Action Behaviors

Animation Action Behaviors

Configure and Play Animation

Class Reference

Starts, stops, pauses, resumes animations and offers additional properties for configuration. Also sends AnimationEvents with the state of the animation. These Events can be received by an AnimationStateCondition.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
Animation	Candera::Animation::AnimationPlayer	The animation that should be controlled.
Action	CgiStudioControl::AnimationAction::Enum	Action that shall be executed on the animation: Start, Stop, Pause, Resume, Speedup, Finish, Change.
Direction	CgiStudioControl::AnimationDirection::Enum	Playback direction of the animation.
Speed	FeatStd::Float	Playback speed of the animation.
RepeatCount	FeatStd::Optional<FeatStd::UInt32>	Count of replays. If set to 0 the animation loops till it gets stopped.
EventDispatchStrategy	EventDispatchStrategyType::Enum	The way the event gets sent. *Possible values: Direct, Broadcast, Bubble, Tunnel, DedicatedBehavior* For more information, please refer to the Event Dispatching page.

DedicatedBehavior	Candera::Behavior*	A dedicated process value behavior that will receive the animation event. *This property can be set in the right side of the Scene Tree Panel.*

Advanced Properties

Property Name	Type	Description
FinishTime	FeatStd::Optional<Candera::Animation::WorldTimeType>	Finish time in milliseconds. *This property is only available if the property "Action" is checked and set to "Finish".*
IncreasedSpeed	FeatStd::Optional<FeatStd::Float>	Increased playback speed of the animation. If set and Play is executed again during playback, animation will speed up (can be used only once per animation!). Zero means immediate position seek. *This property is only available in the following configuration:* *an "Animation" is configured* *"Action" is set to "Start" or "SpeedUp"* *"RepeatMode" is not checked* *"RepeatCount" is not checked*
IncreasedSpeedRevertMargin	FeatStd::Optional<Candera::Animation::WorldTimeType>	Defines trailing part of animation (in milliseconds) that will be played at normal speed again. If not set, speedup will last until the end of animation. *This property is only available if property "IncreasedSpeed" is checked.*
RepeatMode	FeatStd::Optional<AnimationRepeatMode::Enum>	Defines how the animation shall be repeated. *Possible values: Replay, Bounce* *This property is only available if an animation is set in the property "Animation".*
SequenceStartTime	FeatStd::Optional<Candera::Animation::SequenceTimeType>	The start point of the animation. [ms] *This property is only available if an animation is set in the property "Animation".*
SequenceDurationTime	FeatStd::Optional<Candera::Animation::SequenceTimeType>	The duration of the animation. [ms] *This property is only available if an animation is set in the property "Animation".*

Control Action Behaviors

Breadcrumb Action

Class Reference

An Action Behavior that sends breadcrumb action events. It allows you to control a Breadcrumb or a node with a Breadcrumb Behavior attached. This behavior is available for 2D scenes only.

The operations that can be performed are:

- SetPath - sets elements contained in the breadcrumb.
- BackToPrevious - calls breadcrumb to create a path without the last element from the ones existing in the breadcrumb.
- BackToElement - the action will be called if the element exists in the breadcrumb. It will create a path from beginning to the selected element from the breadcrumb.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Action	CgiStudioControl::AnimationAction::Enum	Action that will be called on the breadcrumb. *Possible values: SetPath, BackToPrevious, BackToElement*
Path	String	The path with a separator that will be passed when the action is called. The separator is defined in the receiving breadcrumb.
BreadcrumbNode	Candera::AbstractNodePointer	Breadcrumb Control Node.

Change Selection

Class Reference

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.

Sends a ChangeSelectionEvent to the SelectionGroup. The SelectionGroup then sends a ChangeSelectionEvent to all of its receivers. The event is emitted via **BubbleEventDispatchStrategy** to a SelectionGroupBehavior. For more information, please refer to Event Dispatching page.

Common Properties

This behavior does not have any configurable properties.

PickColor

An action behavior that picks the color of the pixel touched on a node with a bitmap effect and stores it in a Property Provider. The stored color can be referenced from other behaviors, such as Set Color, and applied to other nodes.

