trialType (see p 92)

81 trialType

attrib <**id**> *required* **type** ID

<**answer**> *any*

doc 'The correct answer for this trial. Can be the name of a screen element, a string, a number or mixed xml data.'

<**key**> *any* **type** string

doc 'Keyword, to be used by the corrector'

<**skey**> *any* **type** string

doc 'Part of a split keyword, to be used by the corrector.'

<**answer_element**> *optional* **type** IDREF

doc 'Screen element that contains the user input to be used by the corrector.'

<**screen**> *required* **type** screenRefType (see p 93)

doc 'Screen to be shown'

<**stimulus**> *one or more* **type** stimulusRefType (see p 80)

doc 'One or more stimuli. An adaptive procedure using a fixed parameter will select one of these stimuli. Other procedures select a random stimulus if more than one is present. Notice that it is not guaranteed that all stimuli will be presented an equal number of times, in the case a random stimulus is selected here.'

<**standard**> *any* **type** stimulusRefType (see p 80)

doc 'The standard to be used when procedure/choices¿1. The standard will be presented on every time-slot except for one when the stimulus is presented. If more than one standard is defined, a random standard will be selected for each presentation. If e.g. 3 standards are defined and choices=4, a possible sequence would be standard1 stimulus1 standard2 standard1'

82 screenRefType

attrib <**id**> *required* **type** Name

83 constantProcedureType

parent procedureType (see p 77)

attrib <**id**> *optional* **type** ID

'The ID is optional, except for child procedures of a multiprocedure. IDs are reported in the results file.'

<**parameters**> *required*

parent procedureParametersType (see p 81)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials).

Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

<**trials**> *required* **type** trialsType (see p 92)

doc 'If a trial contains more than one stimulus, a random one will be selected.'

84 adaptiveProcedureType

parent procedureType (see p 77)

attrib <**id**> *optional* **type** ID

'The ID is optional, except for child procedures of a multiprocedure. IDs are reported in the results file.'

<**parameters**> *required*

parent procedureParametersType (see p 81)

<**presentations**> *required* **type** positiveInteger

doc 'Number of times every trial will be presented.'

<**skip**> *optional* **type** integer

doc 'Number of trials that will be presented before the actual presentations start. Eg if skip=2 and presentations=3 then first 2 trials will be presented and then 3*#trials. If the order is sequential, the skipped trials will be the first skip trials from the trial list, repeated if necessary. If the order is random, the skipped trials will be picked from the trial list without replacement, repeating this procedure if necessary (if skip¿nTrials). Trials that are to be skipped in the results analysis are marked with the skip tag in the results file. Warning: this has not been thorougly checked for other procedures than constant.'

<**order**> *required*

doc 'The order in which to present the trials.'

<**defaultstandard**> *optional* **type** IDREF

doc 'The default standard: only relevant if choices¿1 This standard will be used if no standard is defined in a certain trial.'

<**uniquestandard**> *optional* **type** boolean

doc 'If uniquestandard is true and multiple standards are defined per trial, Apex will try to present another standard in each interval of the trial.'

<**choices**> *optional*

doc 'Number of choices: choices stimuli are presented, one of which is the stimulus. The others are the current standard (defined in the trial or the default standard). To be combined with an \"alternatives\" corrector.'

attrib <**select**> *optional* **type** ref:selectChoicesType

'If select is defined, the stimulus will only be presented in the intervals enumerated in the select attribute. e.g.: if select=\"2,3\", the stimulus will never occur in interval 1, but only in 2 and 3'

<**pause_between_stimuli**> *optional* **type** positiveInteger

doc 'If specified, a pause of n milliseconds will be introduced between successive stimulus/standard presentations'

<**time_before_first_stimulus**> *optional* **type** double

doc 'If specified, apex will wait for the given number of seconds before starting the first stimulus when the procedure is started.'

<**input_during_stimulus**> *optional* **type** boolean

<**nUp**> *required* **type** positiveInteger

doc 'Number of times the user has to give a correct answer before the parameter is adapted'

<**nDown**> *required* **type** positiveInteger

doc 'Number of times the user has to give a wrong answer before the parameter is adapted'

<**adapt_parameter**> *one or more* **type** IDREF

doc 'Parameter to be adapted. Can be a fixed, variable or general parameter. If more than one parameter is specified, all parameters will be set to the same value. Only the first parameter given can be a fixed parameter.'

<**start_value**> *required* **type** string

doc 'Start value of the parameter'

<**stop_after_type**> *required*

doc 'Criterion to be used to stop the procedure. The procedure will be stopped after stop_after instances of this event.'

<**stop_after**> *required* **type** positiveInteger

doc 'The procedure ends after stop_after events of type stop_after_type'

<**min_value**> *optional* **type** double

doc 'Minimal value of the parameter. If the procedure tries to go below this value, the parameter is saturated and saturation is reported on screen and in the results file.'

<**max_value**> *optional* **type** double

doc 'Maximal value of the parameter. If the procedure tries to go above this value, the parameter is saturated and saturation is reported on screen and in the results file.'

<**rev_for_mean**> *optional* **type** positiveInteger

doc 'Number of reversals to be taken for mean value. Currently only used in xslt scripts to convert/analyse the resuls file.'

<**larger_is_easier**> *required* **type** boolean

doc 'Larger values of the parameter are easier than smaller values'

<**repeat_first_until_correct**> *optional* **type** boolean

doc 'Repeat the first trial untill the answer is correct.'

<**stepsizes**> *required*

doc 'Defines the stepsizes to be used to adapt the parameter'

<**change_after**> *optional*

doc 'Change the stepsize after a certain number of trials or a certain number of reversals'

<**stepsize**> *one or more*

doc 'The procedure uses a step of size \"size\" after \"begin\" events of type \"change_after\" have occured'

attrib <**begin**> *required* **type** nonNegativeInteger

attrib <**size**> *required* **type** double

<**trials**> *required* **type** trialsType (see p 92)

doc 'Trials are selected according to the sequence parameter. A trial can contain multiple stimuli. If the parameter to be adapted is fixed, a stimulus is selected from this list according to the target value of the parameter of the procedure. If more than one stimulus has the same fixed parameter value, a random stimulus is selected out of this subset. If no exact match for the fixed parameter is found, the closest match is used. If the parameter to be adapted

is variable, the value of the parameter is set right before stimulus output.'