



# Widget and Widget properties reference

October 20, 2023



REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Create the WidgetReference.adoc</b>	<b>2</b>
<b>3</b>	<b>Add test scope / ignore comments</b>	<b>3</b>
<b>4</b>	<b>Update the WidgetReference.adoc</b>	<b>4</b>
<b>5</b>	<b>AngularSliderWidget2D</b>	<b>5</b>
5.1	Property list . . . . .	5
5.2	Message list . . . . .	9
<b>6</b>	<b>AnimationControlWidget2D</b>	<b>10</b>
6.1	Property list . . . . .	10
6.2	Message list . . . . .	13
<b>7</b>	<b>AnimationWidget2D</b>	<b>14</b>
7.1	Property list . . . . .	14
7.2	Message list . . . . .	15
<b>8</b>	<b>AnimationWidget3D</b>	<b>17</b>
8.1	Property list . . . . .	17
8.2	Message list . . . . .	17
<b>9</b>	<b>ArabicLayouterPatchWidget</b>	<b>18</b>
9.1	Property list . . . . .	18
9.2	Message list . . . . .	18
<b>10</b>	<b>BaseLineLayouterWidget2D</b>	<b>19</b>
10.1	Property list . . . . .	19
10.2	Message list . . . . .	20
<b>11</b>	<b>BlurWidget2D</b>	<b>21</b>
11.1	Property list . . . . .	21
11.2	Message list . . . . .	22

<b>12 BlurWidget3D</b>	<b>23</b>
12.1 Property list	23
12.2 Message list	24
<b>13 ButtonAnimationWidget2D</b>	<b>25</b>
13.1 Property list	25
13.2 Message list	29
<b>14 ButtonGroupWidget2D</b>	<b>30</b>
14.1 Property list	30
14.2 Message list	34
<b>15 ButtonImageWidget2D</b>	<b>35</b>
15.1 Property list	35
15.2 Message list	37
<b>16 ButtonTextColorWidget2D</b>	<b>38</b>
16.1 Property list	38
16.2 Message list	41
<b>17 ButtonTextScrollWidget2D</b>	<b>42</b>
17.1 Property list	42
17.2 Message list	43
<b>18 ButtonWidget2D</b>	<b>44</b>
18.1 Property list	44
18.2 Message list	48
<b>19 CameraControlWidget2D</b>	<b>50</b>
19.1 Property list	50
19.2 Message list	51
<b>20 CanvasLayouterWidget2D</b>	<b>52</b>
20.1 Property list	52
20.2 Message list	53
<b>21 ClockWidget2D</b>	<b>54</b>
21.1 Property list	54
21.2 Message list	55
<b>22 CollapseWidget2D</b>	<b>57</b>
22.1 Property list	57
22.2 Message list	58

<b>23 ColorBarWidget2D</b>	<b>59</b>
23.1 Property list . . . . .	59
23.2 Message list . . . . .	60
<b>24 ColorEffectWidget2D</b>	<b>61</b>
24.1 Property list . . . . .	61
24.2 Message list . . . . .	62
<b>25 ColorSwitchWidget2D</b>	<b>63</b>
25.1 Property list . . . . .	63
25.2 Message list . . . . .	65
<b>26 ComboBoxWidget2D</b>	<b>66</b>
26.1 Property list . . . . .	66
26.2 Message list . . . . .	69
<b>27 ControlTemplateWidget2D</b>	<b>70</b>
27.1 Property list . . . . .	70
27.2 Message list . . . . .	71
<b>28 CoverflowListWidget2D</b>	<b>72</b>
28.1 Property list . . . . .	72
28.2 Message list . . . . .	81
<b>29 DelegateWidget2D</b>	<b>82</b>
29.1 Property list . . . . .	82
29.2 Message list . . . . .	85
<b>30 DirectTextureConsumer2D</b>	<b>86</b>
30.1 Property list . . . . .	86
30.2 Message list . . . . .	89
<b>31 DirectTextureConsumer3D</b>	<b>90</b>
31.1 Property list . . . . .	90
31.2 Message list . . . . .	90
<b>32 DirectTextureProvider2D</b>	<b>91</b>
32.1 Property list . . . . .	91
32.2 Message list . . . . .	92
<b>33 DirectTextureProvider3D</b>	<b>93</b>
33.1 Property list . . . . .	93
33.2 Message list . . . . .	93

<b>34 DockSideWidget2D</b>	<b>94</b>
34.1 Property list	94
34.2 Message list	95
<b>35 DropDownListWidget2D</b>	<b>96</b>
35.1 Property list	96
35.2 Message list	105
<b>36 EditFieldWidget2D</b>	<b>106</b>
36.1 Property list	106
36.2 Message list	112
<b>37 EffectControlWidget2D</b>	<b>114</b>
37.1 Property list	114
37.2 Message list	117
<b>38 EnableGroupWidget2D</b>	<b>118</b>
38.1 Property list	118
38.2 Message list	119
<b>39 FlexBaseLineLayouterWidget2D</b>	<b>120</b>
39.1 Property list	120
39.2 Message list	121
<b>40 FlexCanvasLayouterWidget2D</b>	<b>122</b>
40.1 Property list	122
40.2 Message list	123
<b>41 FlexCollapseWidget2D</b>	<b>124</b>
41.1 Property list	124
41.2 Message list	125
<b>42 FlexDockSideWidget2D</b>	<b>126</b>
42.1 Property list	126
42.2 Message list	127
<b>43 FlexDropdownListWidget2D</b>	<b>128</b>
43.1 Property list	128
43.2 Message list	141
<b>44 FlexListWidget2D</b>	<b>142</b>
44.1 Property list	142
44.2 Message list	155

<b>45 FlexOverlayLayouterWidget2D</b>	<b>156</b>
45.1 Property list	156
45.2 Message list	157
<b>46 FlexScrollBar2D</b>	<b>158</b>
46.1 Property list	158
46.2 Message list	163
<b>47 FlexScrollBarButton2D</b>	<b>164</b>
47.1 Property list	164
47.2 Message list	169
<b>48 FlexSizeWidget2D</b>	<b>170</b>
48.1 Property list	170
48.2 Message list	171
<b>49 FlexSwitchWidget2D</b>	<b>172</b>
49.1 Property list	172
49.2 Message list	173
<b>50 FlexToggleWidget2D</b>	<b>174</b>
50.1 Property list	174
50.2 Message list	175
<b>51 FocusGroupWidget2D</b>	<b>176</b>
51.1 Property list	176
51.2 Message list	178
<b>52 GestureWidget2D</b>	<b>179</b>
52.1 Property list	179
52.2 Message list	182
<b>53 GizmoWidget2D</b>	<b>185</b>
53.1 Property list	185
53.2 Message list	190
<b>54 GridAutoArrangeWidget2D</b>	<b>192</b>
54.1 Property list	192
54.2 Message list	193
<b>55 HandWriting_Util_Widget2D</b>	<b>194</b>
55.1 Property list	194
55.2 Message list	199

<b>56 ImageEffectWidget2D</b>	<b>200</b>
56.1 Property list	200
56.2 Message list	201
<b>57 ImageSwitchWidget2D</b>	<b>202</b>
57.1 Property list	202
57.2 Message list	208
<b>58 LabelWidget2D</b>	<b>209</b>
58.1 Property list	209
58.2 Message list	210
<b>59 LineListWidget3D</b>	<b>211</b>
59.1 Property list	211
59.2 Message list	211
<b>60 ListBindingWidget2D</b>	<b>212</b>
60.1 Property list	212
60.2 Message list	213
<b>61 ListFixedPageMovementGapWidget2D</b>	<b>214</b>
61.1 Property list	214
61.2 Message list	215
<b>62 ListItemAnimationMarkerWidget2D</b>	<b>216</b>
62.1 Property list	216
62.2 Message list	217
<b>63 ListMovementAnimationsProviderWidget2D</b>	<b>218</b>
63.1 Property list	218
63.2 Message list	219
<b>64 ListOverscrollWidget2D</b>	<b>220</b>
64.1 Property list	220
64.2 Message list	221
<b>65 ListScrollAnchorWidget2D</b>	<b>222</b>
65.1 Property list	222
65.2 Message list	223
<b>66 ListSpeedAnimationWidget2D</b>	<b>224</b>
66.1 Property list	224
66.2 Message list	225



<b>67 ListWidget2D</b>	<b>226</b>
67.1 Property list . . . . .	226
67.2 Message list . . . . .	236
<b>68 MarginWidget2D</b>	<b>244</b>
68.1 Property list . . . . .	244
68.2 Message list . . . . .	245
<b>69 MarkerWidget2D</b>	<b>246</b>
69.1 Property list . . . . .	246
69.2 Message list . . . . .	247
<b>70 MeshWidget2D</b>	<b>248</b>
70.1 Property list . . . . .	248
70.2 Message list . . . . .	250
<b>71 MeterWidget2D</b>	<b>251</b>
71.1 Property list . . . . .	251
71.2 Message list . . . . .	252
<b>72 MultiSliderHelperWidget3D</b>	<b>253</b>
72.1 Property list . . . . .	253
72.2 Message list . . . . .	253
<b>73 MultiSliderWidget2D</b>	<b>254</b>
73.1 Property list . . . . .	254
73.2 Message list . . . . .	257
<b>74 NodeControlWidget2D</b>	<b>258</b>
74.1 Property list . . . . .	258
74.2 Message list . . . . .	261
<b>75 OPSWidget2D</b>	<b>262</b>
75.1 Property list . . . . .	262
75.2 Message list . . . . .	263
<b>76 OffscreenTouchProxyWidget2D</b>	<b>264</b>
76.1 Property list . . . . .	264
76.2 Message list . . . . .	268
<b>77 OverlayLayouterWidget2D</b>	<b>269</b>
77.1 Property list . . . . .	269
77.2 Message list . . . . .	270

<b>78 PageEditWidget2D</b>	<b>271</b>
78.1 Property list . . . . .	271
78.2 Message list . . . . .	274
<b>79 PanelSlideWidget2D</b>	<b>276</b>
79.1 Property list . . . . .	276
79.2 Message list . . . . .	279
<b>80 ProgressBarWidget2D</b>	<b>281</b>
80.1 Property list . . . . .	281
80.2 Message list . . . . .	282
<b>81 RichTextHighlightWidget2D</b>	<b>283</b>
81.1 Property list . . . . .	283
81.2 Message list . . . . .	284
<b>82 RichTextMarqueeWidget2D</b>	<b>285</b>
82.1 Property list . . . . .	285
82.2 Message list . . . . .	286
<b>83 RichTextWidget2D</b>	<b>288</b>
83.1 Property list . . . . .	288
83.2 Message list . . . . .	292
<b>84 ScopeWidget2D</b>	<b>293</b>
84.1 Property list . . . . .	293
84.2 Message list . . . . .	294
<b>85 ScrollBarButtonWidget2D</b>	<b>295</b>
85.1 Property list . . . . .	295
85.2 Message list . . . . .	299
<b>86 ScrollBarWidget2D</b>	<b>300</b>
86.1 Property list . . . . .	300
86.2 Message list . . . . .	305
<b>87 ScrollableRichTextWidget2D</b>	<b>306</b>
87.1 Property list . . . . .	306
87.2 Message list . . . . .	309
<b>88 ScrollableTextWidget2D</b>	<b>311</b>
88.1 Property list . . . . .	311
88.2 Message list . . . . .	317

<b>89 SimpleButtonWidget3D</b>	<b>318</b>
89.1 Property list . . . . .	318
89.2 Message list . . . . .	319
<b>90 SizeWidget2D</b>	<b>320</b>
90.1 Property list . . . . .	320
90.2 Message list . . . . .	321
<b>91 SliderWidget2D</b>	<b>322</b>
91.1 Property list . . . . .	322
91.2 Message list . . . . .	326
<b>92 SolidColorEffectWidget2D</b>	<b>327</b>
92.1 Property list . . . . .	327
92.2 Message list . . . . .	328
<b>93 SpellerWidget2D</b>	<b>329</b>
93.1 Property list . . . . .	329
93.2 Message list . . . . .	333
<b>94 StepAnimationWidget2D</b>	<b>335</b>
94.1 Property list . . . . .	335
94.2 Message list . . . . .	337
<b>95 SurfaceInputRegionWidget2D</b>	<b>338</b>
95.1 Property list . . . . .	338
95.2 Message list . . . . .	339
<b>96 SurfaceRegionWidget2D</b>	<b>340</b>
96.1 Property list . . . . .	340
96.2 Message list . . . . .	341
<b>97 SwitchWidget2D</b>	<b>342</b>
97.1 Property list . . . . .	342
97.2 Message list . . . . .	343
<b>98 TextAreaWidget2D</b>	<b>344</b>
98.1 Property list . . . . .	344
98.2 Message list . . . . .	351
<b>99 TextBaseLineOffsetWidget2D</b>	<b>352</b>
99.1 Property list . . . . .	352
99.2 Message list . . . . .	353

<b>100TextColorWidget2D</b>	<b>354</b>
100.1Property list . . . . .	354
100.2Message list . . . . .	355
<b>101TextExtensionWidget2D</b>	<b>356</b>
101.1Property list . . . . .	356
101.2Message list . . . . .	357
<b>102TextHighlightWidget2D</b>	<b>358</b>
102.1Property list . . . . .	358
102.2Message list . . . . .	359
<b>103TextWaitAnimationWidget2D</b>	<b>360</b>
103.1Property list . . . . .	360
103.2Message list . . . . .	361
<b>104TextWidget2D</b>	<b>362</b>
104.1Property list . . . . .	362
104.2Message list . . . . .	367
<b>105TextWidget2DV2</b>	<b>368</b>
105.1Property list . . . . .	368
105.2Message list . . . . .	370
<b>106TextureImageWidget3D</b>	<b>372</b>
106.1Property list . . . . .	372
106.2Message list . . . . .	372
<b>107TimerWidget2D</b>	<b>373</b>
107.1Property list . . . . .	373
107.2Message list . . . . .	374
<b>108ToggleWidget2D</b>	<b>375</b>
108.1Property list . . . . .	375
108.2Message list . . . . .	376
<b>109TwoDimensionSliderWidget2D</b>	<b>377</b>
109.1Property list . . . . .	377
109.2Message list . . . . .	381
<b>110ViewSwitchWidget2D</b>	<b>382</b>
110.1Property list . . . . .	382
110.2Message list . . . . .	383

Automatically generated from CGI Studio xml file MetaInfo\_0\_test.xml.

# Chapter 1

## Introduction

- The reference document provides a list of all Widgets and of all properties and all messages of these Widgets. (includes each type, bindable property, description, test link and ignore comment if have)
- The reference document is nearly fully generated from the CGI XML file.
- The reference document will be included into the widget document.
- All the information and the update should be collected and verified by the FeatureTrace analysis.
- The extracted details in either of these files is required to be checked: [HMI Base Feature Trace](#) and [Widget Property Information from HMI Base Widget Guide Documentation](#)

## Chapter 2

# Create the WidgetReference.adoc

- Check path to CGI Studio generated XML file with information of all Widgets.
- Scan CGI Studio generated XML file for Widget information.  
(e.g., all properties of all Widgets with additional details)
- Collect information from existing WidgetReference.adoc.
- Create Widget Reference adoc file.
- Path of script: \tools\ci\_tools\rbcm-jen-hmib\featureTracking\WidRefGen\.

## Chapter 3

# Add test scope / ignore comments

- The test scope link should be added/updated to the document when the creation of TML test case is completed.
- The ignored status should be updated in the column test scope where the feature/property could not be tested.



## Chapter 4

# Update the WidgetReference.adoc

- Using the AddLinkWidRef script to automatically remove old and add new test scope links to WidgetReference.adoc.
- Update the test plan and documents dir path in Configure file at the same path level.
- Open cmd at the current dir then run the command: `python AddLinkWidRef.py`.
- Path of script for updating the test scope: `\views\cmd_super\ai_hmi_cgi\test\HMIBase_TML\TestCases\_scripts\TMLDocGeneratorSc`
- Path of script for updating the ignored status: `\views\cmd_super\ai_hmi_cgi\test\HMIBase_TML\TestCases\_scripts\TMLDocGenerat`

## Chapter 5

# AngularSliderWidget2D

**Name:** AngularSliderWidget2D

**Description:** The slider is the widget for controlling a value by positioning a knob / thumb in a legal range.

**Category:** Range

### 5.1 Property list

Name	Bindable	Type	Description	Test Scope
AngularSliderType	False	Enum	Type of angular slider	
AnimationDuration	False	UShort	Animation duration for which the marker node rotation to be animated	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
BitmapFillNode	False	Node2D	Node to be used for the bitmap corresponding to the current value.	
ContinuousUpdateOnDrag	False	Bool	Set to true if value updates are required while dragging, set to false if update is required only at the end of drag.	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
CurrentValue	True	Float	Current value of the slider marker, should be in the range between MinVal and MaxVal.	TC_W02_01

DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	TC_W02_08
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W02_08
EnableAnimation	False	Bool	Enable animation for marker node	
FillerPositionIsKnobCenter	False	Bool	Filler starts at center of marker, for example round knob.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	

Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTF is simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
MarkerMovement	False	Enum	Defines the way the slider marker moves on tap or drag. For TimedStepMovement the marker will move one step on Press and will keep moving one step for every Repeat, this means that PressRepeat needs to be enabled and Drag needs to be disabled to prevent jumping on Drag.	
MarkerNode	False	Node2D	The Scene Tree Node that supports a bitmap used for the slider marker (Knob, Thumb).	
MarkerNormalBitmap	False	UInt	DEPRECATED! Use the appearance concept to change bitmaps based on state.	
MarkerSelectedBitmap	False	UInt	DEPRECATED! Use the appearance concept to change bitmaps based on state.	
MaxVal	True	Float	Maximum value for slider at last marker (e.g Tuner 108.0 Mhz)	TC_W02_04
MaximumAngle	False	Float	Angle of the Indicator to represent the maximum value	
MinVal	True	Float	Minimum value for slider at first marker (e.g Tuner 88.0 Mhz)	TC_W02_03
MinimumAngle	False	Float	Angle of the Indicator to represent the minimum value	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NumOfSteps	True	UShort	Total number steps for the marker movement. The stepsize will be calculated based on given range and number of steps. (e.g. range 0..20, steps=10 -> stepsize 2)	TC_W02_02

PaddingMaxVal	False	Float	Maximum graphical padding value on right hand, offset in pixel to the last valid marker for maximum value (calculated from the end of the background scale bitmap)	
PaddingMaximumAngle	False	Float	Maximum padding angle for which the marker will be positioned for the maximum value	
PaddingMinVal	False	Float	Minimum graphical padding value on left hand, offset in pixel to the first valid marker for minimum value (calculated from the beginning of the background scale bitmap).	
PaddingMinimumAngle	False	Float	Minimum padding angle for which the marker will be positioned for the minimum value	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
RotationType	False	Enum	Type of angular slider rotation	
SliderBackGround	False	Node2D	The Scene Tree Node that renders sliders background grid with scale markers on top.	
SliderOrientation	False	Enum	Defines the orientation of the slider marker movement and the slider bar.	
StepSize	True	Float	Defines the stepsize for movement of slidermarker.	TC_W02_02

Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W02_08
TouchableRadius	False	Float	Distance from the center of the angular slider to the required touchable region	
TouchableRadiusOffset	False	Float	Offset from the Touchable radius to be considered as touchable region	
UseNumofSteps	False	Bool	If this property is set to true NumOfSteps will be visible and used, else StepSize will be visible and used.	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W02_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 5.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
------	--------------	-------------	---------	--------------	------------

## Chapter 6

# AnimationControlWidget2D

**Name:** AnimationControlWidget2D

**Description:** Provides support to control (configure, start, stop) an animation. The animation can be defined in SceneComposer or created internally by this widget. Also provides support to add an animated property to the animation.

**Category:** Animation

### 6.1 Property list

Name	Bindable	Type	Description	Test Scope
AnimatedPropertyEnabled	True	Bool	Adds an animated property for the associated node to the animation.	TC_W55_17
Animation	False	AnimationBase	Animation created in SceneComposer which is used as external source.	
AnimationPlayerSource	False	Enum	If source is internal then the animation will be created and destroyed by this widget, otherwise an animation created in SceneComposer must be used.	
Channels	True	Enum	Configures the animation channels based on the animated property type. By default all the members of the property type will be animated but this allows to animate only the member X or Y from a Vector2. Or just the member Top from a Margin. Ch_0 - X, Red, Left; Ch_1 - Y, Green, Top; Ch_2 - Z, Blue, Width, Right; Ch_3 - W, Alpha, Height, Bottom;	TC_W55_01
CurrentFirstKeyframeEnabled	True	Bool	When enabled, the value(s) of the first keyframe are taken from the node when the animation is started.	TC_W55_10
Direction	True	Enum	Specifies if the animation is played forwards or backwards.	TC_W55_04
DirectionEnabled	False	Bool	Enables the configuration of Direction parameter for external animations.	