Common Properties

Property Name	Type	Description
PropertyProvider	CgiStudioControl::PropertyProviderBehavior*	Specifies the Property Provider in which the picked color is stored.
TouchIndicationNode	Candera::AbstractNodePointer	Specifies the node used as a marker for the selected position. When set, this behavior moves it to the touch location (for example, a dot on a color wheel).

Conditional Action

Class Reference

A conditional action can be used to add further checking of conditions while actions are already performed. E.g. a click may trigger one action in a specific control state and another if this state is not set.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Condition	Candera::ConditionBehavior*	The ConditionBehavior that TriggerBehavior passes a value to and expect a response from to see if Action needs to be invoked.
Actions	Candera::ActionBehaviorArrayProperty	The set of ActionBehaviors that TriggerBehavior invokes in the event that it receives a positive response from Condition.
ElseActions	Candera::ActionBehaviorArrayProperty	The set of ActionBehaviors that TriggerBehavior invokes in the event that it receives a negative response from Condition.

Postpone	bool	Set this flag to true to perform the actions after all other actions have been performed. This may become necessary if the actions alter the state and other conditional actions may be invoked due to that changed state.
PostponeElse	bool	Set this flag to true to perform the else-actions after all other actions have been performed. This may become necessary if the actions alter the state and other conditional actions may be invoked due to that changed state.

Change Drawer State

Class Reference

An Action that allows to hide/show the contents of a Drawer or a Node with a Drawer Behavior attached. This behavior is available for 2D scenes only.

The operation that can be performed are: Close, Open or Toggle.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
Action	CgiStudioControl::DrawerEventType::Enum	The action that should be performed. *Possible values: Open, Close, Toggle*
EventDispatchStrategy	CgiStudioControl::EventDispatchStrategyType::Enum	The way the event gets sent. *Possible values: Direct, Broadcast, Bubble, Tunnel, DedicatedBehavior* For more information, please refer to the Event Dispatching page.

Advanced Properties

Property Name	Type	Description
DedicatedBehavior	Candera::Behavior*	A dedicated behavior that will receive the event. Only available when Dispatch Strategy is set to 'DedicatedBehavior'.

Set Focus

Sets keyboard focus to the selected control when triggered. Can be used for example to change focus between List controls.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
FocusCamera	Candera::AbstractNodePointer	Set the camera node for this focus Action.

Interpolate Value Action

Changes the animation mode (for example, SpeedBased) of the InterpolateProcessValue behavior when the relevant action is triggered.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
Action	InterpolateValueAction::Enum	Action that will be called on the interpolate value behavior. *Possible values: SetMode, Reset*
AnimationMode	InterpolationAnimationMode::Enum	Use the InterpolationAnimationMode to select the desired type of animation in the InterpolateProcessValueBehavior. *Possible values: SpeedBased, TimeBased, Instant*

Interval Switch Action

This action changes the state of the interval switch behavior (by selecting a target Node). With this behavior we can set the state to the following values: on, off, blinking.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
SwitchState	IntervalSwitchState::Enum	SwitchState: IntervalSwitchState: Enum *Possible values: On, Off, Blinking*

Keyboard Action

Manages keyboard control by executing configured actions such as open, close, toggle, and layout switch on the target keyboard node, using defined layout and auto-hide settings when triggered.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Action	KeyboardAction::Enum	Keyboard action to execute. *Possible values: Open, Close, Toggle, InvokeKey, ShiftLayout, SwitchLayout, SetConfiguration*
KeyboardNode	Candera::AbstractNodePointer	Keyboard Node.
AutoHideKeyboard	bool	Should keyboard be hidden when node loses focus and show again when node is focused?

Advanced Properties

Property Name	Type	Description
Layout	String	Layout name. *This property is only available if "Action" is set to "Open", "ShiftLayout" or "SwitchLayout".*
Notifier	Candera::AbstractNodePointer	Event sender. If Notifier is invalid then default is node which has attached behavior. *This property is only available if "Action" is set to "InvokeKey".*
TextNode	Candera::AbstractNodePointer	Text node. *This property is only available if "Action" is set to "InvokeKey".*

ImageNode	Candera::AbstractNodePointer	ImageNode node. *This property is only available if "Action" is set to "InvokeKey".*
FocusNode	Candera::AbstractNodePointer	FocusNode node. *This property is only available if "Action" is set to "InvokeKey".*
Configuration	String	Keyboard configuration. *This property is only available if "Action" is set to "SetConfiguration".*

Map Action

Call a specified action on selected MapBehavior.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
MapNode	Candera::AbstractNodePointer	Map node on which this action should be called.
Action	MapAction::Enum	Action that will be called on the MapBehavior. *Possible values: Reset, Pan, Zoom, Tilt, LookAt, CenterAtLocation, Rotate, SetRotation, ChangeTheme.*
Latitude	Float	GPS latitude at which the camera looks at.
Longitude	Float	GPS longitude at which the camera looks at.
Zoomlevel	Float	Determines how close is the camera to the ground level.
Bearing	Float	Determines the bearing angle at which the camera is rotated relative to north.
Tilt	Float	Determines the tilt angle at which the camera looks at specified location.
SyleUrl	String	Theme that describes how the map looks like and how is styled.

Map Location Action

Represents a location on the map which can be sent to MapBehavior.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.

TargetNode	Candera::AbstractNodePointer	Map node on which this action should be called.
Latitude	Float	Latitude of the location.
Longitude	Float	Longitude of the location.
LocationName	String	Human-readable name of the location.
Address	String	Address of the location (usually street name).

Navigation Action

Invoke a specified navigation action on selected Map

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
MapNode	Candera::AbstractNodePointer	Map node on which this action should be called.
Action	MapNavigationAction::Enum	Action that will be called on the MapNode. *Possible values: StartNavi, StopNavi, ReverseRoute*

Menu Screen Action

Class Reference

Description

Action Behavior which sends MenuScreen action event allows you control the menu screen. The operations it can do are: add, delete element and change screen page. This behavior is available for 2D scenes only.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.

Action	MenuScreenAction::Enum	Action that will be called on the menu screen. *Possible values: EditMode, Delete, SwapRight, SwapLeft, AddScreen, DeleteLastScreen*
MenuItemNode	Candera::AbstractNodePointer	Needed to identify which node should be deleted. This property is only available if "Action" is set to "Delete".
MenuScreenNode	Candera::AbstractNodePointer	MenuScreen node to which event will be sent.

Page Indicator Action

Class Reference

Sends an event to PageIndicator from external control (change number of pages, change active index)

It is also used by PageIndicatorElement (ActivateDot) to activate clicked element and to define visual properties of the PageIndicatorElement. This behavior is available for 2D scenes only.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Action	PageIndicatorAction::Enum	Action to perform on PageIndicator (change number of pages, change active index) *Possible values: ChangeActiveIndex, ChangeNumberOfPages, ActiveDot*

Properties available when *Action* is set to *ChangeActiveIndex* or *ChangeNumberOf Pages*

Property Name	Type	Description
Value	FeatStd::Int32	Value to be sent to PageIndicator (for ChangeActiveIndex and ChangeNumberOfPages actions)
ChangeValue	ChangeValue::Enum	Type of value change to be sent to PageIndicator (for ChangeActiveIndex and ChangeNumberOfPages actions) *Possible values: Absolute or Relative*
PageIndicatorNode	Candera::AbstractNodePointer	PageIndicator node to communicate with.

Properties for usage within PageIndicatorElement (ActivateDot)

These properties are used only when the behavior is used within PageIndicatorElement and action is set to ActivateDot.

Property Name	Type	Description
RenderNode	Candera::AbstractNodePointer	Render node for current PageIndicatorElement (used only for PageIndicatorElement)
NormalImagePropertyProvider	CgiStudioControl::PropertyProviderBehavior*	Property Behavior which contains the image of normal (unselected) PageIndicatorElement (defines visual aspect of current PageIndicatorElement). *This property can be set in the right side of the Scene Tree panel.*
ActiveImagePropertyProvider	CgiStudioControl::PropertyProviderBehavior*	Property Behavior which contains the image of active (selected) PageIndicatorElement (defines visual aspect of current PageIndicatorElement). *This property can be set in the right side of the Scene Tree panel.*

Set Control State

Class Reference

Sets the state of a control by sending an Event to a control considering the selected EventDispatchStrategy.