Duration	True	Int	When that duration has elapsed, playback will stop or act according to the repeat settings.	TC_W55_13
DurationEnabled	False	Bool	Enables the configuration of Duration parameter for external animations.	
EaseParams	True	custom://String	Not supported yet! Additional parameters required for ease interpolation strategy.	Ignored: This property is not supported yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing for this widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property is dependent on EnableGroupWidget2D, so it will be tested in EnableGroupWidget2D section.
InterpolationStrategy	True	Enum	Interpolation function which calculates the values between keyframes.	TC_W55_18
KeyframeCount	True	UShort	Number of keyframes (it needs to be at least 2).	TC_W55_03
KeyframeValues	True	Float	Values of the keyframes. Depending on the type and the channel configuration of the animated property the number of values for each keyframe is between 1 and 4. 1 value is required for simple types (bool, int, float); 2 values are required for Vector2; 3 values are required for Vector3; 4 values are required for Color, Rectangle and Margin; For example if it is required to animate a property of type Vector2 then 2 values will be required for each keyframe (values at positions 0 and 1 are for the first keyframe and so on).	TC_W55_02
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	



Property	True	Enum	Node property to be animated. Also allows to specify that a widget property should be animated. Only properties of the following types can be animated: bool, int, float, Vector2, Vector3, Rectangle, Color and Margin.	TC_W55_12
RepeatCount	True	UInt	Configures how many times the animation is executed when started using Courier messages. RepeatCount has no effect if the animation is started using ShouldRun property.	
RepeatCountEnabled	False	Bool	Enables the configuration of RepeatCount parameter for external animations.	
RepeatMode	True	Enum	When the animation is played multiple times, it can be played from the beginning for each iteration or it can bounce forwards and backwards.	TC_W55_05
RepeatModeEnabled	False	Bool	Enables the configuration of RepeatMode parameter for external animations.	
SequenceTimes	True	Int	Sequence time of the keyframes.	TC_W55_14
ShouldRun	True	Bool	The animation will run when the value of this property is true. This property is available only if the value of StartMode property is Message.	TC_W55_06
SpeedFactor	True	Float	Increases or decreases the animation speed.	TC_W55_07
SpeedFactorEnabled	False	Bool	Enables the configuration of SpeedFactor parameter for external animations.	
StartMode	False	Enum	Configures how the animation is started. StartMode as Message is only supported for external animations and they will be started by posting Courier::AnimationReqMsg messages from the application. If StartMode is Property then the animation will be started based on ShouldRun property.	
StartTime	True	Int	The point in sequence time where the animation playback shall start.	TC_W55_15
StartTimeEnabled	False	Bool	Enables the configuration of StartTime parameter for external animations.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W55_09

VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
Widget	False	Widget	Widget to be animated.	
WidgetProperty	True	custom://Widget	Widget property to be animated. Only properties of the following types can be animated: bool, int, float, Vector2, Vector3, Rectangle, Color and Margin.	TC_W55_11

6.2 Message list

Name	Subscribers	Description	Members	Distribution	Test Scope
------	-------------	-------------	---------	--------------	------------

## Chapter 7

# AnimationWidget2D

**Name:** AnimationWidget2D

**Description:** Provides support to create simple animations.

**Category:** Animation

### 7.1 Property list

Name	Bindable	Type	Description	Test Scope
Amplitude	False	Float	Amplitude : Courier::Float	
BounceCount	False	Byte	BounceCount : Courier::UInt8	
Channel1	True	custom://FirstChannel	FirstChannel	TC_W01_03
Channel2	True	custom://SecondChannel	SecondChannel	TC_W01_03
Direction	True	Enum	Direction of the Animation	TC_W01_01
EaseDirection	False	Enum	Direction of Ease Interpolation	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing for this widget.
Exponent	False	Float	Exponent : Courier::Float	
ExponentElastic	False	Float	ExponentElastic : Courier::Float	
FunctionType	False	Enum	FunctionType of Ease Interpolation	

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property is dependent on EnableGroupWidget2D, so it will be tested in EnableGroupWidget2D section.
InterpolationStrategy	False	Enum	Type of Interpolation used for Animation	
KeyFrameCount	False	UInt	Number of Key Frames	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OscillationCount	False	Byte	OscillationCount : Courier::UInt8	
Power	False	Byte	Power : Courier::UInt8	
PropertyAnimation	False	Enum	Animation Type	
RepeatTimes	False	UInt	Number of times the Animation has to repeat	
RestitutionCoefficient	False	Float	RestitutionCoefficient : Courier::Float	
SequenceTime	True	custom://String	SequenceTime	TC_W01_04
ShouldRun	True	Bool	Auto Start the Animation when the view is loaded	TC_W01_02
SpeedFactor	False	Float	Speed of Animation	
ToggleDirection	False	Bool	Change the Animation Direction after completion of Animation	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W01_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
WidgetPropertyNameToAnimate	False	custom://String	Provide the name of the widget property which has to be animated	
WidgetToAnimate	False	Widget	Associate a widget whose property is to be animated	

## 7.2 Message list

Name	Subscriber	Description	Members	Distribution	Test Scope
------	------------	-------------	---------	--------------	------------

AnimationAbortIndMsg	Widget, View, Controller	AnimationAbortIndMsg is posted by the AnimationWidget2D, each time when the animation is stopped before the completion of one animation cycle, or when the scene is destroyed (switch to another view) before the animation ends.		sequential	TC_W01_12
AnimationWidgetIndMsg	Widget, View, Controller	AnimationWidgetIndMsg is posted by the AnimationWidget2D, each time when the animation is completed.	IterationCount: Courier::Int32. +	sequential	TC_W01_11
AnimationWidgetReqMsg	Widget, View	The Animation can be started/stopped by posting the AnimationWidgetReqMsg.	AnimationAction: Courier::AnimationAction::Enum. +	sequential	TC_W01_11

## Chapter 8

# AnimationWidget3D

**Name:** AnimationWidget3D

**Description:** Provides support to create simple 3D animations

**Category:** Animation

### 8.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Duration	True	UInt	Set the duration of the animation	
Enabled	False	Bool	Enabled: Enable or disable the widget	
Name	False	CharArray	The name of the widget instance	
Node	False	Node3D	The associated node of the widget.	
PropertyAnimation	False	Enum	Animation Type	
TargetColor	True	Color	Set the target color of animation	
TargetValue	True	Vector3	Set the target value of animation	

### 8.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	-------------------	--------------------	----------------	---------------------	-------------------

## Chapter 9

# ArabicLayouterPatchWidget

**Name:** ArabicLayouterPatchWidget

**Description:** ArabicLayouterPatchWidget

**Category:** Layout

### 9.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enabled	False	Bool	The ArabicLayouterPatch is enabled or disabled for this scene. Last widget in the scene will set the final value.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

### 9.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 10

# BaseLineLayouterWidget2D

**Name:** BaseLineLayouterWidget2D

**Description:** Sets a baseline layouter and the baseline offset on a composite node or a normal group. Associate the widget directly to the targeted composite or group. Don't use the nested level because it can create confusion and lead to errors.

**Category:** Layout

### 10.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
BaseLineOffset	True	Float	Baseline offset used by the layouter.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	



NestedLevel	True	UInt	DEPRECATED: This property was required in the past when it was not possible to associate a widget directly to the composite. The value of this property represents the ancestor's level, relative to the associated node, that will be targeted by this widget. This property should not be used because it creates confusion and can lead to errors. Just associate the widget directly to the targeted composite or group.	
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 10.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 11

# BlurWidget2D

**Name:** BlurWidget2D

**Description:** BlurWidget2D widget

**Category:** Image

### 11.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
BitmapMask	True	Bitmap	Sets the bitmap, which is used as mask for blur.	TC_W56_05
BlendOutput	True	Bool	Which Rendermode the Blur Output camera will be using	
BlurEnabled	True	Bool	Whether or not blurring is enabled.	TC_W56_03
BlurMaskVariant	True	Enum	Sets the selected shader for blur	Ignored: Not tested because this property can not be set dynamically. But tested statically through the initial screenshot.
BlurRadius	True	Int	Sets the radius of the blur.	TC_W56_02
BlurRadiusToSigma	True	Float	Sets the factor to calculate sigma from the blur radius. Typical values are between 0.5 and 0.33.	TC_W56_04
BlurShaderVariant	False	Enum	Sets the selected shader for blur	
CombineMask	False	Bool	Sets flag to determine if the mask render target is combined with the blurred scene.	
Darken	True	Float	Sets the darkening factor for the shader	TC_W56_01
DirtyAreaRectangle	False	Rectangle	Sets the rectangle which is set as dirty area for blurring.	

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: Not tested because this property is inherited and not specific to this Widget..
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
KeepFrameBufferContent	False	Bool	Sets flag to determine if the content outside the blur mask is taken from the input framebuffer or discarded.	
Multipass	False	Bool	Sets single pass or multi pass blur shader.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
RenderNodeMask	False	Node2D	Sets the node, that defines the position of the bitmap used as mask for blur.	
RenderTargetMask	False	RenderTarget	Sets the render target that is used as mask for blur.	
SubsamplingFactor	False	Int	Sets the subsampling factor for the blurred render target.	
UseManualDirtyArea	False	Bool	Sets flag that determines, if a manually defined dirty area is to be used.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: Not tested because this property is inherited and not specific to this Widget..
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 11.2 Message list

Name	Subscribers	Description	Members	Distribution Scope
------	-------------	-------------	---------	--------------------

## Chapter 12

# BlurWidget3D

**Name:** BlurWidget3D

**Description:** BlurWidget3D widget

**Category:** Image

### 12.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
BillboardMask	False	Node3D	Sets the render node, that defines the position of the bitmap used as mask for blur.	
BitmapMask	True	Bitmap	Sets the bitmap, which is used as mask for blur.	
BlendOutput	True	Bool	Which Rendermode the Blur Output camera will be using	
BlurEnabled	True	Bool	Whether or not blurring is enabled.	
BlurMaskVariant	True	Enum	Sets the selected shader for blur	
BlurRadius	True	Int	Sets the radius of the blur.	
BlurRadiusToSigma	True	Float	Sets the factor to calculate sigma from the blur radius. Typical values are between 0.5 and 0.33.	
BlurShaderVariant	False	Enum	Sets the selected shader for blur	
CombineMask	False	Bool	Sets flag to determine if the mask render target is combined with the blurred scene.	
Darken	True	Float	Sets the darkening factor for the shader	
DirtyAreaRectangle	False	Rectangle	Sets the rectangle which is set as dirty area for blurring.	
Enabled	False	Bool	Enabled: Enable or disable the widget	
KeepFramebufferContent	False	Bool	Sets flag to determine if the content outside the blur mask is taken from the input framebuffer or discarded.	
Multipass	False	Bool	Sets single pass or multi pass blur shader.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node3D	The associated node of the widget.	

RenderTargetMask	False	RenderTarget	Sets the render target that is used as mask for blur.	
SubsamplingFactor	False	Int	Sets the subsampling factor for the blurred render target.	
UseManualDirtyArea	False	Bool	Sets flag that determines, if a manually defined dirty area is to be used.	

## 12.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	-------------------	--------------------	----------------	---------------------	-------------------

## Chapter 13

# ButtonAnimationWidget2D

**Name:** ButtonAnimationWidget2D

**Description:** Button extension widget which provides support to create simple animations started when the button state changes. The button state contains 4 flags (Enabled, Pressed, Active and Focused) which can be set to true or false independently. Therefore we have 8 possible transitions which can be used as triggers for the animation. At least one trigger should be enabled otherwise the animation is never started. Additionally for the flags which are not used as triggers it is possible to set up guards, conditions evaluated when the triggers are received. For example start an animation on button press but only if the button is also active.

**Category:** Button

### 13.1 Property list

Name	Bindable	Type	Description	Test Scope
Amplitude	False	Float	Amplitude : Courier::Float	
BounceCount	False	Byte	BounceCount : Courier::UInt8	
Channel1	True	custom://SoundChannel	SoundChannel	Ignored: This property will be tested for Animation widget
Channel2	True	custom://SoundChannel	SoundChannel	Ignored: This property will be tested for Animation widget
CheckActive	True	Bool	If this property is enabled then the button Active flag will be checked before starting the animation.	TC_W60_14
CheckEnabled	True	Bool	If this property is enabled then the button Enabled flag will be checked before starting the animation.	TC_W60_15

CheckFocused	True	Bool	If this property is enabled then the button Focused flag will be checked before starting the animation.	Ignored: The TTFis simulation for the gesture action is not ready yet
CheckPressed	True	Bool	If this property is enabled then the button Pressed flag will be checked before starting the animation.	TC_W60_16
Direction	True	Enum	Direction of the Animation	Ignored: This property will be tested for <i>Animation</i> widget
EaseDirection	False	Enum	Direction of Ease Interpolation	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing
Exponent	False	Float	Exponent : Courier::Float	
ExponentElastic	False	Float	ExponentElastic : Courier::Float	
FunctionType	False	Enum	FunctionType of Ease Interpolation	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property has not been fully developed
InterpolationStrategy	False	Enum	Type of Interpolation used for Animation	
IsActive	True	Bool	The animation is started only if the button Active flag matches this property.	TC_W60_14
IsEnabled	True	Bool	The animation is started only if the button Enabled flag matches this property.	TC_W60_15
IsFocused	True	Bool	The animation is started only if the button Focused flag matches this property.	Ignored: The TTFis simulation for the gesture action is not ready yet
IsPressed	True	Bool	The animation is started only if the button Pressed flag matches this property.	TC_W60_16
KeyFrameCount	False	UInt	Number of Key Frames	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OnActivate	True	Bool	If this property is enabled and the button becomes active then the animation will be started. Guards can be configured to provide additional conditions.	TC_W60_10

OnDeactivate	True	Bool	If this property is enabled and the button becomes inactive then the animation will be started. Guards can be configured to provide additional conditions.	TC_W60_12
OnDisable	True	Bool	If this property is enabled and the button becomes disabled then the animation will be started. Guards can be configured to provide additional conditions.	TC_W60_05
OnEnable	True	Bool	If this property is enabled and the button becomes also enabled then the animation will be started. Guards can be configured to provide additional conditions.	TC_W60_07
OnFocus	True	Bool	If this property is enabled and the button gains focus then the animation will be started. Guards can be configured to provide additional conditions.	Ignored: The TTFis simulation for the gesture action is not ready yet
OnLoseFocus	True	Bool	If this property is enabled and the button loses focus then the animation will be started. Guards can be configured to provide additional conditions.	Ignored: The TTFis simulation for the gesture action is not ready yet
OnPress	True	Bool	If this property is enabled and the button is pressed then the animation will be started. Guards can be configured to provide additional conditions.	TC_W60_01
OnRelease	True	Bool	If this property is enabled and the button is released then the animation will be started. Guards can be configured to provide additional conditions.	TC_W60_03
OscillationCount	False	Byte	OscillationCount : Courier::UInt8	
Power	False	Byte	Power : Courier::UInt8	
PropertyAnimation	False	Enum	Animation Type	
RepeatTimes	False	UInt	Number of times the Animation has to repeat	
RestitutionCoefficient	False	Float	RestitutionCoefficient : Courier::Float	
ReverseOnActivate	True	Bool	If this property is enabled then the animation will run backwards when started by OnActivate trigger	TC_W60_11
ReverseOnDeactivate	True	Bool	If this property is enabled then the animation will run backwards when started by OnDeactivate trigger	TC_W60_13
ReverseOnDisable	True	Bool	If this property is enabled then the animation will run backwards when started by OnDisable trigger	TC_W60_06
ReverseOnEnable	True	Bool	If this property is enabled then the animation will run backwards when started by OnEnable trigger	TC_W60_08



ReverseOnFocus	True	Bool	If this property is enabled then the animation will run backwards when started by OnFocus trigger	Ignored: The TTFis simulation for the gesture action is not ready yet
ReverseOnLoseFocus	True	Bool	If this property is enabled then the animation will run backwards when started by OnLoseFocus trigger	Ignored: The TTFis simulation for the gesture action is not ready yet
ReverseOnPress	True	Bool	If this property is enabled then the animation will run backwards when started by OnPress trigger	TC_W60_02
ReverseOnRelease	True	Bool	If this property is enabled then the animation will run backwards when started by OnRelease trigger	TC_W60_04
SequenceTime	True	custom://String	SequenceTime	Ignored: This property will be tested for <i>Animation</i> widget
ShouldRun	True	Bool	Auto Start the Animation when the view is loaded	Ignored: This property will be tested for <i>Animation</i> widget
SpeedFactor	False	Float	Speed of Animation	
ToggleDirection	False	Bool	Change the Animation Direction after completion of Animation	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W60_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
WidgetPropertyNameToAnimate	False	custom://String	Provide the name of the widget property which has to be animated	
WidgetToAnimate	False	Widget	Associate a widget whose property is to be animated	

13.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 14

# ButtonGroupWidget2D

**Name:** ButtonGroupWidget2D

**Description:** Groups buttons together and provides support for option groups in which maximum one button is active.

**Category:** Button

### 14.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
ActiveIndex	True	Int	Index of the active button. If set to -1 no button will be active.	TC_W61_01
AdornerContainerNode	False	Node2D	If the container node is specified, then the adorner will be added into this node, otherwise it will be added as direct child of the scene.	
AnimationDuration	True	UInt	Duration of the button slide animation. If set to 0 the duration configured in the adorner manager will be used.	TC_W61_02
AnimationEnabled	True	Bool	Enables the button slide animation. The animation will be started only if both this property and the parameter ButtonGroupReqMsg.Animate are true.	TC_W61_03
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development

ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: Not tested because this property is inherited and not specific to this Widget.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W61_08

FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.

PressHold	True	Bool	Enable hold gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.

TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	Ignored: Not tested because this property is inherited and not specific to this Widget.
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W61_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 14.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
ButtonGroupReqMsg	View		Action: enButton-GroupAction.  ButtonIndex: Courier::Int.  Animate: bool. +		

## Chapter 15

# ButtonImageWidget2D

**Name:** ButtonImageWidget2D

**Description:** Provides support to create buttons for which the background or the icon reflects the state of that button. This widget should be attached to a bitmap render node and the button widget should be attached to the same node or an ancestor of it.

**Category:** Button

### 15.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
ActiveDisabledBitmap	True	Image2D	Bitmap used when the button is Disabled, not Pressed, Active and not Focused.	TC_W05_01
ActiveNormalBitmap	True	Image2D	Bitmap used when the button is Enabled, not Pressed, Active and not Focused.	TC_W05_01
ActivePressedBitmap	True	Image2D	Bitmap used when the button is Enabled, Pressed, Active and not Focused.	
ActivePressedDisabledBitmap	True	Image2D	Bitmap used when the button is Disabled, Pressed, Active and not Focused.	
Color	True	Color	Color to be set on the effect of the RenderNode.	TC_W05_03
DisabledBitmap	True	Image2D	Bitmap used when the button is Disabled, not Pressed, not Active and not Focused.	TC_W05_01
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W05_08



FallbackStrategyEnabled	True	Bool	The button image checks the bitmap associated with the current state. If that bitmap is not set and the fallback strategy is enabled other bitmap will be used, otherwise a null bitmap will be set on the bitmap brush. This fallback strategy works as follows: first the Focused flag is ignored, second the Active flag and third the Pressed flag.	TC_W05_02
FocusedActiveDisabledBitmap	True	Image2D	Bitmap used when the button is Disabled, not Pressed, Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedActiveNormalBitmap	True	Image2D	Bitmap used when the button is Enabled, not Pressed, Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedActivePressedBitmap	True	Image2D	Bitmap used when the button is Enabled, Pressed, Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedActivePressedDisabledBitmap	True	Image2D	Bitmap used when the button is Disabled, Pressed, Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedDisabledBitmap	True	Image2D	Bitmap used when the button is Disabled, not Pressed, not Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedNormalBitmap	True	Image2D	Bitmap used when the button is Enabled, not Pressed, not Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.

FocusedPressedBitmap	True	Image2D	Bitmap used when the button is Enabled, Pressed, not Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedPressedDisabledBitmap	True	Image2D	Bitmap used when the button is Disabled, Pressed, not Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NormalBitmap	True	Image2D	Bitmap used when the button is Enabled, not Pressed, not Active and not Focused.	TC_W05_01
PressedBitmap	True	Image2D	Bitmap used when the button is Enabled, Pressed, not Active and not Focused.	
PressedDisabledBitmap	True	Image2D	Bitmap used when the button is Disabled, Pressed, not Active and not Focused.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W05_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 15.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
------	--------------	-------------	---------	--------------	------------

## Chapter 16

# ButtonTextColorWidget2D

**Name:** ButtonTextColorWidget2D

**Description:** Changes the color of a text depending on the button state. This extension widgets uses two more widgets in the same scene: one button widget from which it will get the state and one text widget on which it will set the text color depending on the button state. Those two widgets should be linked to nodes which are ancestors of this widget's node. A particular situation would be when the button widget, the text widget and the button text color widget are linked to the same node.