Supports: Any Node with a Handle Control State attached.

Used in: SpinBox

See also: Set Image, Set Text, Set Color

Example Usage

- Hover - Hover Leave Example

Common Properties

Property Name	Type	Description

Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
State	ControlStateEnum::Enum	Choose the State which has to be changed to True or False. *Possible values: Selected, Pressed, Enabled, Highlighted, KeyboardFocused, ScopeFocused, Hover, Touched*
ChangeTo	bool	Switch the state of the control to True or False.
UseForChildNodes	bool	State change will also affect all child nodes of the target node
EventDispatchStrategy	EventDispatchStrategyType::Enum	The way the event gets sent. *Possible values: Direct, Broadcast, Bubble, Tunnel, DedicatedBehavior* For more information, please refer to the Event Dispatching page.

Advanced Properties

Property Name	Type	Description
DedicatedBehavior	Candera::Behavior *	A dedicated behavior that will receive the event. *This property is only available when Dispatch Strategy is set to 'DedicatedBehavior'.*

Start Timer

Class Reference

Start Timer Behavior starts a timer. If the delay time is passed the behavior's node or target node will receive a Timer Event. The Timer Event can then be handled by an Event Handler with a Check Timer Behavior to trigger an action.

Used in: ScrollBar

See also: Check Timer, OnTimer

Example Usage

- Start Timer Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
Delay	FeatStd::UInt32	Delay time for the timer event (specification in milliseconds)

Custom Action Behaviors

Click to Message

Sends a custom message on invoke action event.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Identifier	String	The identifier of the click message.

Copy Textures

When triggered, the textures of an appearance are used on the appearance of the node. This behavior is available for 3D scenes only.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
Appearance	SharedPointer<Candera::Appearance>	The appearance to set on the node.

Gesture Action Behaviors

Transform Gesture

Use the transform gesture action to apply the transformation of a transform gesture event (pan, pinch and rotate) to a node. You can use this behavior as an action or directly without an event handler. This behavior is available for 2D scenes only.

Pan

If you touch the Render Node with one finger, the content that you have touched will follow your finger. If you touch it with two fingers the point of the content that hits the center between your fingers will follow the center between your fingers.

- Use the Maximum/Minimum properties for X/Y to limit the movement of the position to a certain area.
- Use the Pan mode selector to limit the transformation to X or Y axis.
- Deactivate the Pan to disallow any position change.

Pinch

If you pinch/stretch the content with your fingers the content will be scaled down/up.

- Use the Maximum/Minimum properties to limit the absolute scale of the node.
- Use the Pinch mode selector to either limit the scaling to X or Y axis or to switch to non-uniform scaling where both axis are scaled individually.
- Deactivate the Pinch to disallow any scale change.

Rotate

If you touch the content with two fingers and drag them around the rotation change of the line between your fingers will be applied on the content.

- Use the Maximum/Minimum properties to limit the absolute scale of the node.
- Use the Pinch mode selector to either limit the scaling to X or Y axis or to switch to none uniform scaling where both axis are scaled individually.

> Only if all transformations are allowed the content is able to follow your fingers. Otherwise there may be a displacement that matches the configuration but not the finger position (e.g. if the pan/scale is locked to an axis or Rotation is disabled but scale is still set to uniform scale). The default values are chosen to most fit the common use cases (e.g. it is more likely that uniform scale is required than a non-uniform scale).

Example Usage

- Transform Gesture Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
Pan	FeatStd::Optional <PanGestureType::Enum>	The pan transformation of a pan gesture is applied to the node.
Pinch	FeatStd::Optional <PinchGestureType::Enum>	The scale transformation of a pinch gesture is applied to the node.
Rotate	bool	The rotate transformation of a rotate gesture is applied to the node.

Advanced Properties

Property Name	Type	Description
PanMinimumX	FeatStd::Optional <FeatStd::Float>	The minimum pan X value. *This property is only available if panning in x direction is enabled.*
PanMinimumY	FeatStd::Optional <FeatStd::Float>	The minimum pan Y value. *This property is only available if panning in y direction is enabled.*
PanMaximumY	FeatStd::Optional <FeatStd::Float>	The maximum pan X value. *This property is only available if panning in x direction is enabled.*
PanMaximumX	FeatStd::Optional <FeatStd::Float>	The maximum pan Y value. *This property is only available if panning in y direction is enabled.*
PinchMinimumX	FeatStd::Optional <FeatStd::Float>	The minimum pinch X value. *This property is only available if pinching in x direction is enabled.*
PinchMinimumY	FeatStd::Optional <FeatStd::Float>	The minimum pinch Y value. *This property is only available if pinching in y direction is enabled.*
PinchMaximumX	FeatStd::Optional <FeatStd::Float>	The maximum pinch X value. *This property is only available if pinching in x direction is enabled.*
PinchMaximumY	FeatStd::Optional <FeatStd::Float>	The maximum pinch Y value. *This property is only available if pinching in y direction is enabled.*

PinchMinimumUniform	FeatStd::Optional <FeatStd::Float>	The minimum pinch uniform value. *This property is only available if pinching is enabled and configured for ScaleUniform.*
PinchMaximumUniform	FeatStd::Optional <FeatStd::Float>	The maximum pinch uniform value. *This property is only available if pinching is enabled and configured for ScaleUniform.*
RotateMinimum	FeatStd::Optional <FeatStd::Float>	The minimum rotate value *This property is only available if rotation is enabled.*
RotateMaximum	FeatStd::Optional <FeatStd::Float>	The maximum rotate value *This property is only available if rotation is enabled.*

Property Action Behaviors

Set Appearance

Sets or changes the appearance of a 3D node. Adjusts its material, color, transparency, or texture to update how the node looks when triggered. This behavior is available for 3D scenes only.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
PropertyReference	PropertyReference::Enum	Choose how the property gets set. Via own property or from another Behavior. *Possible values: Self, Other*
Appearance	SharedPointer<Candera::Appearance>	The appearance to set on the node. *This property is only available if the PropertyReference is set to "Self".*
PropertyProvider	PropertyProviderBehavior*	Property Behavior which contains the color which has to be set on the effect. *This property is only available if the PropertyReference is set to "Other".*
UseAsDefault	bool	True if the image shall be used as default image.

Send Value as Event

Class Reference

Sends out a new value. Associated controls with a value property receive the new value.

Example Usage

- Format String Example
- Interpolate Value Example
- Move Node Example
- Send Value As Event Example
- Set Position Example
- Set Rotation Example
- Switch Image Example
- Set Scale Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
PropertyReference	CgiStudioControl::PropertyReference::Enum	Choose how the property gets set. Via own property or from another Behavior. *Possible values: Self, Other*
ChangeValue	ChangeValue::Enum	How the event value shall be handled by the receiving. *Possible values: Absolute, Relative*
EventDispatchStrategy	EventDispatchStrategyType::Enum	The way the event gets sent. *Possible values: Direct, Broadcast, Bubble, Tunnel, DedicatedBehavior* For more information, please refer to the Event Dispatching page.

Advanced Properties

Property Name	Type	Description
Value	FeatStd::Variant	The value of the event which gets sent. *This property is only available when the PropertyReference is set to "Self".*

PropertyProvider	PropertyProviderBehavior*	Property provider which contains the value which has to be set on the send value as event action. *This property can be set in the right side of the Scene Tree panel. It is only available when the PropertyReference is set to "Other".*
DedicatedBehavior	Candera::ProcessValueBehavior*	The dedicated process value behavior that will receive the change value event. *This property can be set in the right side of the Scene Tree panel. It is only available if the EventDispatchStrategy is set to "DedicatedBehavior".*

Set Color

Class Reference

Sets the color of a node with a Color effect attached (e.g. BitmapBrushColorBlend or SolidColorBrush). In 3D, it sets the diffuse color of the material.