**Category:** Button

### 16.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
ActiveDisabledTextColor	True	Color	Text color used when the button is Disabled, not Pressed, Active and not Focused.	Ignored: Not tested because this property is having a reported problem but the fix is not available. Kindly see in RTC_950198.
ActiveNormalTextColor	True	Color	Text color used when the button is Enabled, not Pressed, Active and not Focused.	Ignored: Not tested because this property is having a reported problem but the fix is not available. Kindly see in RTC_950198.

ActivePressedDisabledTextColor	True	Color	Text color used when the button is Disabled, Pressed, Active and not Focused.	
ActivePressedTextColor	True	Color	Text color used when the button is Enabled, Pressed, Active and not Focused.	
DisabledTextColor	True	Color	Text color used when the button is Disabled, not Pressed, not Active and not Focused.	Ignored: Not tested because this property is having a reported problem but the fix is not available. Kindly see in RTC_950198.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W06_03
FocusedActiveDisabledTextColor	True	Color	Text color used when the button is Disabled, not Pressed, Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedActiveNormalTextColor	True	Color	Text color used when the button is Enabled, not Pressed, Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedActivePressedDisabledTextColor	True	Color	Text color used when the button is Disabled, Pressed, Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedActivePressedTextColor	True	Color	Text color used when the button is Enabled, Pressed, Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.

FocusedDisabledTextColor	True	Color	Text color used when the button is Disabled, not Pressed, not Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedNormalTextColor	True	Color	Text color used when the button is Enabled, not Pressed, not Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedPressedDisabledTextColor	True	Color	Text color used when the button is Disabled, Pressed, not Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
FocusedPressedTextColor	True	Color	Text color used when the button is Enabled, Pressed, not Active and Focused.	Ignored: Not tested because focus states are not supported in the test app.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NormalTextColor	True	Color	Text color used when the button is Enabled, not Pressed, not Active and not Focused.	Ignored: Not tested because this property is having a reported problem but the fix is not available. Kindly see in RTC_950198.

OutlineColorConfiguration	True	Bool	If set to true this widget will configure the outline color property of the text widget instead of the text color. Please refer to the outline effects documentation for information on how to render text with outline.	TC_W06_02
PressedDisabledTextColor	True	Color	Text color used when the button is Disabled, Pressed, not Active and not Focused.	
PressedTextColor	True	Color	Text color used when the button is Enabled, Pressed, not Active and not Focused.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W06_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 16.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 17

# ButtonTextScrollWidget2D

**Name:** ButtonTextScrollWidget2D

**Description:** Scrolls the text associated with a button widget. This extension widgets uses two more widgets in the same scene: one button widget from which it will get the state and one text widget on which it will set the text scrolling depending on the button state. Those two widgets should be linked to nodes which are ancestors of this widget's node. A particular situation would be when the button widget, the text widget and the button text scroll widget are linked to the same node.

**Category:** Button

### 17.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
ScrollOnFocusGain	True	Bool	Starts scrolling the text when the button gains the focus.	
StopOnSpeedLock	True	Bool	Stops scrolling the text when the speedlock is enabled.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

17.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------



## Chapter 18

# ButtonWidget2D

**Name:** ButtonWidget2D

**Description:** Interactive widget which sends reaction messages when pressed or released. The button state contains several flags (enabled, pressed, active and focused) which are set in various ways. Additional widgets like ButtonImageWidget2D or ButtonTextColorWidget2D are required to model the appearance of a button which provides feedback about its state. See the widget documentation for more details.

**Category:** Button

### 18.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	TC_W04_03

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: This property is tested through Button in List of List-Combination Widget 2D
Drag	True	Bool	Enable drag gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
EditorFocused	False	Bool	Sets the focused flag in order to preview the appearance in SceneComposer. It has no effect in the simulation or on the target.	
EditorPressed	False	Bool	Sets the pressed flag in order to preview the appearance in SceneComposer. It has no effect in the simulation or on the target.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W04_08
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.

FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
IsActive	True	Bool	Indicates if the button is active. For toggle/radio button active means checked/selected. For normal button active means that the option/action is active (for example track is playing, station is tuned, etc).	TC_W04_02
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PostButtonReactionMsg	False	Bool	ButtonReactionMsg messages will be posted only if this property is true.	TC_W04_01

PressHold	True	Bool	Enable hold gesture detection	Ignored: This property is tested through Button in List of List-Combination Widget 2D
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: This property is tested through Button in List of List-Combination Widget 2D
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: This property is tested through Button in List of List-Combination Widget 2D
TimerConfiguration	False	Enum	Configures the timer.	

TouchHandler	True	Int	Specifies the id of the touch handler for this button. A touch handler can provide a custom sensitive area for buttons (for example circle, triangle, etc). Touch handlers are registered at application start-up.	Ignored: Not tested because this property is inherited and not specific to this Widget.
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W04_01
TouchableArea	False	Rectangle	Configures the touchable area of the button. X and Y are coordinates relative to the upper left corner of the node effective bounding rectangle. If width or height are zero or negative the effective bounding rectangle of the node will be used as touchable area (X and Y will also be ignored).	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W04_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 18.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
------	--------------	-------------	---------	--------------	------------

ButtonListItemUpdMsg	Controller, Model	ListId: Courier::UInt32.  Hdl: Courier::UInt32.  SubHdl: Courier::UInt32.  Reaction: enReaction. +		TC_W25_04
ButtonReactionMsg	Controller, View, Model	enReaction: enReaction.  SurfaceId: Courier::UInt32. +	sequential	TC_W25_04
ButtonSlideMsg	Controller, Model	Event: hmibase::widget: +	sequential gesture::enGestureEvent::Enum	TC_W04_11

## Chapter 19

# CameraControlWidget2D

**Name:** CameraControlWidget2D

**Description:** Controls the properties of a camera allowing to expose them outside composites or to change them using data binding sources.

**Category:** Common

### 19.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

UseForTouch	True	Bool	Enables the associated camera to be used in the beginning of the touch session for checking if the touch coordinates are inside the bounding rectangle of a widget's node. Cameras which render into window surfaces usually need to have it enabled. Cameras which render into framebuffers need it sometimes disabled to prevent the buttons from being touchable outside their visual position.	
UseForTouchEnabled	False	Bool	Enables the configuration of the camera internal property UseForTouch.	
Viewport	True	Rectangle	Sets the Viewport property of the associated camera.	TC_W62_04
ViewportEnabled	False	Bool	Enables the configuration of the camera property Viewport.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 19.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>



## Chapter 20

# CanvasLayouterWidget2D

**Name:** CanvasLayouterWidget2D

**Description:** DEPRECATED! Don't use this widget! Use an overlay layouter instead. Sets a canvas layouter (position will be kept as defined and layouting will be forwarded to the child nodes) on a node.

**Category:** Deprecated

### 20.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
LayoutChildren	True	Bool	The value that will be set for the LayoutChildren property of the ancestor's FlexCanvasLayouter instance.	
Name	False	CharArray	The name of the widget instance	
NestedLevel	True	UInt	The ancestor's level, relative to the associated node, that will have the layouter set to an FlexCanvasLayouter.	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

20.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 21

# ClockWidget2D

**Name:** ClockWidget2D

**Description:** ClockWidget is used to display an analogue clock with a clock face as background image and two or three clock hand images for hours, minute and second. The clock is skinnable, meaning the complete set of images(clock face and hands) can be changed during runtime.

**Category:** Extra

### 21.1 Property list

Name	Bindable	Type	Description	Test Scope
ClockFaceImageName	False	custom://String	String for the clock face image (e.g. 'ClockFaceBmp'.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing.
Hour	True	Byte	The hour value (0..23).	TC_W07_02
HourImageName	False	custom://String	String for the hour image stored in the asset. The can be used with an image set of 60 images, in this case the name should contain a placeholder [%d] for the hours index. If the name has no placeholder, than only one single image will be uploaded and internal SetRotate() will update the hour (e.g. 'ClockHour_%02dBmp').	
ImagePath	False	custom://String	Path to the clock images which must include a placeholder [%d] for the skin index (e.g. 'Widgets#Scenes2D#Bitmaps#Clock#Skin%d')	

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Minute	True	Byte	The Minute value (0..59).	TC_W07_02
MinuteImageName	False	custom://string	String for the hour image stored in the asset. The can be used with an image set of 60 images, in this case the name should contain a placeholder [%d] for the minutes index. If the name has no placeholder, than only one single image will be uploaded and internal SetRotate() will update the minute (e.g. 'ClockMin_%02dBmp').	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Second	True	Byte	The Second value (0..59).	TC_W07_02
SecondImageName	False	custom://string	String for the hour image stored in the asset. The can be used with an image set of 60 images, in this case the name should contain a placeholder [%d] for the seconds index. If the name has no placeholder, than only one single image will be uploaded and internal SetRotate() will update the seconds (e.g. 'ClockSec_%02dBmp').	
SkinIndex	True	Byte	The index of the skin (0,1,...).	TC_W07_01
SmoothRotation	False	Bool	Hour and minute hands movement will be smooth , if it sets to true	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W07_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 21.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 22

# CollapseWidget2D

**Name:** CollapseWidget2D

**Description:** DEPRECATED! Don't use this widget! Instead set the Collapsible property of the node and hide it using the Visible property of any widget attached to it. Collapses a node by settings its size to (0,0) and its visibility to false.

**Category:** Deprecated

### 22.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Collapsed	True	Bool	Whether the ancestor will be collapsed or not.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
NestedLevel	True	UInt	The ancestor's level, relative to the associated node, that will be collapsed.	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

22.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 23

# ColorBarWidget2D

**Name:** ColorBarWidget2D

**Description:** Provides support for a horizontal or a vertical color bar. The color bar thickness is 1 pixel (use node scale or stretch to achieve a different thickness) and its length is the sum of all segments length. Each segment is defined by a length and a color.

**Category:** Common

### 23.1 Property list

Name	Bindable	Type	Description	Test Scope
Colors	True	Color	Colors used for segments	TC_W46_01
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property is inherited and not specific to this Widget.
Id	True	UInt	Unique identifier of a widget. ColorBarDataUpdMsg identifies this Id to send update to specific widget	TC_W46_03
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property is inherited and not specific to this Widget.
Lengths	True	UInt	Array of length defines length of each segment in the color bar.	TC_W46_02
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Orientation	False	Enum	Setup the orientation of color bar, should be vertical or horizontal	



Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W46_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 23.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
ColorBarDataUpdatedMsg	Widget	Response from DataModel.  Properties: Id: Define the id of bar item Data: The data array, define the color and length of each bar	Id: ::Candera::UInt32.  Data: tColorBarDataPtr. +		TC_W46_03

## Chapter 24

# ColorEffectWidget2D

**Name:** ColorEffectWidget2D

**Description:** Manipulates the Color property of the existing effect of a RenderNode.

**Category:** Common

### 24.1 Property list

Name	Bindable	Type	Description	Test Scope
Color	True	Color	Color to be set on the effect of the RenderNode.	TC_W45_01
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This derived property is not used in ColorEffect widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W45_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

24.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 25

# ColorSwitchWidget2D

**Name:** ColorSwitchWidget2D

**Description:** Selects a color from an array based on an index or a source widget state and sets it on a target color property of an effect or another widget.

**Category:** Common

### 25.1 Property list

Name	Bindable	Type	Description	Test Scope
Colors	True	Color	Colors : Candra::ArrayProperty<Candra::Color>	TC_W42_02
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing.
Index	True	UInt	Index used to select the color.	TC_W42_01
IndexSource	False	Enum	Specifies how the index is determined. It can be explicitly specified using the Index property or it can be obtained based on the state of another widget associated to the same node or an ancestor node.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget

Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
StateFlags	False	Enum	<p>Specifies which flags of the source widget state are used to determine the index. The number of colors required for the Colors array property depends on the number of flags which are used:</p> <ul style="list-style-type: none"> <li>- 1 flag (Enabled, Pressed, Active, Focused) =&gt; 2 colors [0, 1],</li> <li>- 2 flags (PressedEnabled, ActiveEnabled, ActivePressed, FocusedEnabled, FocusedPressed, FocusedActive) =&gt; 4 colors [00, 01, 10, 11],</li> <li>- 3 flags (ActivePressedEnabled, FocusedPressedEnabled, FocusedActivePressed) =&gt; 8 colors [000, 001, 010, 011, 100, 101, 110, 111],</li> <li>- 4 flags (FocusedActivePressedEnabled) =&gt; 16 colors [0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111].</li> </ul> <p>The order of the flags should match the order of the colors. For example, PressedEnabled requires 4 colors:</p> <ul style="list-style-type: none"> <li>- NotPressed_NotEnabled=Colors[0],</li> <li>- NotPressed_Enabled=Colors[1],</li> <li>- Pressed_NotEnabled=Colors[2],</li> <li>- Pressed_Enabled=Colors[3].</li> </ul>	
Target	False	Enum	Specifies how the target color property is determined. It can be a property of an effect contained in the node or it can be a property of a widget associated to the same node as this widget.	
Visible	True	Bool	<p>Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.</p>	TC_W42_09
VisibleEnabled	False	Bool	<p>Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.</p>	

25.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 26

# ComboBoxWidget2D

**Name:** `ComboBoxWidget2D`

**Description:** Provides support to implement comboboxes which are combination between a button (or other touchable item) and a drop down panel

**Category:** Under construction

### 26.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	
ButtonNode	False	Node2D	The node acting as a selection button and when it is pressed, content of the combo box will be open/closed.	
ComboBoxId	True	UInt	A unique identifier of the combo box.	
ContentNode	False	Node2D	The node containing the actual content of the combobox. This node should have additional group node (For layouting purpose), where contents like button, text, bg shall be added. Don't use any layouters on the content node as widget will override with its custom layouter. Also make sure ArabicLayouterPatchWidget should be added in the scene for layouting the content node correctly	

ContentSize	True	Vector2	The size to be set for the content node [Total size of dropdown panel]. For example, adding 3 button inside the drop down panel where each button size is (W*H) 200*50 then total size to be set on the property is 200*150	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	
DoubleTap	True	Bool	Enable double tap gesture detection	
Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	
FocusOrder	True	Short	Focus order. Zero has the highest priority.	
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	



InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	
Rotate	True	Bool	Enable rotate gesture detection	
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	
Touchable	True	Bool	Widget is Touchable or not	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	
ViewportNode	False	Node2D	The node is mapped to list node. In order to properly position the content of the combo box above or below the combobox button its required to know the available space in the viewport hence the list node should be mapped to viewport node.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 26.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
ComboBoxReqMsgView	View		ComboBoxId: ::Courier::UInt32.  Action: enComboBox- Action::Enum. +	sequential	
ComboBoxUpdMsgModel	Model, Controller		ComboBoxId: ::Courier::UInt32.  IsContentOpen: bool. +	sequential	

## Chapter 27

# ControlTemplateWidget2D

**Name:** `ControlTemplateWidget2D`

**Description:** DEPRECATED! Don't use this widget! Use ListBindingWidget2D instead.

**Category:** Deprecated

### 27.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
EnabledBindingIndex	False	Int	Binds the Enabled property of a widget to an integer value.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
ItemsBindingIndex	False	Int	Binds the property List.ListId to an integer value. It can also be used to fill a sublist with a vector data. See the list widget user guide.	
Name	False	CharArray	The name of the widget instance	
NameBindingIndex	False	Int	Binds the Name property of a widget to a text value.	
Node	False	Node2D	The associated node of the widget.	

SelectedBindingIndex	False	Int	Binds a widget property to an integer value. The following widget properties can be bound: Button.IsActive, ButtonGroup.ActiveIndex, Collapse.Collapsed, ColorSwitch.Index, ImageSwitch.Index, List.StartIndex, Switch.Index, TextColor.Active and Toggle.Enabled.	
TextBindingIndex	False	Int	Binds the Text property of a widget to a text value. The following widgets support it: Text, Label, ScrollableText and TextArea.	
UserDataBindingIndex	False	Int	Binds the UserData to an integer value.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 27.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 28

# CoverflowListWidget2D

**Name:** CoverflowListWidget2D

**Description:** Particular implementation of the ListWidget2D, specialized for coverflow.

**Category:** List

### 28.1 Property list

Name	Bindable	Type	Description	Test Scope
AcceptImmediatePositioning	True	Bool	If set to true, immediate positioning is taken into consideration. Please check the other properties to enable immediate positioning for different events.	
AcceptImmediatePositioningOnFirstAppearance	True	Bool	If set to true, the first ListChangeMsg with ListChangeSet that is received after rendering is enabled for the view and if immediate positioning is activated, then the position is set to the received value without using an animation; further position setting uses an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AcceptImmediatePositioningOnItemScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangeDown or ListChangeUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	

AcceptImmediatePositioningOnPageScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangePageDown or ListChangePageUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AcceptImmediatePositioningOnPositionSet	True	Bool	If set to true, if a ListChangeMsg with ListChangeSet is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AnimationOffset	False	Short	Offset from first visible item to the position of the first animated item, give a negative value for the offset to the first invisible item of the list	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
AutoViewportSize	False	Bool	If enabled, the size of the viewport will be calculated as NumberOfItems multiplied with the size of the InvalidItemTemplate. This is done for the direction of scrolling; the other direction will remain unmodified.	
BufferSize	True	UInt	Used for windowed data. The buffer size of the elements requested prior and after the visible ones.	
CachedLayout	False	Bool	Caches the layout of the list entries, if set to true the layout calculation is done only once for each list item and then cached.	
ConfigureFocusGroup	True	Bool	Focus group is configured explicitly or uses default configuration.	Ignored: The TTFis simulation for the gesture action is not ready yet.
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
Coverflow	True	Bool	True if the list is a coverflow.	

CustomAnimationsGroupNode	False	Node2D	Root node of animated nodes that are placeholders for the real items in the list. The animated properties will be applied to the nodes that will be added by the list to the ItemsNode.	
CustomListAnimations	False	Animation	Array of animations for all visible list elements.	
DefaultFocusOrder	True	Short	Focus order of the element which should become focused when this group becomes active.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	
DragAndSwipeOnNeed	True	Bool	If true then drag and swipe will be dynamically disabled when a new touch session starts if all items are visible.	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DynamicGrid	False	Bool	Whether or not the grid is automatically adapting to the text size.	
DynamicGridEmptyCellsPolicy	False	Enum	The policy to distribute the empty spaces in the last row of the list	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	

ExpandAutoScroll	True	Bool	Whether or not expand animations scroll the list so that the expanded item fully fits inside the viewport.	
FixedPageScrolling	True	Bool	Works only if all the data is available (no windowed data). Should be used only with small lists. If set to true, the pages always start at a fixed position, similar to a book. If set to false, the pages are always considered relative to the current start index.	
FixedPageScrollingOffset	True	Float	Percentage of the viewport needed to be scrolled/swiped with the touch in order to go to the next/previous page.	
FocusAnchorId	True	UInt		Ignored: Not available in CGI configuration, not mentioned in user guide
FocusAnchorMode	False	Enum		
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTF is simulation for the gesture action is not ready yet.
FocusLayer	True	Short	Specifies to which focus layer this group belongs. There should be distinct layers for main surface screens, subspeller, popups to prevent focusing of elements in the main screen when a popup is displayed. Only elements in the groups with the highest layer will be focused.	Ignored: The TTF is simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTF is simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
FocusWrapAround	True	Bool	Focus wraps around to first/last element.	Ignored: The TTF is simulation for the gesture action is not ready yet.



Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedIndex	True	Int	The index of the currently focused item.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedNode	False	Node2D	The node that will highlight the focused node.	
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
IgnoreListChangeMsg	False	Enum	Defines when the ListChangeMsg will be ignored during the list scrolling or swiping	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
InvalidItemTemplate	False	Node2D	Template used for not yet available data or if no other template is found for some data.	
ItemsNode	False	Node2D	The list items will be added as children of this node.	
LimitExceededDownAnimation	False	Animation	Animation that is played when the user wants to exceed the list on bottom. It is only played if LimitExceededDownAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitExceededDownAnimationTime	True	UInt	Defines the how much time (ms) the exceeded down animation takes.	
LimitExceededDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on bottom of the list.	

LimitExceededDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitExceededUpAnimation	False	Animation	Animation that is played when the user wants to exceed the list on top. It is only played if LimitExceededUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitExceededUpAnimationTime	True	UInt	Defines the how much time (ms) the exceeded up animation takes.	
LimitExceededUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on top of the list.	
LimitExceededUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitReachedDownAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedDownAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedDownAnimationTime	True	UInt	Defines the how much time (ms) the reached down animation takes.	
LimitReachedDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on bottom of the list.	
LimitReachedDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitReachedUpAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedUpAnimationTime	True	UInt	Defines the how much time (ms) the reached up animation takes.	
LimitReachedUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on top of the list.	
LimitReachedUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.