Supports

- Controls:
  - Text
  - Text Value
- Effects:
  - BitmapBrushColorBlend (as used in TextNode)
  - GlBitmapBrushColorMaskBlend
  - GlDropShadowBitmapBrushColorBlend
  - GlFlipBitmapBrushColorBlend
  - GlOutlineBitmapBrushColorBlend
  - SolidColorBrushColorBlend (as used in SolidColorNode)

See also: Set Image, Set Text, Set Control State

Example Usage

- Thermometer Example
- Hover - Hover Leave Example
- OnClick Example
- Set Image Example
- Start Timer Example

Common Properties

Property Name	Type	Description

Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The node which is modified. If not set, then the node where the behavior is attached to, will be modified.
Property Reference	PropertyReference::Enum	Choose how the property gets set. Via own property or from another Behavior. *Possible values: Self, Other*
Color	Candera::Color	The color to set. *This property is only available if PropertyReference is set to "Self".*
Use For Child Nodes	bool	If set to true, also child nodes are affected by this action.
Levels	UInt8	Number of levels to step down through groups. Default is zero. *This property is only available if UseForChildNodes is checked.*

Advanced Properties

Property Name	Type	Description
Property Provider	PropertyProviderBehavior*	Property Provider which contains the color which has to be set on the effect. *This property can be set in the right side of the Scene Tree panel.* *It is only available if PropertyReference is set to"Other".*

Set Image

Class Reference

Sets or changes the image of a node (such as Bitmap) with an attached effect (BitmapBrush) when an action is triggered. Can be used to update the image property of a BitmapBrush effect when used with OnClick.

See also: Set Text, Set Color, Set Control State

Example Usage

- Set Image Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.

Target Node	Candera::AbstractNodePointer	The node which gets the given image. If not set, then the node where the behavior is attached too, will be modified.
Property Reference	PropertyReference::Enum	Choose how the property gets set. Via own property or from another Behavior. *Possible values: Self, Other*
Image	SharedPointer<Image2D>	The image to set on the effect. *This property is only available for 2D scenes.*
Use As Default	bool	True if the image shall be used as default image.

Advanced Properties

Property Name	Type	Description
Texture	SharedPointer<Candera::TextureImage>	The image to set on the appearance. *This property is only available for 3D scenes.*
PropertyProvider	PropertyProviderBehavior*	Property Provider which contains the color which has to be set on the effect. *This property can be set in the right side of the Scene Tree panel.* *It is only available if PropertyReference is set to "Other"*

Set MVG Fill Color

Sets the fill color of an MVG node using an MVG brush effect (for example, MVGRectBrush) when an action behavior is triggered.

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Target Node	Candera::AbstractNodePointer	The node which gets filled with color. If not set, then the node where the behavior is attached too, will be modified.
PropertyReference	CgiStudioControl::PropertyReference::Enum	Choose how the property gets set. Via own property or from another Behavior.
Color	Candera::Color	The fill color to be set.
PropertyProvider	CgiStudioControl::PropertyProviderBehavior	Property Behavior which contains the color which has to be set on the effect. *It is only available if PropertyReference is set to "Other"*

Set MVG Stroke Color

Sets the stroke (outline/border) color of an MVG node using an effect. The color can be applied dynamically when an action behavior is triggered. Can set the stroke color of an MVGArc node when used with OnPress.

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Target Node	Candera::AbstractNodePointer	The node which gets the stroke color. If not set, then the node where the behavior is attached too, will be modified.
PropertyReference	CgiStudioControl::PropertyReference::Enum	Choose how the property gets set. Via own property or from another Behavior.
Color	Candera::Color	The fill color to be set.
PropertyProvider	CgiStudioControl::PropertyProviderBehavior	Property Behavior which contains the color which has to be set on the effect. *It is only available if PropertyReference is set to "Other"*

Render Child Nodes

Class Reference

Enables or disables child nodes of a group according to the operation mode. The same float input is used to select a single node or affect multiple nodes. When the action triggred.

Example Usage

- Render Child Nodes Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The node which switches child nodes. If not set, then the node where the behavior is attached to, will modified its children.
Index	Float	Index of the child node which shall be active.
OperationMode	OperationMode::Enum	When set to MultipleNodes, EnableChildNodes will enable the first N children of Group Node. When Operation Mode is set to SingleNode, only the Nth node is enabled. *Possible values: MultipleNodes, SingleNode*

Levels	UInt8	Number of levels to step down through groups. Default is zero.
NormalizeValue	bool	If set to true, the received value has to be between 0 and 1 and will be mapped to the number of children. Otherwise the value has to be the index of the child.

Set Text

Class Reference

Sets/Changes the text of a TextNode.

See also: Set Image, Set Color, Set Control State

Example Usage

- Set Text Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which gets the given text. If not set, then the node where the behavior is attached to, will be modified.
PropertyReference	PropertyReference::Enum	Choose how the property gets set. Via own property or from another Behavior. *Possible values: Self, Other*
Text	String	The text that will be set to the Text Node. *This property is available if the PropertyReference is set to "Self".*

Advanced Properties

Property Name	Type	Description
PropertyProvider	PropertyProviderBehavior*	Property Behavior which contains the color which has to be set on the effect. *This property can be set in the right side of the Scene Tree panel.* *It is only available if the PropertyReference is configured with "Other".*

Set Text Style

Sets the configured text style to a text node when the action is triggered.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which gets the given text. If not set, then the node where the behavior is attached to, will be modified.
Style	SharedPointer<TextRendering::SharedStyle>	The Style that will be set to the Text Node.

Enable Rendering

Class Reference

If invoked by a TriggerBehavior, changes the visibility of a node.

Example Usage

- Enable Rendering Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The node which gets the visibility flag. If not set, then the node where the behavior is attached to, will be modified.
Visible	bool	According to this value, the affected node gets rendered or not.

Send Value Changed As Event

Sends out a new value. Associated controls with a value property receive the new value.

Example Usage

- Process Value Changed Event - State Machine Example
- Usecase for Send Value Changed as Event and reception in state machine(s)

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Value	FeatStd::Variant	The value of the event which gets sent. *This property is only available if PropertyReference is set to "Self".*
Sender	Candera::AbstractNodePointer	The node for which the event shall be emitted. If it is not set then the TargetNode or Node (if TargetNode is not set) will be taken instead.
EventDispatchStrategy	EventDispatchStrategyType::Enum	The way the event gets sent. *Possible values: Direct, Broadcast, Bubble, Tunnel, DedicatedBehavior* For more information, please refer to the Event Dispatching page.
PropertyReference	PropertyReference::Enum	Choose how the property gets set. Via own property or from another Behavior. *Possible values: Self, Other*
Target Node	Candera::AbstractNodePointer	The target node to which the events are routed from the source node. If this property is not set, the event will be routed to the node of the behavior.

Advanced Properties

Property Name	Type	Description
DedicatedBehavior	ProcessValueBehavior*	The dedicated process value behavior that will receive the change value event. *This property can be set in the right side of the Scene Tree panel.* *It is only available if the EventDispatchStrategy is set to "DedicatedBehavior".*
PropertyProvider	PropertyProviderBehavior*	Property Behavior which contains the color which has to be set on the effect. *This property can be set in the right side of the Scene Tree panel.* *It is only available if the PropertyReference is set to "Other".*

Scroll Action Behaviors

Send Scrollable Event

Class Reference

Enables a scrollable behavior (for example, a List behavior) to scroll to a specific target position based on the selected scrolling type (horizontal or vertical) and value (item-wise or page-wise) when triggered. Can be used to manage the scroll direction of a list control.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Target Node	Candera::AbstractNodePointer	The node which is modified. If not set, then the node where the behavior is attached to, will be modified.
Direction	Optional<ScrollDirection::Enum>	The direction in which the scroll should be performed. *Possible values: Vertical, Horizontal*
Type	ScrollableEventType::Enum	The scrolling type. It can be either relative or absolute scrolling in combination of being either itemwise or pagewise scrolling.
Value	Float	The number of items or pages scrolled to.
TreatAnimationDurationZeroAsNotSet	bool	When the AnimationDuration is set to 0 it will not be used as a valid animation duration. If a immediate scroll to the target position is intended this flag has to be set to false.
AnimationDuration	UInt32	Duration of the scrolling animation.
EventDispatchStrategy	EventDispatchStrategyType::Enum	The way the event gets sent. *Possible values: Direct, Broadcast, Bubble, Tunnel, DedicatedBehavior* For more information, please refer to the Event Dispatching page.