ListAlignment	False	Enum	Only applicable for lists with less then the maximum number of visible items. If set to 'End' the items will be aligned at the end of the list.	
ListId	True	UInt	A unique identifier of the list that will be used to request data to the model and react to data sent from the model.	
LockOutFixMovementOffset	False	Float	Offset in percent of the viewport required for LockOutType LockOutFixMovement.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NumberOfItems	True	UInt	The number of items is only available for itemwise lists(PixelWiseScrollingEnabled set to false). For stack layouted lists it has the meaning of the number of elements that would be visible. For grid layouted lists it has the meaning of the number of columns(for horizontal scrolling) or rows(for vertical scrolling) that will be seen. The number of rows(for horizontal scrolling) or columns(for vertical scrolling) will be taken from the GridLayouter.	
PageIndicationAccuracy	False	Float	Accuracy of the current page position indication.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PixelWiseAnimationTime	True	UInt	Time in ms to finish the pixel wise animation	
PixelWiseScrollingEnabled	True	Bool	If set to true, scrolling is done pixel by pixel, meaning that an item can be partially visible. If set to false, scrolling is done item by item; items ar always completely visible.	
PostListChanged	False	Bool	If true, the list will post a ListChangedUpdMsg message everytime the first visible index changes.	
PreparedItemsBufferSize	False	Byte	Number of additional items being prepared when the list is idle	
PreparedItemsUpdateTriggerOffset	False	Byte	Offset to the edge of the PreparedItemsBuffer - the prepared item buffer will be updated and old items discarded as soon as this offset is reached	

PreserveFocus	True	Bool	Current focused element is preserved when group becomes inactive.	Ignored: The TTF is simulation for the gesture action is not ready yet.
PreserveScrollIndex	False	Bool	If enabled, and the list is cached, the widget will remember the last scrolled position and set the same on re-entering the List View, unless set to a different value by the model.	
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollAnimationInput	False	Enum	Input of scroll animation - Either pixel position or index of list items	
ScrollbarAlwaysVisible	True	Bool	If set to true, scrollbar is always visible, regardless if it's needed or not. If set to false, it's only visible if needed.	
ScrollingOrientation	True	Enum	Scrolling orientation: vertical or horizontal	
ScrollingType	False	Enum	Sets the way this list scrolls: Default beginning to end, continuous or last page not filled.	
ShortPixelWiseAnimationTime	True	UInt	Short time in ms to finish the pixel wise animation; used when setting the position with a ListChangeSet request type	
ShowItemsOnViewActivated	False	Bool	Whether items are visible or not on view shown, before receiving a ListCustomAnimationReqMsg.	
Snap	False	Enum	Specifies how the list should behave after scrolling. SnapToEdge - List will snap to the nearest edge, SnapToCenter - List will snap to the center	TC_W27_04
SnapHoldTime	False	UInt	Timeout in milliseconds, if reached no snapping will be done	
SnapOffsetThreshold	False	UInt	Threshold of offsetDelta = Abs(offsetTop - offsetBottom) in pixels - if the offset delta is smaller than the threshold the snap will be done to the edge closer to the touch position	

StartIndex	True	Int	The index of the first visible item. Negative indices are accepted in case of circular scrolling.	
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
SwipingAcceleration	True	Float	Acceleration used to slow down swiping	
SwipingDistanceThreshold	True	UInt	Deprecated. Please do not use this, as it will be removed in a future version. Minimum distance in display units needed to drag in order to begin scrolling	Ignored: Deprecated
SwipingMaxDistance	True	Float	Maximum distance that can be covered within one swipe	
SwipingTimerThreshold	True	UInt	Minimum time in ms needed to press in order to focus an item	
SwipingVelocityThreshold	True	Float	Deprecated. Please do not use this, as it will be removed in a future version. Minimum velocity in display units/ms needed to move with in order to begin swiping	Ignored: Deprecated
Tap	True	Bool	Enable press and tap gesture detection	
TemplateGroup	False	Node2D	The parent of template nodes. Template nodes will be cloned based on the item needed to be displayed and the clone will be added to the ItemsNode.	
TemplateScrollAnimations	False	Animation	Array of template animations for list scrolling.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
UpdateTriggerOffset	True	UInt	Used for windowed data. The offset relative to the first or last visible item index, respectively used to request new data when reached.	
UsableViewportPadding	False	Margin	Viewport offset for visible items in the list - useful in case the list is overlapped by some other content.	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

28.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 29

# DelegateWidget2D

**Name:** DelegateWidget2D

**Description:** Delegates updates and messages to a widget controller. The widget controllers are registered at start-up and attached to widgets using the property ControllerId.

**Category:** Common

### 29.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.

Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.



GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.

Touchable	True	Bool	Widget is Touchable or not	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 29.2 Message list

<i>Name</i>	<i>Subscribes</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	-------------------	--------------------	----------------	---------------------	-------------------

## Chapter 30

# DirectTextureConsumer2D

**Name:** DirectTextureConsumer2D

**Description:** DirectTextureConsumer2D

**Category:** DirectTexture

### 30.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AlignSlaveSurfacePosition	True	Bool	set to true if position of slave surface should be aligned to widget position	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
Camera	False	Camera2D	The camera on which the slave surface should be synced	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.

GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
InstanceId	True	UInt	same unique identifier as the one used for the related provider instance	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NodeSizeBasedReposition	True	Bool	set to true if slave surface to be repositioned if Node size is equal to surface size	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
RawTouchSupport	True	Bool	enable when all kind of touch events are required. This will overrule the settings done in category Gestures!!! Handle with care as this will produce some bus load during move	
ReplacementNode	False	Node2D	the node which will get rendered when direct texture content is not available	

Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SlaveSurfaceId	True	UInt	reposition the surface with this id on every position change of the connected node	
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 30.2 Message list

Name	Subscribers	Description	Members	Distribution	Test Scope
------	-------------	-------------	---------	--------------	------------

## Chapter 31

# DirectTextureConsumer3D

**Name:** DirectTextureConsumer3D

**Description:** Direct Texture Consumer

**Category:** Examples

### 31.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Camera	False	Node3D	Camera used for intersection test.	
Enabled	False	Bool	Enabled: Enable or disable the widget	
Name	False	CharArray	The name of the widget instance	
Node	False	Node3D	The associated node of the widget.	

### 31.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 32

# DirectTextureProvider2D

**Name:** DirectTextureProvider2D

**Description:** DirectTextureProvider2D

**Category:** DirectTexture

### 32.1 Property list

Name	Bindable	Type	Description	Test Scope
Camera	False	Camera2D	The camera which contains the screen content	
Camera2	False	Camera2D	Another camera used for triple buffering only	
Camera3	False	Camera2D	Another camera used for triple buffering only	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FrameBuffer	False	RenderTarget	Off screen render target where to render the content to	
FrameBuffer2	False	RenderTarget	Another off screen render target used for triple buffering only	
FrameBuffer3	False	RenderTarget	Another off screen render target used for triple buffering only	



InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
InstanceId	True	UInt	System wide unique id to identify the offscreen content	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
ReleaseBufferOnHide	True	Bool	set to true if buffer to be released when view is hidden	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 32.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 33

# DirectTextureProvider3D

**Name:** DirectTextureProvider3D

**Description:** Direct Texture Widget

**Category:** Examples

### 33.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Camera	False	Node3D	Camera used for intersection test.	
Enabled	False	Bool	Enabled: Enable or disable the widget	
FrameBuffer	False	RenderTarget	Frame buffer object	
Name	False	CharArray	The name of the widget instance	
Node	False	Node3D	The associated node of the widget.	

### 33.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 34

# DockSideWidget2D

**Name:** DockSideWidget2D

**Description:** DEPRECATED! Don't use this widget!

**Category:** Deprecated

### 34.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
DockSide	True	Enum	The value to set for the DockSide property of the ancestor node.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
NestedLevel	True	UInt	The ancestor's level, relative to the associated node, that will have the DockSide property modified.	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

34.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 35

# DropDownListWidget2D

**Name:** DropDownListWidget2D

**Description:** Provides support to implement comboboxes (which are composed by a button and a dropdown list).

**Category:** List

### 35.1 Property list

Name	Bindable	Type	Description	Test Scope
AcceptImmediatePositioning	True	Bool	If set to true, immediate positioning is taken into consideration. Please check the other properties to enable immediate positioning for different events.	
AcceptImmediatePositioningOnFirstAppearance	True	Bool	If set to true, the first ListChangeMsg with ListChangeSet that is received after rendering is enabled for the view and if immediate positioning is activated, then the position is set to the received value without using an animation; further position setting uses an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AcceptImmediatePositioningOnItemScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangeDown or ListChangeUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	

AcceptImmediatePositioningOnPageScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangePageDown or ListChangePageUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AcceptImmediatePositioningOnPositionSet	True	Bool	If set to true, if a ListChangeMsg with ListChangeSet is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AnimationOffset	False	Short	Offset from first visible item to the position of the first animated item, give a negative value for the offset to the first invisible item of the list	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
AutoViewportSize	False	Bool	If enabled, the size of the viewport will be calculated as NumberOfItems multiplied with the size of the InvalidItemTemplate. This is done for the direction of scrolling; the other direction will remain unmodified.	
BufferSize	True	UInt	Used for windowed data. The buffer size of the elements requested prior and after the visible ones.	
ButtonNode	False	Node2D	The node acting as a button and when pressed will open the drop down list.	
CachedLayout	False	Bool	Caches the layout of the list entries, if set to true the layout calculation is done only once for each list item and then cached.	
ConfigureFocusGroup	True	Bool	Focus group is configured explicitly or uses default configuration.	Ignored: The TTFis simulation for the gesture action is not ready yet.
ContentNode	False	Node2D	The node containing the selected item.	

ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time
Coverflow	True	Bool	True if the list is a coverflow.	
CustomAnimationsGroupNode	False	Node2D	Root node of animated nodes that are placeholders for the real items in the list. The animated properties will be applied to the nodes that will be added by the list to the ItemsNode.	
CustomListAnimations	False	Animation	Array of animations for all visible list elements.	
DefaultFocusOrder	True	Short	Focus order of the element which should become focused when this group becomes active.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	
DragAndSwipeOnNeed	True	Bool	If true then drag and swipe will be dynamically disabled when a new touch session starts if all items are visible.	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DynamicGrid	False	Bool	Whether or not the grid is automatically adapting to the text size.	
DynamicGridEmptyCellsPolicy	False	Enum	The policy to distribute the empty spaces in the last row of the list	

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
ExpandAutoScroll	True	Bool	Whether or not expand animations scroll the list so that the expanded item fully fits inside the viewport.	
FixedPageScrolling	True	Bool	Works only if all the data is available (no windowed data). Should be used only with small lists. If set to true, the pages always start at a fix position, similar to a book. If set to false, the pages are always considered relative to the current start index.	
FixedPageScrollingOffset	True	Float	Percentage of the viewport needed to be scrolled/swiped with the touch in order to go to the next/previous page.	
FocusAnchorId	True	UInt		Ignored: Not available in CGI configuration, not mentioned in user guide
FocusAnchorMode	False	Enum		
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusLayer	True	Short	Specifies to which focus layer this group belongs. There should be distinct layers for main surface screens, subspeller, popups to prevent focusing of elements in the main screen when a popup is displayed. Only elements in the groups with the highest layer will be focused.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	



FocusWrapAround	True	Bool	Focus wraps around to first/last element.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedIndex	True	Int	The index of the currently focused item.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedNode	False	Node2D	The node that will highlight the focused node.	
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
IgnoreListChangeMsg	False	Enum	Defines when the ListChangeMsg will be ignored during the list scrolling or swiping	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
InvalidItemTemplate	False	Node2D	Template used for not yet available data or if no other template is found for some data.	
ItemsNode	False	Node2D	The list items will be added as children of this node.	
LimitExceededDownAnimation	False	Animation	Animation that is played when the user wants to exceed the list on bottom. It is only played if LimitExceededDownAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitExceededDownAnimationTime	True	UInt	Defines the how much time (ms) the exceeded down animation takes.	

LimitExceededDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on bottom of the list.	
LimitExceededDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitExceededUpAnimation	False	Animation	Animation that is played when the user wants to exceed the list on top. It is only played if LimitExceededUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitExceededUpAnimationTime	True	UInt	Defines the how much time (ms) the exceeded up animation takes.	
LimitExceededUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on top of the list.	
LimitExceededUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitReachedDownAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedDownAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedDownAnimationTime	True	UInt	Defines the how much time (ms) the reached down animation takes.	
LimitReachedDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on bottom of the list.	
LimitReachedDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitReachedUpAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedUpAnimationTime	True	UInt	Defines the how much time (ms) the reached up animation takes.	
LimitReachedUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on top of the list.	

LimitReachedUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
ListAlignment	False	Enum	Only applicable for lists with less than the maximum number of visible items. If set to 'End' the items will be aligned at the end of the list.	
ListId	True	UInt	A unique identifier of the list that will be used to request data to the model and react to data sent from the model.	
ListNode	False	Node2D	The node containing the dropdown list.	
ListPosition	False	Vector2	Value to be set for ListNode if UserListPosition is true.	
ListSize	False	Vector2	The size to be set for the ListNode.	
LockOutFixMovementOffset	False	Float	Offset in percent of the viewport required for LockOutType LockOutFixMovement.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NumberOfItems	True	UInt	The number of items is only available for itemwise lists(PixelWiseScrollingEnabled set to false). For stack layouted lists it has the meaning of the number of elements that would be visible. For grid layouted lists it has the meaning of the number of columns(for horizontal scrolling) or rows(for vertical scrolling) that will be seen. The number of rows(for horizontal scrolling) or columns(for vertical scrolling) will be taken from the GridLayouter.	
PageIndicationAccuracy	False	Float	Accuracy of the current page position indication.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PixelWiseAnimationTime	True	UInt	Time in ms to finish the pixel wise animation	
PixelWiseScrollingEnabled	True	Bool	If set to true, scrolling is done pixel by pixel, meaning that an item can be partially visible. If set to false, scrolling is done item by item; items are always completely visible.	
PostListChanged	False	Bool	If true, the list will post a ListChangedUpdMsg message everytime the first visible index changes.	
PreparedItemsBufferSize	False	Byte	Number of additional items being prepared when the list is idle	

PreparedItemsUpdateTriggerOffset	False	Byte	Offset to the edge of the PreparedItemsBuffer - the prepared item buffer will be updated and old items discarded as soon as this offset is reached	
PreserveFocus	True	Bool	Current focused element is preserved when group becomes inactive.	Ignored: The TTF is simulation for the gesture action is not ready yet.
PreserveScrollIndex	False	Bool	If enabled, and the list is cached, the widget will remember the last scrolled position and set the same on re-entering the List View, unless set to a different value by the model.	
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollAnimationInput	False	Enum	Input of scroll animation - Either pixel position or index of list items	
ScrollbarAlwaysVisible	True	Bool	If set to true, scrollbar is always visible, regardless if it's needed or not. If set to false, it's only visible if needed.	
ScrollingOrientation	True	Enum	Scrolling orientation: vertical or horizontal	
ScrollingType	False	Enum	Sets the way this list scrolls: Default beginning to end, continuous or last page not filled.	
ShortPixelWiseAnimationTime	True	UInt	Short time in ms to finish the pixel wise animation; used when setting the position with a ListChangeSet request type	
ShowItemsOnViewActivated	False	Bool	Whether items are visible or not on view shown, before receiving a ListCustomAnimationReqMsg.	
Snap	False	Enum	Specifies how the list should behave after scrolling. SnapToEdge - List will snap to the nearest edge, SnapToCenter - List will snap to the center	
SnapHoldTime	False	UInt	Timeout in milliseconds, if reached no snapping will be done	

SnapOffsetThreshold	False	UInt	Threshold of offsetDelta = Abs(offsetTop - offsetBottom) in pixels - if the offset delta is smaller than the threshold the snap will be done to the edge closer to the touch position	
StartIndex	True	Int	The index of the first visible item. Negative indices are accepted in case of circular scrolling.	
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
SwipingAcceleration	True	Float	Acceleration used to slow down swiping	
SwipingDistanceThreshold	True	UInt	Deprecated. Please do not use this, as it will be removed in a future version. Minimum distance in display units needed to drag in order to begin scrolling	Ignored: Deprecated
SwipingMaxDistance	True	Float	Maximum distance that can be covered within one swipe	
SwipingTimerThreshold	True	UInt	Minimum time in ms needed to press in order to focus an item	
SwipingVelocityThreshold	True	Float	Deprecated. Please do not use this, as it will be removed in a future version. Minimum velocity in display units/ms needed to move with in order to begin swiping	Ignored: Deprecated
Tap	True	Bool	Enable press and tap gesture detection	
TemplateGroup	False	Node2D	The parent of template nodes. Template nodes will be cloned based on the item needed to be displayed and the clone will be added to the ItemsNode.	
TemplateScrollAnimations	False	Animation	Array of template animations for list scrolling.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
UpdateTriggerOffset	True	UInt	Used for windowed data. The offset relative to the first or last visible item index, respectively used to request new data when reached.	
UsableViewportPadding	False	Margin	Viewport offset for visible items in the list - useful in case the list is overlapped by some other content.	
UseListPosition	False	Bool	If set to true, sets the position given by ListPosition property for the ListNode. If set to false, it uses the position set by the layout.	

UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 35.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
DropdownChangeMsg	Widget		DropdownChangeType: ::Candera::DropdownChangeType.  ListId: ::Courier::UInt32. +		
DropdownCurrentValueChangedMsg	Model Controller		CurrentValue: ::Courier::UInt32. +		
DropdownStateChangeMsg	Widget, Model		ListId: ::Courier::UInt32.  Open: bool. +		

## Chapter 36

# EditFieldWidget2D

**Name:** EditFieldWidget2D

**Description:** Widget used to display text input given by user dynamically

**Category:** Input

### 36.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
AutoCompletion	False	Bool	Property to enable AutoCompletion feature	
AutoCompletionText	True	custom://Bindable	Bindable property for AutoCompletion text to be used when AutoCompletion is enabled	TC_W16_16
AutoCompletionTextFont	False	Font	Font for AutoCompletionText	
AutoCompletionTextNode	False	Node2D	AutoCompletionText node	
BackgroundHeight	False	UInt	Height of EditField Background	
BackgroundNode	False	Node2D	Background image node	
BackgroundWidth	False	UInt	Width of EditField Background	
BlinkInterval	False	UInt	Specify the interval in msec for blinking of cursor	

ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time
CultureDependentAlignment	False	Bool	Default = True. The text and cursor changes according to text direction. Setting to FALSE makes edit field works like Latin [Left to Right]. Applicable to entry text and match mode text and has no impact for Highlight text, Auto complete texts.	
CursorAnimationPath	False	custom://Specify the Candera path of the cursor animation to be used for blinking of cursor		
CursorBgBitmap	True	Image2D	Cursor Bitmap; Bitmap to be set on the BitmapBrushEffect inside the RenderNode	TC_W16_14
CursorNode	False	Node2D	Cursor image node	
DefaultDisabledTextColor	False	Color	DEPRECATED! Don't use this property. Even if the value is set, there is no effect on EditFieldWidget.	
DefaultEnabledTextColor	False	Color	DEPRECATED! Don't use this property. Even if the value is set, there is no effect on EditFieldWidget.	
DisabledAutoCompletionTextColor	True	Color	Color for AutoCompletionText in disabled state	TC_W16_16
DisabledBgBitmap	True	Image2D	Disabled Background; Bitmap to be set on the BitmapBrushEffect inside the RenderNode	TC_W16_14
DisabledEntryTextColor	True	Color	Color for EntryText in disabled state	TC_W16_11
DisabledInstructionTextColor	True	Color	Color for InstructionText in Disabled state	TC_W16_05
DisabledMatchTextColor	True	Color	Color for MatchText in disabled state	TC_W16_15
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: EditField-Widget2D does not support to test this property.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: EditField-Widget2D does not support to test this property.



Drag	True	Bool	Enable drag gesture detection	Ignored: EditField-Widget2D does not support to test this property.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
EditFieldInfo	True	custom://	Bindable property to send the EditField Text to other components	TC_W16_12
EditFieldTextDirection	False	Enum	Utilized for text flow direction. Legacy : old behavior, Implicit: depends on character, LeftToRight: As LTR, RightToLeft: As RTL, Culture: As culture direction, Node: As Layout Direction	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W16_08
EnabledAutoCompletionTextColor	True	Color	Color for AutoCompletionText in enabled state	TC_W16_16
EnabledEntryTextColor	True	Color	Color for EntryText in enabled state	TC_W16_10
EnabledInstructionTextColor	True	Color	Color for InstructionText in enabled state	TC_W16_06
EnabledMatchTextColor	True	Color	Color for MatchText in enabled state	TC_W16_15
EndTextNode	False	Node2D	EndText node	
EntryTextFont	False	Font	Font for EntryText	
EntryTextNode	False	Node2D	EntryText node	
EntryTextReceived	True	custom://	Bindable property to receive Entry Text string if Configured in Match Mode	TC_W16_12
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.

FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: EditField-Widget2D does not support to test gesture simulations.
GlyphSpacing	False	UShort	Property to add space between the characters in pixels	
HighlightCount	True	UInt	Bindable property to receive Highlight Count if Configured in free Mode	TC_W16_02
HighlightDisabledTextColor	True	Color	Default Color for Highlight Text in Disabled state	TC_W16_11
HighlightEnabledTextColor	True	Color	Default Color for Highlight Text in Enabled state	TC_W16_10
HighlightStartIndex	True	UInt	Bindable property to receive Highlight Start Index if Configured in free Mode	TC_W16_03
HighlightTextNode	False	Node2D	HighlightText node	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property is dependent on EnableGroupWidget2D, so it will be tested in EnableGroupWidget2D section.
InstructionText	True	custom://Bindable	Bindable property to set the instruction text that will be displayed.	TC_W16_13
InstructionTextFont	False	Font	Font for InstructionText	
KeyReceived	True	custom://Bindable	Bindable property to receive key press inputs for the EditField text	TC_W16_12
LineSpacing	False	UInt	LineSpacing in MultiLine Text for the Editfield	

MaskChar	False	custom://Mask	Mask char if masking is required, else leave it blank	
MatchTextFont	False	Font	Font for MatchText	
MatchTextNode	False	Node2D	MatchText node	
MaxTextLength	False	UInt	Maximum acceptable length of EditField text	
MaximumNumberOfLines	False	UInt	MaximumNumberOfLines for the Editfield to take input in multiple lines upto value set in MaximumNumberOfLines	
MultiLineLayouting	False	Bool	MultiLine Layouting for the Editfield to take input in multiple lines.	
Name	False	CharArray	The name of the widget instance	
NewCursorPosition	True	UInt	Bindable property to receive NewCursorPosition if Configured in free Mode	TC_W16_01
Node	False	Node2D	The associated node of the widget.	
NormalBgBitmap	False	Image2D	Enabled Background; Bitmap to be set on the BitmapBrushEffect inside the RenderNode	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	Ignored: EditField-Widget2D does not support to test this property.
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: EditField-Widget2D does not support to test this property.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
RestrictShiftOperations	False	Bool	Restricts the movement of cursor in highlight area only in EditField text	

Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SelectedBgBitmap	True	Image2D	Selected Background; Bitmap to be set on the BitmapBrushEffect inside the RenderNode	TC_W16_14
SuggestionText	True	custom://Bindable	Bindable property for suggestion text to be used if Configured in Match Mode	TC_W16_15
Swipe	True	Bool	Enable swipe gesture detection	Ignored: EditField-Widget2D does not support to test this property.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: EditField-Widget2D does not support to test this property.
TextAreaMarginX	False	UInt	X-Offset from Background Image for TextArea	
TextAreaMarginY	False	UInt	Y-Offset from Background Image for TextArea	
TextStyle	False	TextStyle	The TextStyle used for texts like EntryText, matchText etc. Individual fonts will be used if TextStyle is empty.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W16_07
TruncationText	True	custom://Bindable	Bindable property to set the truncation text that will be displayed.	TC_W16_04
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W16_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
WidgetTextMode	False	Enum	Operation Mode for the EditFieldWidget	

## 36.2 Message list

Name	Subscriber	Description	Members	Distribution	Test Scope
EditFieldAutoCompleteMsg	Model Updator	<p>Updator Message from EditFiled to notify the subscribers about the various properties/values change</p> <ul style="list-style-type: none"> <li>- This message is applicable for FreeMode</li> <li>- ViewId: corresponds to the view from where the message is fired</li> <li>- Sender: corresponds to the widget instance which fires the message</li> </ul>			TC_W16_19
EditFieldHighlightRelatedOperationMsg	Renderable View	<p>Courier Message from Model to Widget to provide Highlight Related Operation on Highlight Text</p> <ul style="list-style-type: none"> <li>- This message is applicable for FreeMode</li> <li>- ViewId: corresponds to the view from where the message is fired</li> <li>- Sender: corresponds to the widget instance which fires the message</li> <li>- Action: corresponds to the Action( Delete, Replace, Select)</li> <li>- Text: corresponds to the text for Replace Action (used by Replace)</li> <li>- Index: corresponds to the Index (used by Select Action)</li> <li>- Count: corresponds to the Count (used by Select Action)</li> </ul>	<p>Action: enEditFieldTextAction::Enum.</p> <p>Text: Candra::String.</p> <p>Index: Courier::UInt8.</p> <p>Count: Courier::UInt8.</p>		TC_W16_17

EditFieldUpdMsg	Model	<p>Courier Message from EditFiled to notify the subscribers about the various properties/values change</p> <ul style="list-style-type: none"> <li>- ViewId: corresponds to the view from where the message is fired</li> <li>- Sender: corresponds to the widget instance which fires the message</li> <li>- Text: corresponds to the text in m_sEdifieldIfo, This is total text including Left,Hightlight and End Text</li> <li>- CursorIndex: corresponds to the CursorIndex</li> <li>- VisibleStartIndex: corresponds to the VisibleStartIndex</li> <li>- VisibleCharCount: corresponds to the VisibleCharCount</li> <li>- HighlightStartIndex: corresponds to HighlightStartIndex</li> <li>- HighlightCount: corresponds to the HighlightCount</li> </ul>	<p>Text: Can-dera::String.</p> <p>CursorIndex: Courier::UInt8.</p> <p>VisibleStartIndex: Courier::UInt8.</p> <p>VisibleCharCount: Courier::UInt8.</p> <p>HighlightStartIndex: Courier::UInt8.</p> <p>HighlightCount: Courier::UInt8.</p> <p>EffectiveRendering: bool.</p>	TC_W16_18
-----------------	-------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------

## Chapter 37

# EffectControlWidget2D

**Name:** EffectControlWidget2D

**Description:** Controls the properties of an effect allowing to expose them outside composites or to change them using data binding sources.

**Category:** Common

### 37.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
BlurFilterSize	True	Byte	See Effect's property BlurFilterSize	TC_W68_01
BlurFilterSizeEnabled	False	Bool	Enables configuration of the property BlurFilterSize	
Color	True	Color	See Effect's property Color	TC_W68_02
ColorEnabled	False	Bool	Enables configuration of the property Color	
DropShadowColor	True	Color	See Effect's property DropShadowColor	TC_W68_03
DropShadowColorEnabled	False	Bool	Enables configuration of the property DropShadowColor	
DropShadowDistance	True	UInt	See Effect's property DropShadowDistance	TC_W68_03
DropShadowDistanceEnabled	False	Bool	Enables configuration of the property DropShadowDistance	
DropShadowEnabled	True	Bool	See Effect's property DropShadowEnabled	TC_W68_03
DropShadowEnabledEnabled	False	Bool	Enables configuration of the property DropShadowEnabled	
DropShadowLightAngle	True	Int	See Effect's property DropShadowLightAngle	TC_W68_03
DropShadowLightAngleEnabled	False	Bool	Enables configuration of the property DropShadowLightAngle	
DropShadowScale	True	UInt	See Effect's property DropShadowScale	TC_W68_03
DropShadowScaleEnabled	False	Bool	Enables configuration of the property DropShadowScale	

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FlipH	True	Bool	See Effect's property FlipH	TC_W68_04
FlipHEnabled	False	Bool	Enables configuration of the property FlipH	
FlipV	True	Bool	See Effect's property FlipV	TC_W68_04
FlipVEnabled	False	Bool	Enables configuration of the property FlipV	
GradientCenter	True	Vector2	See Effect's property GradientCenter	TC_W68_11
GradientCenterColor	True	Color	See Effect's property GradientCenterColor	TC_W68_12
GradientCenterColorEnabled	False	Bool	Enables configuration of the property GradientCenterColor	
GradientCenterEnabled	False	Bool	Enables configuration of the property GradientCenter	
GradientDirection	True	Vector2	See Effect's property GradientDirection	TC_W68_13
GradientDirectionEnabled	False	Bool	Enables configuration of the property GradientDirection	
GradientMagnitude	True	Float	See Effect's property GradientMagnitude	TC_W68_11
GradientMagnitudeEnabled	False	Bool	Enables configuration of the property GradientMagnitude	
GradientNegativeColor	True	Color	See Effect's property GradientNegativeColor	TC_W68_13
GradientNegativeColorEnabled	False	Bool	Enables configuration of the property GradientNegativeColor	
GradientPositiveColor	True	Color	See Effect's property GradientPositiveColor	TC_W68_13
GradientPositiveColorEnabled	False	Bool	Enables configuration of the property GradientPositiveColor	
GradientRadialColor	True	Color	See Effect's property GradientRadialColor	TC_W68_12
GradientRadialColorEnabled	False	Bool	Enables configuration of the property GradientRadialColor	
GradientSize	True	Vector2	See Effect's property GradientSize	TC_W68_11
GradientSizeEnabled	False	Bool	Enables configuration of the property GradientSize	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
MaskImage	True	Image2D	See Effect's property MaskImage	Ignored: Candera does not support to change image for this property



MaskImageEnabled	False	Bool	Enables configuration of the property MaskImage	
MaskNode	False	Node2D	See Effect's property MaskNode	
MaskNodeEnabled	False	Bool	Enables configuration of the property MaskNode	
MirrorAlpha	True	Float	See Effect's property MirrorAlpha	TC_W68_06
MirrorAlphaEnabled	False	Bool	Enables configuration of the property MirrorAlpha	
MirrorAxisFrom	True	Vector2	See Effect's property MirrorAxisFrom	TC_W68_06
MirrorAxisFromEnabled	False	Bool	Enables configuration of the property MirrorAxisFrom	
MirrorAxisTo	True	Vector2	See Effect's property MirrorAxisTo	TC_W68_06
MirrorAxisToEnabled	False	Bool	Enables configuration of the property MirrorAxisTo	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OutlineColor	True	Color	See Effect's property OutlineColor	TC_W68_05
OutlineColorEnabled	False	Bool	Enables configuration of the property OutlineColor	
OutlineWidth	True	Byte	See Effect's property OutlineWidth	TC_W68_05
OutlineWidthEnabled	False	Bool	Enables configuration of the property OutlineWidth	
ShadowColor	True	Color	See Effect's property ShadowColor	TC_W68_10
ShadowColorEnabled	False	Bool	Enables configuration of the property ShadowColor	
ShadowOffset	True	Vector2	See Effect's property ShadowOffset	TC_W68_10
ShadowOffsetEnabled	False	Bool	Enables configuration of the property ShadowOffset	
ShadowScale	True	Vector2	See Effect's property ShadowScale	TC_W68_10
ShadowScaleEnabled	False	Bool	Enables configuration of the property ShadowScale	
ShearAngleX	True	Float	See Effect's property ShearAngleX	TC_W68_14
ShearAngleXEnabled	False	Bool	Enables configuration of the property ShearAngleX	
ShearAngleY	True	Float	See Effect's property ShearAngleY	TC_W68_14
ShearAngleYEnabled	False	Bool	Enables configuration of the property ShearAngleY	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W68_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

37.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 38

# EnableGroupWidget2D

**Name:** EnableGroupWidget2D

**Description:** Provides support to enable or disable descendant widgets.

**Category:** Common

### 38.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W10_08
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	TC_W10_07
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W10_09

VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
----------------	-------	------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

38.2 Message list

<i>Name</i>	<i>Subscribes</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	-------------------	--------------------	----------------	---------------------	-------------------

## Chapter 39

# FlexBaseLineLayouterWidget2D

**Name:** FlexBaseLineLayouterWidget2D

**Description:** DEPRECATED! Don't use this widget! Use BaseLineLayouterWidget2D instead.

**Category:** Deprecated

### 39.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
BaseLineOffset	True	Float	Baseline offset used by the layouter.	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex

Name	False	CharArray	The name of the widget instance	
NestedLevel	True	UInt	<p>DEPRECATED: This property was required in the past when it was not possible to associate a widget directly to the composite.</p> <p>The value of this property represents the ancestor's level, relative to the associated node, that will be targeted by this widget. This property should not be used because it creates confusion and can lead to errors. Just associate the widget directly to the targeted composite or group.</p>	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	<p>Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.</p>	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex
VisibleEnabled	False	Bool	<p>Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.</p>	

## 39.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 40

# FlexCanvasLayouterWidget2D

**Name:** FlexCanvasLayouterWidget2D

**Description:** DEPRECATED! Don't use this widget! Use an overlay layouter instead.

**Category:** Deprecated

### 40.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex
LayoutChildren	True	Bool	The value that will be set for the LayoutChildren property of the ancestor's FlexCanvasLayouter instance.	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex

Name	False	CharArray	The name of the widget instance	
NestedLevel	True	UInt	The ancestor's level, relative to the associated node, that will have the layouter set to an FlexCanvasLayouter.	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget but without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 40.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------



## Chapter 41

# FlexCollapseWidget2D

**Name:** FlexCollapseWidget2D

**Description:** DEPRECATED! Don't use this widget! Instead set the Collapsible property of the node and hide it using the Visible property of any widget attached to it.

**Category:** Deprecated

### 41.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Collapsed	True	Bool	Whether the ancestor will be collapsed or not.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Name	False	CharArray	The name of the widget instance	
NestedLevel	True	UInt	The ancestor's level, relative to the associated node, that will be collapsed.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 41.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 42

# FlexDockSideWidget2D

**Name:** FlexDockSideWidget2D

**Description:** DEPRECATED! Don't use this widget!

**Category:** Deprecated

### 42.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
DockSide	True	Enum	The value to set for the DockSide property of the ancestor node.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

Name	False	CharArray	The name of the widget instance	
NestedLevel	True	UInt	The ancestor's level, relative to the associated node, that will have the DockSide property modified.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 42.2 Message list

<i>Name</i>	<i>Subscribable</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 43

# FlexDropDownListWidget2D

**Name:** FlexDropDownListWidget2D

**Description:** DEPRECATED! Don't use this widget! Use DropDownListWidget2D instead.

**Category:** Deprecated

### 43.1 Property list

Name	Bindable	Type	Description	Test Scope
AcceptImmediatePositioning	True	Bool	If set to true, immediate positioning is taken into consideration. Please check the other properties to enable immediate positioning for different events.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
AcceptImmediatePositioningOnFirstAppearance	True	Bool	If set to true, the first ListChangeMsg with ListChangeSet that is received after rendering is enabled for the view and if immediate positioning is activated, then the position is set to the received value without using an animation; further position setting uses an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

AcceptImmediatePositioningOnItemScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangeDown or ListChangeUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
AcceptImmediatePositioningOnPageScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangePageDown or ListChangePageUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
AcceptImmediatePositioningOnPositionSet	True	Bool	If set to true, if a ListChangeMsg with ListChangeSet is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
AnimationOffset	False	Short	Offset from first visible item to the position of the first animated item, give a negative value for the offset to the first invisible item of the list	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
AutoViewportSize	False	Bool	If enabled, the size of the viewport will be calculated as NumberOfItems multiplied with the size of the InvalidItemTemplate. This is done for the direction of scrolling; the other direction will remain unmodified.	
BufferSize	True	UInt	Used for windowed data. The buffer size of the elements requested prior and after the visible ones.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
ButtonNode	False	Node2D	The node acting as a button and when pressed will open the drop down list.	

CachedLayout	False	Bool	Caches the layout of the list entries, if set to true the layout calculation is done only once for each list item and then cached.	
ConfigureFocusGroup	True	Bool	Focus group is configured explicitly or uses default configuration.	Ignored: The TTF is simulation for the gesture action is not ready yet.
ContentNode	False	Node2D	The node containing the selected item.	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time
Coverflow	True	Bool	True if the list is a coverflow.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
CustomAnimationsGroupNode	False	Node2D	Root node of animated nodes that are placeholders for the real items in the list. The animated properties will be applied to the nodes that will be added by the list to the ItemsNode.	
CustomListAnimations	False	Animation	Array of animations for all visible list elements.	
DefaultFocusOrder	True	Short	Focus order of the element which should become focused when this group becomes active.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DragAndSwipeOnNeed	True	Bool	If true then drag and swipe will be dynamically disabled when a new touch session starts if all items are visible.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DynamicGrid	False	Bool	Whether or not the grid is automatically adapting to the text size.	
DynamicGridEmptyCellsPolicy	False	Enum	The policy to distribute the empty spaces in the last row of the list	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex



ExpandAutoScroll	True	Bool	Whether or not expand animations scroll the list so that the expanded item fully fits inside the viewport.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
FixedPageScrolling	True	Bool	Works only if all the data is available (no windowed data). Should be used only with small lists. If set to true, the pages always start at a fixed position, similar to a book. If set to false, the pages are always considered relative to the current start index.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
FixedPageScrollingOffset	True	Float	Percentage of the viewport needed to be scrolled/swiped with the touch in order to go to the next/previous page.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
FocusAnchorId	True	UInt		Ignored: Not available in CGI configuration, not mentioned in user guide
FocusAnchorMode	False	Enum		
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTF is simulation for the gesture action is not ready yet.
FocusLayer	True	Short	Specifies to which focus layer this group belongs. There should be distinct layers for main surface screens, subspeller, popups to prevent focusing of elements in the main screen when a popup is displayed. Only elements in the groups with the highest layer will be focused.	Ignored: The TTF is simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTF is simulation for the gesture action is not ready yet.

FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
FocusWrapAround	True	Bool	Focus wraps around to first/last element.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedIndex	True	Int	The index of the currently focused item.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedNode	False	Node2D	The node that will highlight the focused node.	
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
IgnoreListChangeMsg	False	Enum	Defines when the ListChangeMsg will be ignored during the list scrolling or swiping	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
InvalidItemTemplate	False	Node2D	Template used for not yet available data or if no other template is found for some data.	
ItemsNode	False	Node2D	The list items will be added as children of this node.	

LimitExceededDownAnimation	False	Animation	Animation that is played when the user wants to exceed the list on bottom. It is only played if LimitExceededDownAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitExceededDownAnimationTime	True	UInt	Defines the how much time (ms) the exceeded down animation takes.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitExceededDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on bottom of the list.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitExceededDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitExceededUpAnimation	False	Animation	Animation that is played when the user wants to exceed the list on top. It is only played if LimitExceededUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitExceededUpAnimationTime	True	UInt	Defines the how much time (ms) the exceeded up animation takes.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitExceededUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on top of the list.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

LimitExceededUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitReachedDownAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedDownAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedDownAnimationTime	True	UInt	Defines the how much time (ms) the reached down animation takes.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitReachedDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on bottom of the list.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitReachedDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitReachedUpAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedUpAnimationTime	True	UInt	Defines the how much time (ms) the reached up animation takes.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

LimitReachedUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on top of the list.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitReachedUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
ListAlignment	False	Enum	Only applicable for lists with less then the maximum number of visible items. If set to 'End' the items will be aligned at the end of the list.	
ListId	True	UInt	A unique identifier of the list that will be used to request data to the model and react to data sent from the model.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
ListNode	False	Node2D	The node containing the dropdown list.	
ListPosition	False	Vector2	Value to be set for ListNode if UserListPosition is true.	
ListSize	False	Vector2	The size to be set for the ListNode.	
LockOutFixMovementOffset	False	Float	Offset in percent of the viewport required for LockOutType LockOutFixMovement.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NumberOfItems	True	UInt	The number of items is only available for itemwise lists(PixelWiseScrollingEnabled set to false). For stack layouted lists it has the meaning of the number of elements that would be visible. For grid layouted lists it has the meaning of the number of columns(for horizontal scrolling) or rows(for vertical scrolling) that will be seen. The number of rows(for horizontal scrolling) or columns(for vertical scrolling) will be taken from the GridLayouter.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PageIndicationAccuracy	False	Float	Accuracy of the current page position indication.	

PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PixelWiseAnimationTime	True	UInt	Time in ms to finish the pixel wise animation	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PixelWiseScrollingEnabled	True	Bool	If set to true, scrolling is done pixel by pixel, meaning that an item can be partially visible. If set to false, scrolling is done item by item; items are always completely visible.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PostListChanged	False	Bool	If true, the list will post a ListChangedUpdMsg message everytime the first visible index changes.	
PreparedItemsBufferSize	False	Byte	Number of additional items being prepared when the list is idle	
PreparedItemsUpdateTriggerOffset	False	Byte	Offset to the edge of the PreparedItemsBuffer - the prepared item buffer will be updated and old items discarded as soon as this offset is reached	
PreserveFocus	True	Bool	Current focused element is preserved when group becomes inactive.	Ignored: The TTF is simulation for the gesture action is not ready yet.
PreserveScrollIndex	False	Bool	If enabled, and the list is cached, the widget will remember the last scrolled position and set the same on re-entering the List View, unless set to a different value by the model.	
PressHold	True	Bool	Enable hold gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollAnimationInput	False	Enum	Input of scroll animation - Either pixel position or index of list items	
ScrollbarAlwaysVisible	True	Bool	If set to true, scrollbar is always visible, regardless if it's needed or not. If set to false, it's only visible if needed.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
ScrollingOrientation	True	Enum	Scrolling orientation: vertical or horizontal	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
ScrollingType	False	Enum	Sets the way this list scrolls: Default beginning to end, continuous or last page not filled.	
ShortPixelWiseAnimationTime	True	UInt	Short time in ms to finish the pixel wise animation; used when setting the position with a ListChangeSet request type	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

ShowItemsOnViewActivated	False	Bool	Whether items are visible or not on view shown, before receiving a ListCustomAnimationReqMsg.	
Snap	False	Enum	Specifies how the list should behave after scrolling. SnapToEdge - List will snap to the nearest edge, SnapToCenter - List will snap to the center	
SnapHoldTime	False	UInt	Timeout in milliseconds, if reached no snapping will be done	
SnapOffsetThreshold	False	UInt	Threshold of offsetDelta = Abs(offsetTop - offsetBottom) in pixels - if the offset delta is smaller than the threshold the snap will be done to the edge closer to the touch position	
StartIndex	True	Int	The index of the first visible item. Negative indices are accepted in case of circular scrolling.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Swipe	True	Bool	Enable swipe gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipeDirection	False	Enum	Direction in which swipe should be detected	
SwipingAcceleration	True	Float	Acceleration used to slow down swiping	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipingDistanceThreshold	True	UInt	Deprecated. Please do not use this, as it will be removed in a future version. Minimum distance in display units needed to drag in order to begin scrolling	Ignored: Deprecated
SwipingMaxDistance	True	Float	Maximum distance that can be covered within one swipe	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex



SwipingTimerThreshold	True	UInt	Minimum time in ms needed to press in order to focus an item	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipingVelocityThreshold	True	Float	Deprecated. Please do not use this, as it will be removed in a future version. Minimum velocity in display units/ms needed to move with in order to begin swiping	Ignored: Deprecated
Tap	True	Bool	Enable press and tap gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
TemplateGroup	False	Node2D	The parent of template nodes. Template nodes will be cloned based on the item needed to be displayed and the clone will be added to the ItemsNode.	
TemplateScrollAnimations	False	Animation	Array of template animations for list scrolling.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
UpdateTriggerOffset	True	UInt	Used for windowed data. The offset relative to the first or last visible item index, respectively used to request new data when reached.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

UsableViewportPadding	False	Margin	Viewport offset for visible items in the list - useful in case the list is overlapped by some other content.	
UseListPosition	False	Bool	If set to true, sets the position given by ListPosition property for the ListNode. If set to false, it uses the position set by the layout.	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 43.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 44

# FlexListWidget2D

**Name:** FlexListWidget2D

**Description:** DEPRECATED! Don't use this widget! Use ListWidget2D instead.

**Category:** Deprecated

### 44.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AcceptImmediatePositioning	True	Bool	If set to true, immediate positioning is taken into consideration. Please check the other properties to enable immediate positioning for different events.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
AcceptImmediatePositioningOnFirstAppearance	True	Bool	If set to true, the first ListChangeMsg with ListChangeSet that is received after rendering is enabled for the view and if immediate positioning is activated, then the position is set to the received value without using an animation; further position setting uses an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

AcceptImmediatePositioningOnItemScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangeDown or ListChangeUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
AcceptImmediatePositioningOnPageScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangePageDown or ListChangePageUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
AcceptImmediatePositioningOnPositionSet	True	Bool	If set to true, if a ListChangeMsg with ListChangeSet is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
AnimationOffset	False	Short	Offset from first visible item to the position of the first animated item, give a negative value for the offset to the first invisible item of the list	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
AutoViewportSize	False	Bool	If enabled, the size of the viewport will be calculated as NumberOfItems multiplied with the size of the InvalidItemTemplate. This is done for the direction of scrolling; the other direction will remain unmodified.	
BufferSize	True	UInt	Used for windowed data. The buffer size of the elements requested prior and after the visible ones.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
CachedLayout	False	Bool	Caches the layout of the list entries, if set to true the layout calculation is done only once for each list item and then cached.	

ConfigureFocusGroup	True	Bool	Focus group is configured explicitly or uses default configuration.	Ignored: The TTF is simulation for the gesture action is not ready yet.
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time
Coverflow	True	Bool	True if the list is a coverflow.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
CustomAnimationsGroupNode	False	Node2D	Root node of animated nodes that are placeholders for the real items in the list. The animated properties will be applied to the nodes that will be added by the list to the ItemsNode.	
CustomListAnimations	False	Animation	Array of animations for all visible list elements.	
DefaultFocusOrder	True	Short	Focus order of the element which should become focused when this group becomes active.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.

Drag	True	Bool	Enable drag gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DragAndSwipeOnNeed	True	Bool	If true then drag and swipe will be dynamically disabled when a new touch session starts if all items are visible.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DynamicGrid	False	Bool	Whether or not the grid is automatically adapting to the text size.	
DynamicGridEmptyCellsPolicy	False	Enum	The policy to distribute the empty spaces in the last row of the list	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
ExpandAutoScroll	True	Bool	Whether or not expand animations scroll the list so that the expanded item fully fits inside the viewport.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

FixedPageScrolling	True	Bool	Works only if all the data is available (no windowed data). Should be used only with small lists. If set to true, the pages always start at a fix position, similar to a book. If set to false, the pages are always considered relative to the current start index.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
FixedPageScrollingOffset	True	Float	Percentage of the viewport needed to be scrolled/swiped with the touch in order to go to the next/previous page.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
FocusAnchorId	True	UInt		Ignored: Not available in CGI configuration, not mentioned in user guide
FocusAnchorMode	False	Enum		
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusLayer	True	Short	Specifies to which focus layer this group belongs. There should be distinct layers for main surface screens, subspeller, popups to prevent focusing of elements in the main screen when a popup is displayed. Only elements in the groups with the highest layer will be focused.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
FocusWrapAround	True	Bool	Focus wraps around to first/last element.	Ignored: The TTFis simulation for the gesture action is not ready yet.

Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedIndex	True	Int	The index of the currently focused item.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedNode	False	Node2D	The node that will highlight the focused node.	
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
IgnoreListChangeMsg	False	Enum	Defines when the ListChangeMsg will be ignored during the list scrolling or swiping	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
InvalidItemTemplate	False	Node2D	Template used for not yet available data or if no other template is found for some data.	
ItemsNode	False	Node2D	The list items will be added as children of this node.	
LimitExceededDownAnimation	False	Animation	Animation that is played when the user wants to exceed the list on bottom. It is only played if LimitExceededDownAnimationType is CustomAnimation or circular scrolling is not enabled.	



LimitExceededDownAnimationTime	True	UInt	Defines the how much time (ms) the exceeded down animation takes.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitExceededDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on bottom of the list.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitExceededDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitExceededUpAnimation	False	Animation	Animation that is played when the user wants to exceed the list on top. It is only played if LimitExceededUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitExceededUpAnimationTime	True	UInt	Defines the how much time (ms) the exceeded up animation takes.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitExceededUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on top of the list.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitExceededUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.

LimitReachedDownAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedDownAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedDownAnimationTime	True	UInt	Defines the how much time (ms) the reached down animation takes.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitReachedDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on bottom of the list.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitReachedDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitReachedUpAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedUpAnimationTime	True	UInt	Defines the how much time (ms) the reached up animation takes.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LimitReachedUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on top of the list.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

LimitReachedUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
ListAlignment	False	Enum	Only applicable for lists with less than the maximum number of visible items. If set to 'End' the items will be aligned at the end of the list.	
ListId	True	UInt	A unique identifier of the list that will be used to request data to the model and react to data sent from the model.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
LockOutFixMovementOffset	False	Float	Offset in percent of the viewport required for LockOutType LockOutFixMovement.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NumberOfItems	True	UInt	The number of items is only available for itemwise lists(PixelWiseScrollingEnabled set to false). For stack layouted lists it has the meaning of the number of elements that would be visible. For grid layouted lists it has the meaning of the number of columns(for horizontal scrolling) or rows(for vertical scrolling) that will be seen. The number of rows(for horizontal scrolling) or columns(for vertical scrolling) will be taken from the GridLayouter.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PageIndicationAccuracy	False	Float	Accuracy of the current page position indication.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PixelWiseAnimationTime	True	UInt	Time in ms to finish the pixel wise animation	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

PixelWiseScrollingEnabled	True	Bool	If set to true, scrolling is done pixel by pixel, meaning that an item can be partially visible. If set to false, scrolling is done item by item; items are always completely visible.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PostListChanged	False	Bool	If true, the list will post a ListChangedUpdMsg message everytime the first visible index changes.	
PreparedItemsBufferSize	False	Byte	Number of additional items being prepared when the list is idle	
PreparedItemsUpdateTriggerOffset	False	Byte	Offset to the edge of the PreparedItemsBuffer - the prepared item buffer will be updated and old items discarded as soon as this offset is reached	
PreserveFocus	True	Bool	Current focused element is preserved when group becomes inactive.	Ignored: The TTF is simulation for the gesture action is not ready yet.
PreserveScrollIndex	False	Bool	If enabled, and the list is cached, the widget will remember the last scrolled position and set the same on re-entering the List View, unless set to a different value by the model.	
PressHold	True	Bool	Enable hold gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.

Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollAnimationInput	False	Enum	Input of scroll animation - Either pixel position or index of list items	
ScrollbarAlwaysVisible	True	Bool	If set to true, scrollbar is always visible, regardless if it's needed or not. If set to false, it's only visible if needed.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
ScrollingOrientation	True	Enum	Scrolling orientation: vertical or horizontal	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
ScrollingType	False	Enum	Sets the way this list scrolls: Default beginning to end, continuous or last page not filled.	
ShortPixelWiseAnimationTime	True	UInt	Short time in ms to finish the pixel wise animation; used when setting the position with a ListChangeSet request type	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
ShowItemsOnViewActivated	False	Bool	Whether items are visible or not on view shown, before receiving a ListCustomAnimationReqMsg.	
Snap	False	Enum	Specifies how the list should behave after scrolling. SnapToEdge - List will snap to the nearest edge, SnapToCenter - List will snap to the center	
SnapHoldTime	False	UInt	Timeout in milliseconds, if reached no snapping will be done	
SnapOffsetThreshold	False	UInt	Threshold of offsetDelta = Abs(offsetTop - offsetBottom) in pixels - if the offset delta is smaller than the threshold the snap will be done to the edge closer to the touch position	

StartIndex	True	Int	The index of the first visible item. Negative indices are accepted in case of circular scrolling.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Swipe	True	Bool	Enable swipe gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipeDirection	False	Enum	Direction in which swipe should be detected	
SwipingAcceleration	True	Float	Acceleration used to slow down swiping	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipingDistanceThreshold	True	UInt	Deprecated. Please do not use this, as it will be removed in a future version. Minimum distance in display units needed to drag in order to begin scrolling	Ignored: Deprecated
SwipingMaxDistance	True	Float	Maximum distance that can be covered within one swipe	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipingTimerThreshold	True	UInt	Minimum time in ms needed to press in order to focus an item	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipingVelocityThreshold	True	Float	Deprecated. Please do not use this, as it will be removed in a future version. Minimum velocity in display units/ms needed to move with in order to begin swiping	Ignored: Deprecated

Tap	True	Bool	Enable press and tap gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
TemplateGroup	False	Node2D	The parent of template nodes. Template nodes will be cloned based on the item needed to be displayed and the clone will be added to the ItemsNode.	
TemplateScrollAnimations	False	Animation	Array of template animations for list scrolling.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
UpdateTriggerOffset	True	UInt	Used for windowed data. The offset relative to the first or last visible item index, respectively used to request new data when reached.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
UsableViewportPadding	False	Margin	Viewport offset for visible items in the list - useful in case the list is overlapped by some other content.	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

44.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------



## Chapter 45

# FlexOverlayLayoutWidget2D

**Name:** FlexOverlayLayoutWidget2D

**Description:** DEPRECATED! Don't use this widget! Use OverlayLayoutWidget2D instead.

**Category:** Deprecated

### 45.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Name	False	CharArray	The name of the widget instance	

NestedLevel	True	UInt	DEPRECATED: This property was required in the past when it was not possible to associate a widget directly to the composite. The value of this property represents the ancestor's level, relative to the associated node, which will be targeted by this widget. This property should not be used because it creates confusion and can lead to errors. Just associate the widget directly to the targeted composite.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 45.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 46

# FlexScrollBar2D

**Name:** FlexScrollBar2D

**Description:** DEPRECATED! Don't use this widget! Use ScrollBarWidget2D instead.

**Category:** Deprecated

### 46.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AdditionalPageBackwardNode	False	Node2D	An auxiliary node acting as a page backward button. This node is not part of the knob movement area.	
AdditionalPageButtons	False	Bool	Use additional page buttons that are not resized and are not part of the knob movement area.	
AdditionalPageForwardNode	False	Node2D	An auxiliary node acting as a page forward button. This node is not part of the knob movement area.	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
BackwardNode	False	Node2D	The node acting as back button.	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time

DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DynamicSliderSize	False	Bool	Whether or not the slider size shall reflect the number of items in the list, if set to true the specified slider size will represent the minimum size.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
ForwardNode	False	Node2D	The node acting as forward button.	
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
KnobDragMode	False	Enum	The mode the knob/slider moves when dragged: ListPosition follows the list current position relative to its whole content; PointerPosition follows the pointer.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

OverscrollSize	True	Float	The percentage of one of the overscroll areas relative to the size of the slider back. There are two regions of overscroll: one at the beginning of the scrollbar and one at the end.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PageBackwardNode	False	Node2D	The node acting as a page back button; the part between the knob and BackwardNode.	
PageForwardNode	False	Node2D	The node acting as a page forward button; the part between the knob and ForwardNode.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
RepeatMessage	False	UInt	The time between triggering two consecutive events(back, forward, pageBack or pageForward) when the corresponding button is kept pressed.	

Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollableNode	False	Node2D	The node containing a FlexScrollable widget.	
SliderNode	False	Node2D	The node that will act as a slider and can be dragged. Its parent must be the ancestor of the PageBackwardNode and PageForwardNode.	
SliderSize	True	Float	The percentage of the slider size relative to the size of the slider back.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Swipe	True	Bool	Enable swipe gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.

Touchable	True	Bool	Widget is Touchable or not	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
TrackPressMode	False	Enum	Specifies the scrollbar behavior when it is touched either above or below the knob - PageScroll: The list is scrolled one page up / down, JumpToTouchCoordinate: Causes the knob to jump to the touch coordinate	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 46.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------



## Chapter 47

# FlexScrollBarButton2D

**Name:** FlexScrollBarButton2D

**Description:** DEPRECATED! Don't use this widget! Use ScrollBarButtonWidget2D instead.

**Category:** Deprecated

### 47.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Amount	False	Int	The amount for the change type. For example, if set to 2 and ChangeType is set to Previous, it will request an action to go 2 items back.	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ChangeType	False	Enum	The type of action that will be issued to the scrollbar.	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time

DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
EditorFocused	False	Bool	Sets the focused flag in order to preview the appearance in SceneComposer. It has no effect in the simulation or on the target.	
EditorPressed	False	Bool	Sets the pressed flag in order to preview the appearance in SceneComposer. It has no effect in the simulation or on the target.	

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

IsActive	True	Bool	Indicates if the button is active. For toggle/radio button active means checked/selected. For normal button active means that the option/action is active (for example track is playing, station is tuned, etc).	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OnPress	False	Bool	If set to true, then the action is done when pressing the button, otherwise it's done when releaseing the button.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PostButtonReactionMsg	False	Bool	ButtonReactionMsg messages will be posted only if this property is true.	
PressHold	True	Bool	Enable hold gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.

Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
TimerConfiguration	False	Enum	Configures the timer.	
TouchHandler	True	Int	Specifies the id of the touch handler for this button. A touch handler can provide a custom sensitive area for buttons (for example circle, triangle, etc). Touch handlers are registered at application start-up.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

TouchableArea	False	Rectangle	Configures the touchable area of the button. X and Y are coordinates relative to the upper left corner of the node effective bounding rectangle. If width or height are zero or negative the effective bounding rectangle of the node will be used as touchable area (X and Y will also be ignored).	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 47.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 48

# FlexSizeWidget2D

**Name:** FlexSizeWidget2D

**Description:** DEPRECATED! Don't use this widget! Use SizeWidget2D instead.

**Category:** Deprecated

### 48.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Name	False	CharArray	The name of the widget instance	

NestedLevel	True	UInt	The ancestor's level, relative to the associated node, that will have the Size property modified.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Node	False	Node2D	The associated node of the widget.	
Size	True	Vector2	The value to set for the Size property of the ancestor node.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 48.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>



## Chapter 49

# FlexSwitchWidget2D

**Name:** FlexSwitchWidget2D

**Description:** DEPRECATED! Don't use this widget! Use SwitchWidget2D instead.

**Category:** Deprecated

### 49.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
BitmaskEnabled	True	Bool	If set to true, the Index will be considered as bitmask and each bit will toggle the visibility of a node. Bit 0 from Index will control the first child node, bit 1 the second child node, etc. Up to 32 nodes can be controlled this way.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Index	True	UInt	Index of the associated node's child that will be visible.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 49.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 50

# FlexToggleWidget2D

**Name:** FlexToggleWidget2D

**Description:** DEPRECATED! Don't use this widget! Use ToggleWidget2D instead.

**Category:** Deprecated

### 50.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Enabled	True	Bool	DEPRECATED. Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. The same node property is configured also by the widget property Visible which is available in all widgets. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
UseDeprecatedEnabled	False	Bool	Enables the usage of the deprecated property Enabled. The default value of this property is true for legacy reasons (notice that also default VisibleEnabled is true for the same reasons). It is recommended to set this property to false and to control the node visibility by using the Visible property which is available in all widgets.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget is deprecated and should be replaced by the same name widget that without Flex
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 50.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 51

# FocusGroupWidget2D

**Name:** FocusGroupWidget2D

**Description:** Candra Widget 2D Base Class

**Category:** Common

### 51.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ConfigureFocusGroup	True	Bool	Focus group is configured explicitly or uses default configuration.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DefaultFocusOrder	True	Short	Focus order of the element which should become focused when this group becomes active.	Ignored: The TTFis simulation for the gesture action is not ready yet.

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FocusAnchorId	True	UInt		Ignored: Not available in CGI configuration, not mentioned in user guide
FocusAnchorMode	False	Enum		
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusLayer	True	Short	Specifies to which focus layer this group belongs. There should be distinct layers for main surface screens, subspeller, popups to prevent focusing of elements in the main screen when a popup is displayed. Only elements in the groups with the highest layer will be focused.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusMarkerNode	False	Node2D	Focus marker which can be used by a focus animation manager to animate the changing of the focus inside this group (and its descendant groups). The default focus animation manager will create a translation animation which will move the marker from the old focus position to the new focus position. Please refer to the focus group section inside the widget user guide and also to the widget focus management concept document for more details.	
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	

FocusWrapAround	True	Bool	Focus wraps around to first/last element.	Ignored: The TTF is simulation for the gesture action is not ready yet.
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTF is simulation for the gesture action is not ready yet.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PreserveFocus	True	Bool	Current focused element is preserved when group becomes inactive.	Ignored: The TTF is simulation for the gesture action is not ready yet.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 51.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
------	--------------	-------------	---------	--------------	------------

## Chapter 52

# GestureWidget2D

**Name:** GestureWidget2D

**Description:** Posts gesture messages. This is a delegate widget and its behavior is controlled by a widget controller attached using the property ControllerId.

**Category:** Gesture

### 52.1 Property list

Name	Bindable	Type	Description	Test Scope
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: Not tested because this property is inherited and not specific to this Widget.



DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet. <a href="#">TC_W69_01</a>
Drag	True	Bool	Enable drag gesture detection	<a href="#">TC_W69_02</a>
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	<a href="#">TC_W69_08</a>
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	

Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTF is simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	TC_W69_04
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and not specific to this Widget.
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	TC_W69_01
PressRepeat	True	Bool	Enable repeat gesture detection	TC_W69_01
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	TC_W69_03
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	TC_W69_01

TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W69_02
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W69_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 52.2 Message list

Name	Subscribers	Description	Members	Distribution	Test Scope
DragGestureMsg	Controller, Model		Event: enGestureEvent::Enum.  Coordinate1X: Courier::Int.  Coordinate1Y: Courier::Int.  Velocity1X: Courier::Int.  Velocity1Y: Courier::Int. +	sequential	TC_W69_02

GestureMsg			Timestamp: Courier::Int. +	sequential	Ignored: Can't be manipulated for automation testing.
PinchSpreadGestureMsg	Controller, Model		Event: enGestureEvent::Enum.  Coordinate1X: Courier::Int.  Coordinate1Y: Courier::Int.  Coordinate2X: Courier::Int.  Coordinate2Y: Courier::Int. +	sequential	Ignored: Can't be manipulated for automation testing.
RotateGestureMsg	Controller, Model		Event: enGestureEvent::Enum.  Angle: Courier::Int16.  Coordinate1X: Courier::Int.  Coordinate1Y: Courier::Int.  Coordinate2X: Courier::Int.  Coordinate2Y: Courier::Int. +	sequential	Ignored: Can't be manipulated for automation testing.
SwipeGestureMsg	Controller, Model		Velocity1X: Courier::Int.  Velocity1Y: Courier::Int.  Velocity2X: Courier::Int.  Velocity2Y: Courier::Int.  Velocity3X: Courier::Int.  Velocity3Y: Courier::Int. +	sequential	TC_W69_03

TapGestureMsg	Controller, Model	Event: en- TapEvent::Enum.  Coordinate1X: Courier::Int.  Coordinate1Y: Courier::Int. +	sequential	IC_W69_01
---------------	----------------------	-------------------------------------------------------------------------------------------------------------	------------	-----------

## Chapter 53

# GizmoWidget2D

**Name:** GizmoWidget2D

**Description:** Provides support to move, resize and rotate nodes. This is a delegate widget and its behavior is controlled by a widget controller attached using the property ControllerId. The default gizmo controller uses the gizmo nodes from the property Nodes which are configured with the property GizmoEditModes (these 2 array properties need to have the same number of entries because there is a one to one mapping between gizmo nodes and gizmo edit modes). Two colors are used by the default gizmo controller: 1st color is set on gizmo nodes when no node is touched, 2nd color is used when at least one node is touched. Rotation is not supported together with move and resize by the default gizmo controller.

**Category:** Gesture

### 53.1 Property list

Name	Bindable	Type	Description	Test Scope
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development.
ColorIndex	True	UInt	Index of the color to be used if the gizmo is not pressed.	Ignored: Not tested because this property is inherited and not specific to this Widget.
Colors	True	Color	Colors used for different states (normal, pressed, warning, etc).	Ignored: Not suggested to be changed during run time.

ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time.
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: Not tested because this property is inherited and not specific to this Widget.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	TC_W70_02
DragDirection	False	Enum	Direction in which drag should be detected	TC_W70_02
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W70_08
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTF is simulation for the gesture action is not ready yet.

FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	TC_W70_04
GizmoEditModes	True	Byte	Edit modes used for the gizmo nodes. The enum array properties are not working due to a bug in SceneComposer so we have to use integers. 0 - None 1 - DragTranslateX 3 - DragTranslateY 5 - DragResizeRight 6 - DragResizeLeft 7 - DragResizeBottom 8 - DragResizeTop 9 - DragRotateClockwise 10 - DragRotateAntiClockwise 11 - PinchSpreadX 12 - PinchSpreadY	Ignored: Registered at start-up, should not be changed during run time.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget.



KeepAspectRatio	True	Bool	Preserves the aspect ratio while resizing using PinchSpread gestures. The aspect ratio is calculated from the Size property when the touch session starts.	Ignored: require using PinchSpread which not supported on TTFis.
LimitArea	True	Rectangle	Limits the movement and the size of the node.	Ignored: The application use them to initialize the widget, not suggested to be changed during runtime.
MaximumSize	True	Vector2	Maximum size of the node.	Ignored: The application use them to initialize the widget, not suggested to be changed during runtime.
MinimumSize	True	Vector2	Minimum size of the node.	Ignored: The application use them to initialize the widget, not suggested to be changed during runtime.
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Nodes	True	Node2D	Each node should have an edit mode associated by property GizmoEditMode. See the widget tooltip.	Ignored: Registered at start-up, should not be changed during run time.
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.

Position	True	Vector2	Position of the node.	Ignored: The application use them to initialize the widget, not suggested to be changed during runtime.
PressHold	True	Bool	Enable hold gesture detection	TC_W70_03
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotation	True	Float	Rotation of the node. In order to have the rotation against the center of the node the PivotOffset must be set accordingly.	Ignored: Not tested because this property is inherited and not specific to this Widget.
Size	True	Vector2	Size of the node. The node must have StretchBehaviour set to Fill.	Ignored: The application use them to initialize the widget, not suggested to be changed during runtime.

Swipe	True	Bool	Enable swipe gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W70_01
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W70_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: Not tested because this property is inherited and not specific to this Widget.

## 53.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
GizmoUpdMsg	Model		Position: Candra::Vector2.  Size: Candra::Vector2.  Rotation: Courier::Float.  Completed: bool.  Event: enGestureEvent::Enum. +	sequential	C_W70_11

## Chapter 54

# GridAutoArrangeWidget2D

**Name:** GridAutoArrangeWidget2D

**Description:** DEPRECATED! Don't use this widget! Instead use the GridAutoArangement property of the group. Auto arranges the children of a group with Grid layouter

**Category:** Deprecated

### 54.1 Property list

Name	Bindable	Type	Description	Test Scope
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
----------------	-------	------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

54.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 55

# HandWriting\_Util\_Widget2D

**Name:** HandWriting\_Util\_Widget2D

**Description:** HandWritingRecognition widget

**Category:** Input

### 55.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AccessMode	True	Enum	DynamicImgReadWrite status	Ignored: Andrei confirm that this feature was not completely implemented.
AddLineListWidget	False	Widget	Input the LineList3D widget	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
CharacterCompletionTime	False	UInt	CharacterCompletionTime : Courier::UInt32	TC_W41_01
CharactersToPredictedWord	True	custom://String	Stores the characters entered in edit field by user, which will be used to predicted words, IsBindable = true, Type=FeatStd::String	TC_W41_02

ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: Andrei confirmed this property only works for Button and Slider. It has nothing to do with touching them.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: HandWritingWidget2D does not support to test this property. No reaction on touching in the drawing area when Drag is enabled.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.



Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W41_08
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: HandWritingWidget2D does not support to test gesture simulations.
IncludeGestureRecognition	False	Bool	Include Range to recognize gestures	
IncludeLatinLowerCaseLetters	False	Bool	Recognize Latin LowerCase Letters	
IncludeLatinUpperCaseLetters	False	Bool	Recognize Latin UpperCase Letters	
IncludeNumbers	False	Bool	Recognize Numbers also	
IncludeSymbols	False	Bool	Recognize symbols also	

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property is dependent on EnableGroupWidget2D, so it will be tested in EnableGroupWidget2D section.
IntelligentPredictionMode	False	Enum	Set the IntelligentPrediction mode for recognizing the characters and words	
Language	False	Enum		
LineColor	False	Color	LineColor : Candra::Color	
LineWidth	False	Float	LineWidth : Courier::Float	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PredictedWordsSourcePointer	True	HWR Widget back end interface	Shared pointer instance of the HDRBackEndInterface	Ignored: Andrei confirm that this feature was not completely implemented.
PressHold	True	Bool	Enable hold gesture detection	Ignored: HandWritingWidget2D does not support to test this property. No reaction on touching in the drawing area when PressHold is enabled.

PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: HandWritingWidget2D does not support to test this property. No reaction on touching in the drawing area when PressRepeat is enabled.
PriorityMode	False	Enum	Set the Priority mode for recognizing the characters	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	Ignored: HandWritingWidget2D does not support to test this property. No reaction on touching in the drawing area when Swipe is enabled.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W41_07

UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W41_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 55.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
HWR_CandidateUpdateModel	Model	Update the list of predicted words.	Candidate_Char: Courier::Char*" pointer="true. +		TC_W41_02
HWR_TextWidgetUpdate	Widget	Update the recognized character shown on drawing area.	HWR_TextData: Can-dera::String. +		TC_W41_02

## Chapter 56

# ImageEffectWidget2D

**Name:** ImageEffectWidget2D

**Description:** Controls the bitmap property of the existing bitmap brush of a render node.

**Category:** Image

### 56.1 Property list

Name	Bindable	Type	Description	Test Scope
Bitmap	True	Image2D	Bitmap to be set on the BitmapBrushEffect inside the RenderNode.	TC_W12_01
Color	True	Color	Color to be set on the effect of the RenderNode.	TC_W12_02
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W12_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

56.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 57

# ImageSwitchWidget2D

**Name:** ImageSwitchWidget2D

**Description:** Controls the bitmap property of the existing bitmap brush of a render node based on an index.

**Category:** Image

### 57.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Bitmap0	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap1	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap10	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap11	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap12	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap13	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap14	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap15	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap16	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	

Bitmap17	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap18	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap19	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap2	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap20	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap21	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap22	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap23	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap24	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap25	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap26	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap27	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap28	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap29	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap3	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap30	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap31	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap32	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	



Bitmap33	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap34	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap35	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap36	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap37	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap38	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap39	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap4	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap40	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap41	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap42	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap43	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap44	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap45	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap46	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap47	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap48	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap49	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap5	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	

Bitmap50	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap51	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap52	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap53	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap54	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap55	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap56	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap57	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap58	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap59	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap6	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap60	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap61	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap62	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap63	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap64	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap65	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap66	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	

Bitmap67	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap68	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap69	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap7	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap70	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap71	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap72	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap73	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap74	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap75	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap76	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap77	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap78	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap79	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap8	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap80	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap81	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap82	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap83	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	

Bitmap84	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap85	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap86	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap87	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap88	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap89	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap9	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap90	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap91	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap92	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap93	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap94	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap95	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap96	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap97	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap98	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	
Bitmap99	False	Image2D	Bitmap to be set on the existing BitmapBrushEffect of the associated RenderNode.	

Count	True	UInt	Number of bitmaps.	Ignored: Using <i>Count</i> to validate the <i>Index</i> may cause problems during initialization when <i>Count</i> is set after the <i>Index</i> .
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If <i>InheritEnabled</i> is true then this widget is considered effectively enabled only if both local <i>Enabled</i> and the value inherited from the ancestor <i>EnableGroupWidget2D</i> are true.	Ignored: This derived property is not used in <i>ImageSwitch</i> widget.
Index	True	UInt	Index of the selected bitmap.	TC_W11_01
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local <i>Enabled</i> and the value inherited from the ancestor <i>EnableGroupWidget2D</i> are true. If this widget has no <i>EnableGroupWidget2D</i> ancestor or <i>InheritEnabled</i> is false then only the local <i>Enabled</i> is used.	Ignored: Not tested because this property is inherited and will be tested with <i>Enabled-GroupWidget</i>
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property <i>EnableRendering</i> which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W11_09
VisibleEnabled	False	Bool	Enables the configuration of the node property <i>EnableRendering</i> which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 57.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
------	--------------	-------------	---------	--------------	------------

## Chapter 58

# LabelWidget2D

**Name:** LabelWidget2D

**Description:** UNDER DEVELOPMENT! Don't use this widget yet! Label widget capable for rendering text in 2D scenes, single line and multiline.

**Category:** Under construction

### 58.1 Property list

Name	Bindable	Type	Description	Test Scope
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Text	True	custom://String	String text to be displayed on the label	
TruncationText	False	custom://String	Truncation text to be rendered in UTF-8 encoding	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

58.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 59

# LineListWidget3D

**Name:** LineListWidget3D

**Description:** A Widget for Drawing a list of lines.

**Category:** Polygon

### 59.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Name	False	CharArray	The name of the widget instance	
Node	False	Node3D	The associated node of the widget.	

### 59.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>



## Chapter 60

# ListBindingWidget2D

**Name:** ListBindingWidget2D

**Description:** Implements the list binding. See the list widget documentation.

**Category:** List

### 60.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
EnabledBindingIndex	False	Int	Binds the Enabled property of a widget to an integer value.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
ItemsBindingIndex	False	Int	Binds the property List.ListId to an integer value. It can also be used to fill a sublist with a vector data. See the list widget user guide.	
Name	False	CharArray	The name of the widget instance	
NameBindingIndex	False	Int	Binds the Name property of a widget to a text value.	
Node	False	Node2D	The associated node of the widget.	

SelectedBindingIndex	False	Int	Binds a widget property to an integer value. The following widget properties can be bound: Button.IsActive, ButtonGroup.ActiveIndex, Collapse.Collapsed, ColorSwitch.Index, ImageSwitch.Index, List.StartIndex, Switch.Index, TextColor.Active and Toggle.Enabled.	
TextBindingIndex	False	Int	Binds the Text property of a widget to a text value. The following widgets support it: Text, Label, ScrollableText and TextArea.	
UserDataBindingIndex	False	Int	Binds the UserData to an integer value.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 60.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 61

# ListFixedPageMovementGapWidget2D

**Name:** ListFixedPageMovementGapWidget2D

**Description:** Inserts a gap inbetween pages during fixed page movement of teh list.

**Category:** List

### 61.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
GapSize	True	Int	The gap size in pizel for the white space that will be created between pages during list movement.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

61.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 62

# ListItemAnimationMarkerWidget2D

**Name:** ListItemAnimationMarkerWidget2D

**Description:** Controls the size property of a node.

**Category:** List

### 62.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Animation	False	Animation	The animation to mark with a certain type.	
AnimationType	True	Enum	The type to mark the animaton.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

62.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 63

# ListMovementAnimationsProviderWidget2D

**Name:** ListMovementAnimationsProviderWidget2D

**Description:** Controls the size property of a node.

**Category:** List

### 63.1 Property list

Name	Bindable	Type	Description	Test Scope
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
GoToDuration	True	UInt	The duration of the go to movement	
GoToExponent	True	Float	Exponent value for the exponential out go to movement type.	
GoToType	True	Enum	The type of animation that is used when the list is scrolled directly to a target item.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
ItemByItemDuration	True	UInt	The duration of the item by item movement	
ItemByItemExponent	True	Float	Exponent value for the exponential out item by item movement type.	
ItemByItemType	True	Enum	The type of animation that is used when the list is scrolled item by item.	

Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PageByPageDuration	True	UInt	The duration of the page by page movement	
PageByPageExponent	True	Float	Exponent value for the exponential out page by page movement type.	
PageByPageType	True	Enum	The type of animation that is used when the list is scrolled page by page.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 63.2 Message list

<i>Name</i>	<i>Subscribable</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------



## Chapter 64

# ListOverscrollWidget2D

**Name:** ListOverscrollWidget2D

**Description:** ListOverscrollWidget2D widget

**Category:** List

### 64.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
DragAttenuationFactor	True	Float	It influences the Drag behavior when overscrolling. Valid values are between 0.0 and 1.0. If the value is 0.0f then there is no resistance, the list will follow the finger pixel by pixel. If the value is 1.0f then the drag will be fully attenuated, the list will not move at all.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	

MaxRelaxingSpeed	True	Float	Represents the initial speed for the relaxing movement if the overscroll distance would be equal to the viewport size. The speed is expressed in viewport ratio / seconds; for example, a speed equal to 0.5 means half of the viewport would be relaxed in one second.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OnDrag	True	Bool	Enables or disables overscroll for drag	
OnSwipe	True	Bool	Enables or disables overscroll for swipe	
SwipeMaxDistanceFactor	True	Float	It influences the Swipe behavior when overscrolling. Valid values are between 0.0 and 1.0. It represents the ratio of the viewport size that could be reached when overscrolling by swipe. If the drag overscroll is already greater than this distance, no additional overscrolling by swipe is performed.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 64.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 65

# ListScrollAnchorWidget2D

**Name:** ListScrollAnchorWidget2D

**Description:** ListScrollAnchorWidget2D widget

**Category:** List

### 65.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AnchorData	False	Enum	Whether the anchor position refers to a row or a column	
AnchorPosition	False	Enum	Position of the anchor, either First, Center or Last	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

65.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 66

# ListSpeedAnimationWidget2D

**Name:** ListSpeedAnimationWidget2D

**Description:** ListSpeedAnimationWidget2D widget

**Category:** List

### 66.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AccelerationTransitionTimeFraction	False	Float	The fraction of the page by page animation time that the transition to the acceleration state will take	
BehindItemFactor	False	Float	Scroll speed factor that will be applied for items behind the reference item/column - will be multiplied with the delta to the touched item/column.	
DecelerationTransitionTimeFraction	False	Float	The fraction of the page by page animation time that the transition to the deceleration state will take	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	

MaxSpeed	False	Int	The maximum speed that will be considered. Speeds exceeding this value will be limited to it.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
ReferenceItemFactor	False	Float	Scroll speed factor that will be applied for the reference item/column.	
TouchAsReference	False	Bool	Whether the touched item is considered as reference or not while applying the below factors. If this is set to false, the leading item in the direction of the movement will be considered the reference item.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 66.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 67

# ListWidget2D

**Name:** ListWidget2D

**Description:** Provides support for creating lists of elements. See the list widget documentation.

**Category:** List

### 67.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AcceptImmediatePositioning	True	Bool	If set to true, immediate positioning is taken into consideration. Please check the other properties to enable immediate positioning for different events.	
AcceptImmediatePositioningOnFirstAppearance	True	Bool	If set to true, the first ListChangeMsg with ListChangeSet that is received after rendering is enabled for the view and if immediate positioning is activated, then the position is set to the received value without using an animation; further position setting uses an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AcceptImmediatePositioningOnItemScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangeDown or ListChangeUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	

AcceptImmediatePositioningOnPageScroll	True	Bool	If set to true, if a ListChangeMsg with ListChangePageDown or ListChangePageUp is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AcceptImmediatePositioningOnPositionSet	True	Bool	If set to true, if a ListChangeMsg with ListChangeSet is received and immediate positioning is activated, then the position is set to the received value without using an animation. If set to false, then the immediate positioning is ignored and the animation is always used to scroll to the given position.	
AnimationOffset	False	Short	Offset from first visible item to the position of the first animated item, give a negative value for the offset to the first invisible item of the list	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
AutoViewportSize	False	Bool	If enabled, the size of the viewport will be calculated as NumberOfItems multiplied with the size of the InvalidItemTemplate. This is done for the direction of scrolling; the other direction will remain unmodified.	
BufferSize	True	UInt	Used for windowed data. The buffer size of the elements requested prior and after the visible ones.	TC_W26_10
CachedLayout	False	Bool	Caches the layout of the list entries, if set to true the layout calculation is done only once for each list item and then cached.	
ConfigureFocusGroup	True	Bool	Focus group is configured explicitly or uses default configuration.	Ignored: The TTF is simulation for the gesture action is not ready yet.
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time
Coverflow	True	Bool	True if the list is a coverflow.	



CustomAnimationsGroupNode	False	Node2D	Root node of animated nodes that are placeholders for the real items in the list. The animated properties will be applied to the nodes that will be added by the list to the ItemsNode.	
CustomListAnimations	False	Animation	Array of animations for all visible list elements.	
DefaultFocusOrder	True	Short	Focus order of the element which should become focused when this group becomes active.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: Not tested because this property is inherited and not specific to this Widget.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	TC_W26_04
DragAndSwipeOnNeed	True	Bool	If true then drag and swipe will be dynamically disabled when a new touch session starts if all items are visible.	TC_W26_11
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DynamicGrid	False	Bool	Whether or not the grid is automatically adapting to the text size.	TC_W28_01
DynamicGridEmptyCellsPolicy	False	Enum	The policy to distribute the empty spaces in the last row of the list	TC_W28_01

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W26_08
ExpandAutoScroll	True	Bool	Whether or not expand animations scroll the list so that the expanded item fully fits inside the viewport.	
FixedPageScrolling	True	Bool	Works only if all the data is available (no windowed data). Should be used only with small lists. If set to true, the pages always start at a fixed position, similar to a book. If set to false, the pages are always considered relative to the current start index.	TC_W26_07
FixedPageScrollingOffset	True	Float	Percentage of the viewport needed to be scrolled/swiped with the touch in order to go to the next/previous page.	TC_W28_04
FocusAnchorId	True	UInt		Ignored: Not available in CGI configuration, not mentioned in user guide
FocusAnchorMode	False	Enum		
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTF is simulation for the gesture action is not ready yet.
FocusLayer	True	Short	Specifies to which focus layer this group belongs. There should be distinct layers for main surface screens, subspeller, popups to prevent focusing of elements in the main screen when a popup is displayed. Only elements in the groups with the highest layer will be focused.	Ignored: The TTF is simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTF is simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	

FocusWrapAround	True	Bool	Focus wraps around to first/last element.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedIndex	True	Int	The index of the currently focused item.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusedNode	False	Node2D	The node that will highlight the focused node.	
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	TC_W28_02
IgnoreListChangeMsg	False	Enum	Defines when the ListChangeMsg will be ignored during the list scrolling or swiping	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
InvalidItemTemplate	False	Node2D	Template used for not yet available data or if no other template is found for some data.	
ItemsNode	False	Node2D	The list items will be added as children of this node.	
LimitExceededDownAnimation	False	Animation	Animation that is played when the user wants to exceed the list on bottom. It is only played if LimitExceededDownAnimationType is CustomAnimation or circular scrolling is not enabled.	