Advanced Properties

Property Name	Type	Description
DedicatedBehavior	Behavior*	The dedicated process value behavior that will receive the change value event. *This property can be set in the right side of the Scene Tree panel.* *It is only available if the EventDispatchStrategy is set to "DedicatedBehavior".*

Streaming Action Behaviors

Configure and Play Stream

Class Reference

Starts, stops, pauses, resumes video streams by emitting an VideoControl Event. These Events can be received by Video Stream Behaviors.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The node which receives the event. If not set, then the node where the behavior is attached to, will receive it.
Action	VideoControl::Enum	Action that will be executed on the stream. *Possible values: Play, Replay, Pause, Resume, Stop.*
EventDispatchStrategy	EventDispatchStrategyType::Enum	The strategy in which the event is sent. *Possible values: Direct, Broadcast, Bubble, Tunnel, DedicatedBehavior.* For more information, please refer to the Event Dispatching page.

Advanced Properties

Property Name	Type	Description
DedicatedBehavior	Candera::Behavior*	A dedicated behavior that will receive the event. *This property can be set in the right side of the Scene Tree panel.* *It is only available if the EventDispatchStrategy is set to "DedicatedBehavior".*

Transition Action Behaviors

Perform Transition Requests

Class Reference

Used to specify whether a node should be activated or deactivated by the transition framework when the associated TransitionActionBehavior is triggered.

Used to execute transitions via TransitionRequestBehaviors.

See also: Transition Request

Example Usage

- Perform Transition Requests Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Requests	ArrayProperty<TransitionRequestBehavior*>	The first TransitionRequestBehavior in the request-set.
Variant	String	Default Variant which will be used for attached Requests. Not that Requests can also have their own Variant.

Transition Request

Class Reference

Used to specify whether a node should be activated or deactivated by the transition framework when the associated TransitionActionBehavior is triggered.

Example Usage

- Perform Transition Requests Example

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
RequestType	CgiStudioControl::TransitionRequestType::Enum	Whether a node should be Activated or Deactivated by the transition. Currently running transitions on a node can be ended using the Finish transition type. *Possible values: Activate, Deactivate, Finish*
Variant	FeatStd::String	Variant which will be used for this Request.
Identifier	Candera::Transitions::Identifier	The Node or Scene that will be subject to the transition.

Views

Jump to View

Use this action to request jump/switching to the configured view (via the configured node). Depending on the view state setup (e.g. with view trees or a state machine) the request will be accepted or denied. The action itself will emit a JumpToViewEvent on the configured node.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
TargetNode	Candera::AbstractNodePointer	The Node which is modified. If not set, then the node where the behavior is attached to, will be modified.
View	ViewIdentifier	Identifier of the view that is requested to be shown.

> There is **no documentation** for behaviors of the **Internal** subcategory as they are used internally by Scene Composer and **not intended for customer use**.

Send Virtual Key

When triggered the action behavior sends a KeyEvent for down and one frame later a KeyEvent for up

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
KeyType	KeyType::Enum	Choose the type of the key. With KeyCode you can select any key. *Possible values: KeyCode, Modifier*
Modifier	KeyModifierEnum::Enum	Modifier Key (e.g. Shift) *Possible values: None, Key_Shift, Key_Alt, Key_Ctrl, Key_CapsLock, Key_Num* *This property is only available if "KeyType" is set to "Modifier".*
KeyCode	KeyboardKeyCode::Enum	With KeyCode you can select any key. *Possible values: Key_None, ...<all keyboard key codes>... Key_OEM8* *This property is only available if "KeyType" is set to "KeyCode".*

Set Culture

Class Reference

Action Behavior which activates a culture based on its 'locale' value. Receives a locale as value. Invoking the action changes the application culture to a culture fitting to this locale.

Common Properties

Property Name	Type	Description
Asynchronous	bool	Set this flag to true to perform the action after all other actions have been performed.
Locale	FeatStd::String	Locale of a culture. Invoking the action will set the culture active.

Revision #68
Created 2023-02-27 05:44:16 UTC by Kanai Tomoaki
Updated 2026-07-29 00:21:08 UTC by Tsuyoshi.Kato