LimitExceededDownAnimationTime	True	UInt	Defines the how much time (ms) the exceeded down animation takes.	
LimitExceededDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on bottom of the list.	Ignored: Not tested because focus states are not supported in the test app.
LimitExceededDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitExceededUpAnimation	False	Animation	Animation that is played when the user wants to exceed the list on top. It is only played if LimitExceededUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitExceededUpAnimationTime	True	UInt	Defines the how much time (ms) the exceeded up animation takes.	
LimitExceededUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is exceeded on top of the list.	Ignored: Not tested because focus states are not supported in the test app.
LimitExceededUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
LimitReachedDownAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedDownAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedDownAnimationTime	True	UInt	Defines the how much time (ms) the reached down animation takes.	TC_W29_07
LimitReachedDownAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on bottom of the list.	TC_W29_06
LimitReachedDownBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.

LimitReachedUpAnimation	False	Animation	Animation that is played when the user reaches the list on top during swipe, scroll or set. It is only played if LimitReachedUpAnimationType is CustomAnimation or circular scrolling is not enabled.	
LimitReachedUpAnimationTime	True	UInt	Defines the how much time (ms) the reached up animation takes.	TC_W29_05
LimitReachedUpAnimationType	True	Enum	Sets the kind of animation that will be played if the lists limit is reached during swipe, scroll or set on top of the list.	TC_W29_04
LimitReachedUpBounceAmplitude	True	Vector2	Amplitude vector used for generating the margin bounce animation.	Ignored: It is related to animation, cannot test by automation.
ListAlignment	False	Enum	Only applicable for lists with less then the maximum number of visible items. If set to 'End' the items will be aligned at the end of the list.	
ListId	True	UInt	A unique identifier of the list that will be used to request data to the model and react to data sent from the model.	TC_W26_01
LockOutFixMovementOffset	False	Float	Offset in percent of the viewport required for LockOutType LockOutFixMovement.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NumberOfItems	True	UInt	The number of items is only available for itemwise lists(PixelWiseScrollingEnabled set to false). For stack layouted lists it has the meaning of the number of elements that would be visible. For grid layouted lists it has the meaning of the number of columns(for horizontal scrolling) or rows(for vertical scrolling) that will be seen. The number of rows(for horizontal scrolling) or columns(for vertical scrolling) will be taken from the GridLayouter.	TC_W26_06
PageIndicationAccuracy	False	Float	Accuracy of the current page position indication.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PixelWiseAnimationTime	True	UInt	Time in ms to finish the pixel wise animation	

PixelWiseScrollingEnabled	True	Bool	If set to true, scrolling is done pixel by pixel, meaning that an item can be partially visible. If set to false, scrolling is done item by item; items are always completely visible.	TC_W28_03
PostListChanged	False	Bool	If true, the list will post a ListChangedUpdMsg message everytime the first visible index changes.	
PreparedItemsBufferSize	False	Byte	Number of additional items being prepared when the list is idle	
PreparedItemsUpdateTriggerOffset	False	Byte	Offset to the edge of the PreparedItemsBuffer - the prepared item buffer will be updated and old items discarded as soon as this offset is reached	
PreserveFocus	True	Bool	Current focused element is preserved when group becomes inactive.	Ignored: The TTF is simulation for the gesture action is not ready yet.
PreserveScrollIndex	False	Bool	If enabled, and the list is cached, the widget will remember the last scrolled position and set the same on re-entering the List View, unless set to a different value by the model.	
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollAnimationInput	False	Enum	Input of scroll animation - Either pixel position or index of list items	
ScrollbarAlwaysVisible	True	Bool	If set to true, scrollbar is always visible, regardless if it's needed or not. If set to false, it's only visible if needed.	TC_W30_02
ScrollingOrientation	True	Enum	Scrolling orientation: vertical or horizontal	TC_W26_03
ScrollingType	False	Enum	Sets the way this list scrolls: Default beginning to end, continuous or last page not filled.	
ShortPixelWiseAnimationTime	True	UInt	Short time in ms to finish the pixel wise animation; used when setting the position with a ListChangeSet request type	

ShowItemsOnViewActivated	False	Bool	Whether items are visible or not on view shown, before receiving a ListCustomAnimationReqMsg.	
Snap	False	Enum	Specifies how the list should behave after scrolling. SnapToEdge - List will snap to the nearest edge, SnapToCenter - List will snap to the center	
SnapHoldTime	False	UInt	Timeout in milliseconds, if reached no snapping will be done	
SnapOffsetThreshold	False	UInt	Threshold of offsetDelta = Abs(offsetTop - offsetBottom) in pixels - if the offset delta is smaller than the threshold the snap will be done to the edge closer to the touch position	
StartIndex	True	Int	The index of the first visible item. Negative indices are accepted in case of circular scrolling.	TC_W26_02
Swipe	True	Bool	Enable swipe gesture detection	TC_W26_05
SwipeDirection	False	Enum	Direction in which swipe should be detected	
SwipingAcceleration	True	Float	Acceleration used to slow down swiping	Ignored: Not tested because this property is having a reported problem but the fix is not available. Kindly see in RTC_959172
SwipingDistanceThreshold	True	UInt	Deprecated. Please do not use this, as it will be removed in a future version. Minimum distance in display units needed to drag in order to begin scrolling	Ignored: Deprecated
SwipingMaxDistance	True	Float	Maximum distance that can be covered within one swipe	Ignored: Not tested because this property is having a reported problem but the fix is not available. Kindly see in RTC_959172

SwipingTimerThreshold	True	UInt	Minimum time in ms needed to press in order to focus an item	Ignored: Not tested because this property is having a reported problem but the fix is not available. Kindly see in <a href="#">RTC_959172</a>
SwipingVelocityThreshold	True	Float	Deprecated. Please do not use this, as it will be removed in a future version. Minimum velocity in display units/ms needed to move with in order to begin swiping	Ignored: Deprecated
Tap	True	Bool	Enable press and tap gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
TemplateGroup	False	Node2D	The parent of template nodes. Template nodes will be cloned based on the item needed to be displayed and the clone will be added to the ItemsNode.	
TemplateScrollAnimations	False	Animation	Array of template animations for list scrolling.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	<a href="#">TC_W26_08</a>
UpdateTriggerOffset	True	UInt	Used for windowed data. The offset relative to the first or last visible item index, respectively used to request new data when reached.	<a href="#">TC_W26_10</a>
UsableViewportPadding	False	Margin	Viewport offset for visible items in the list - useful in case the list is overlapped by some other content.	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.



Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W26_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

67.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

ListAnimatedChangeReqMsg		<p>ListAnimatedChangeReqMsg message is sent from the DM to the ListWidget to request animated movement for the list as well as triggering an external synchronized animation. This message can be sent from an external component as a reaction to up/down hardware button presses or to software buttons from a scrollbar for example.</p> <p>Properties:</p> <p>ListId: the id of the targeted list.</p> <p>ListChangeType: the type of the requested movement:</p> <p>ListChangeUp – moves the list backwards by Value items</p> <p>ListChangeDown – moves the list forward by Value items</p> <p>ListChangePageUp – moves the list backward by Value pages</p> <p>ListChangePageDown – moves the list forward by Value pages</p> <p>ListChangeSet – sets the position of the list to Value</p> <p>Value: the amount of the move.</p> <p>MovementDetails</p> <p>ExternalAnimationAction: see AnimationReqMsg</p> <p>ViewId: see AnimationReqMsg</p> <p>CompositePath: see AnimationReqMsg</p> <p>AnimationId: see AnimationReqMsg</p> <p>AnimationProperties: see AnimationReqMsg</p> <p>ListChangeMsgSource: the source of the message</p> <p>ImmediatePositioning: use no list animation for going to the required position</p>	<p>ListId: ::Courier::UInt32.</p> <p>ListChangeType: ListChangeType.</p> <p>Value: ::Courier::Int32.</p> <p>MovementDetails: ListMovement-Details.</p> <p>ExternalAnimationAction: Courier::AnimationAction::Enum" default- Value="Courier::AnimationAction::Start.</p> <p>ViewId: Courier::ViewId" default- Value="Courier::ViewId().</p> <p>CompositePath: Courier::CompositePath" default- Value="Courier::CompositePath().</p> <p>AnimationId: Courier::ItemId" default- Value="Courier::ItemId().</p> <p>AnimationProperties: Courier::AnimationProperties" default- Value="Courier::AnimationProperties().</p> <p>ListChangeMsgSource: ListChangeMsgSource- gSourceType" default- Value="ListChangeMsgSourceUnknown.</p> <p>ImmediatePositioning: bool" default- Value="false. +</p>	TC_W47_04
ListCacheReqMsg	View	<p>ListCacheReqMsg request is treated by the list widget to explicitly cache the visual item at the given index, after cloning, until explicitly requested to delete from cache</p>	<p>ListId: Courier::UInt32.</p> <p>Index: Courier::UInt32.</p> <p>Cache: bool. +</p>	TC_W47_01

ListChangeMsg	View	<p>ListChangeMsg message is sent from the DM to the ListWidget to request movement for the list. This message can be sent from an external component as a reaction to up/down hardware button presses or to software buttons from a scrollbar for example.</p> <p>Properties:</p> <p>ListId: the id of the targeted list.</p> <p>Value: the amount of the move.</p> <p>ListChangeType: the type of the requested movement:</p> <p>ListChangeUp – moves the list backwards by Value items</p> <p>ListChangeDown – moves the list forward by Value items</p> <p>ListChangePageUp – moves the list backward by Value pages</p> <p>ListChangePageDown – moves the list forward by Value pages</p> <p>ListChangeSet – sets the position of the list to Value</p>	<p>ListId: ::Courier::UInt32.</p> <p>ListChangeType: ListChangeType.</p> <p>Value: ::Courier::Int32.</p> <p>ListChangeMsgSource: ListChangeMsgSource "default-Value="ListChangeMsgSourceUnknown.</p> <p>ImmediatePositioning: bool" default-Value="false. +</p>	TC_W47_02
ListChangedUpdateModel	Model	<p>Posted by the ListWidget if its PostListChanged property is true and the first visible index has changed. This message is posted to the model to inform other components that the displayed content of the list has changed., e.g FirstVisibleIndex</p> <p>Properties:</p> <p>listId: the id of the list that sends the message.</p> <p>MovementStatus: how is the list moving (finished moving, item, page, set, scrolling, swiping)</p> <p>Circular: true if the list is circular scrollable; false otherwise</p> <p>StartIndex:[deprecated] the new value of the first visible index</p> <p>FirstVisibleIndex: the new value of the first visible index</p> <p>FirstItemFullyVisible: true if the first item is fully visible; false otherwise. If this is true, it means that the FirstVisibleIndex represents also the start index of the list; if it's false, the start index should be FirstVisibleIndex+1</p> <p>MinIndex: the value of the minimum possible value for the first visible index</p> <p>MaxIndex: the value of the maximum possible value for the first visible index</p> <p>DataSequenceNumber: sequence number of the data used to prepare the visible items (see ListDataProvider.getSequenceNumber)</p>	<p>ListId: ::Courier::UInt32.</p> <p>MovementStatus: ListMovementStatusType.</p> <p>ListSource: ListChangeMsgSourceType.</p> <p>Circular: bool.</p> <p>StartIndex: ::Courier::UInt32.</p> <p>FirstVisibleIndex: ::Courier::UInt32.</p> <p>FirstItemFullyVisible: bool.</p> <p>MinIndex: ::Courier::UInt32.</p> <p>MaxIndex: ::Courier::UInt32.</p> <p>DataSequenceNumber: ::FeatStd::SizeType.</p> <p>PagePosition: ::Courier::Float. +</p>	TC_W47_03

ListContentUpdMsgModel	ListContentUpdMsg	ListContentUpdMsg is send by the ListWidget each time new data is received by either ListDataProviderResMsg or ListDataProviderUpdMsg DataSequenceNumber: sequence number of the data used to prepare the visible items (see ListDataProvider.getSequenceNumber)	ListId: Courier::UInt32.  FirstVisibleIndex: Courier::UInt32.  MaxFirstVisbleIndex: Courier::UInt32.  DataSequenceNumber: ::Feat-Std::SizeType. +	TC_W47_08
ListCustomAnimationControlMsg	ListCustomAnimationIndMsg	ListCustomAnimationIndMsg message is posted when the animation is finished.	ListId: ::Courier::UInt32. +	TC_W47_05
ListCustomAnimationViewReqMsg	ListCustomAnimationReqMsg	ListCustomAnimationReqMsg message is treated by the list widget in order to play a custom animation given by it's name.	ListId: ::Courier::UInt32.  Action: CustomListAnimationAction. +	TC_W47_05
ListCustomAnimationControlResMsg	ListCustomAnimationResqMsg	ListCustomAnimationResqMsg message is posted in response to ListCustomAnimationReqMsg. Success indicates if the processing was done or not	ListId: ::Courier::UInt32.  Action: CustomListAnimationAction.  Success: bool. +	TC_W47_05
ListDataProviderUpdMsg		Updates the entries of the previously sent ListDataProvider by replacing them. It does not insert or remove entries. Properties: UpdateInfo: contains the entries to be updated.	Updater: tSharedPtrDataProviderUpdater. +	TC_W47_07

ListDateProviderReqMsg	Model Controller	Request is sent by the ListWidget to request data from the model. The model should react to this message and, depending on the listId, it should prepare the data to send to the list. The model can request data to other components when this message is received or it can do it at a different time and cache the received values so that it will be able to fulfill the request. Properties: listId: the id of the list that sends the message. It will be used by the response message to identify the target list. startIndex: the index of the first item that is requested. windowElementSize: the number of requested elements. circular: whether the requesting list is circular or not* The index of the first requested element. Even in case of circular scrolling, this is positive, as the startIndex + windowElementSize can yield values greater than the list size, in which case the indices should be treated as % virtualListSize* The number of requested element. In case of circular scrolling, the startIndex + windowElementSize can yield values greater than the list size, in which case the indices should be treated as % virtualListSize	listId: ::Courier::UInt32.  startIndex: ::Courier::Int32.  windowElementSize: ::Courier::UInt32.  circular: bool.  cachedData: bool.  scrollAnchorId: Courier::UInt32" default-Value="MAX_UINT_VALUE. +	TC_W47_06
ListDateProviderResMsg		This message is sent as a response to a ListDateProviderReqMsg. It should provide the list with a valid set of data that it previously requested via a ListDateProviderReqMsg. The model posts this kind of message after it builds a DataProvider containing the relevant data the list requested. See onListDateProviderReq for a concrete example Properties: ListDateProvider: a pointer to an instance of DataProvider that will be used by the list to display the requested set of data.	ListDateProvider: tSharedPtrData- aProvider. +	TC_W47_06
ListFocusChangeReqMsg		This message is currently not used.	ListId: ::Courier::UInt32.  Steps: ::Courier::Int32. +	Ignored: This message is currently not used.
ListItemExpandEndUpdMsg	Model Controller	ListItemExpandEndUpdMsg message is issued by the list widget to signal the end of a certain expand animation	ListId: ::Courier::UInt32.  ItemIndex: ::Courier::UInt32.  Forward: bool. +	TC_W47_10

ListItemExpandReqMsg	Model	ListItemExpandReqMsg message is treated by the list widget in order to play an expand animation for the given item, in the requested direction.	ListId: ::Courier::UInt32.  ItemIndex: ::Courier::UInt32.  Forward: bool. +	TC_W47_10
ListMorphReqMsg	Control Model	ListMorphReqMsg request sent to control and data model to trigger the morphing of the specified list into a state identified by Hint.	ListId: Courier::UInt32.  Hint: Courier::UInt32. +	Ignored: This message has nothing to be handled inside the ListWidget, just a message for user's use.
ListMovementFinishedMsg	Model	Posted by the ListWidget when the movement triggered by touch gestures or a ListChangeMsg has finished Properties: listId: the id of the list that sends the message. FirstVisibleIndex: the value of the first visible index	ListId: ::Courier::UInt32.  FirstVisibleIndex: ::Courier::UInt32. +	TC_W47_11
ListStatusUpdMsg	Model View	Posted by the ListWidget normally to the model each time for example its swiping state has changed. This message should be treated by other components that need to react to changes in the swiping state of the list. These components can be either items contained inside the list or other various external components. Properties: Status: the new swiping state; possible states: ListIdle, ListTouched, ListFocused, ListScrolling, ListSwiping ListId: the id of the list that has changed its state View: the ViewId of the View containing the ListWidget instance List: the name of the ListWidget instance Direction: Prev, Next, Unavailable LockoutActive: on/off	Status: ListStatusType.  ListId: ::Courier::UInt32.  ChangeSource: ListChangeMsgSourceTypes.  ChangeType: ListGeneralChangeType.  Direction: ListStatusDirectionType.  LockoutActive: bool. +	TC_W47_12

ListVisibiltyUpdMsg	Model View	<p>ListVisibiltyUpdMsg posted by the ListWidget to the model each time when the List comes in foreground. This message should be treated by other components that need to react to changes, for example in the lockout component of the DM to lock the screen immediately after the list becomes visible.</p> <p>Properties:</p> <p>VisibilityStatus: the new visibility state; possible states: ListVisible, ListInVisible</p> <p>ListId: the id of the list that has changed its visibility</p> <p>View: the ViewId of the View containing the ListWidget instance</p> <p>List: the name of the ListWidget instance</p>	<p>VisibilityStatus: ListStatusVisi- bilityType.</p> <p>ListId: ::Courier::UInt32. +</p>		TC_W47_13
ListVisualContentUpdateMsg	Model	<p>ListVisualContentUpdateMsg is sent by the ListWidget each frame</p>	<p>ListId: Courier::UInt32.</p> <p>FirstVisibleIndex: Courier::UInt32.</p> <p>FirstCompleteVisibleIndex: Courier::UInt32.</p> <p>NumberOfVisibleItems: Courier::UInt32.</p> <p>NumberOfCompleteVisibleItems: Courier::UInt32.</p> <p>ConfiguredNumberOfItems: Courier::UInt32. +</p>		TC_W47_09

LockOutMsg	View	<p>LockoutMsg message has to be sent from the DM to the ListWidget to request for lockout (speed lockout feature)</p> <p>Properties:</p> <p>ReceiverId: the id of the list that should be locked; Zero (0) means broadcast to all lists</p> <p>LockOut: The LockOutType decides how the list reacts on user input</p> <ul style="list-style-type: none"> <li>- LockOutNone: deactivates any LockOut, normal operation of the list and scrollbar</li> <li>- LockOutPageWise: instead of pixelwise swipe on any user input pagewise scrolling is enabled</li> <li>- LockOutPageWisePrev: pagewise scrolling, but in prev direction locked (up/left)</li> <li>- LockOutPageWiseNext: pagewise scrolling, but in next direction locked (dn/right)</li> <li>- LockOutDirPrev: up/left operations are locked, down or right operations allowed</li> <li>- LockOutDirNext: up/left operations allowed, down or right operations are locked</li> <li>- LockOutComplete: everything is locked, list and scrollbar</li> <li>- LockOutFixMovement: pagewise scrolling is enabled but only after a specified percent of the viewport was scrolled, see also ListWidget property LockOutFixMovmentOffset</li> </ul> <p>in case of coverflow the entries will be scrolled itemwise.</p>	<p>ReceiverId: ::Courier::UInt32.</p> <p>LockOut: LockOutType. +</p>		TC_W47_14
SliderCurrentValueChangedMsg	Slider Controller	<p>SliderCurrentValueChangedMsg message is posted from ListWidget when the CurrentValue from SliderWidget property changes. External components should react to this message to possibly change their states based on the current value.</p> <p>Properties:</p> <p>View: the ViewId of the View containing the DropDownListWidget instance</p> <p>Sender: the name of the DropDownListWidget instance</p> <p>CurrentValue: the current value of the slider</p> <p>TouchState: the touch state of the slider</p>	<p>CurrentValue: ::Courier::Float.</p> <p>TouchState: SliderTouch-State. +</p>		Ignored: This message has nothing to be handled by the List-Widget, will be tested in the Slider-Widget.
SourceTypeLockOutMsg	View	<p>SourceTypeLockOutMsg message has to be sent from the DM to the ListWidget to request for lockout (speed lockout feature)</p> <p>Properties:</p> <p>ReceiverId: the id of the list that should be locked; Zero (0) means broadcast to all lists</p>	<p>ReceiverId: ::Courier::UInt32.</p> <p>LockOut: hmibase::widget::list::SourceTypeLockoutMap +</p>		



## Chapter 68

# MarginWidget2D

**Name:** MarginWidget2D

**Description:** Controls the margin properties of a node. To be able to animate different margins a property for each margin is provided

**Category:** Layout

### 68.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and not specific to this Widget.
MarginBottom	False	Float	Bottom margin of attached node	
MarginLeft	False	Float	Left margin of attached node.	
MarginRight	False	Float	Right margin of attached node	
MarginTop	False	Float	Top margin of attached node	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W74_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

68.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 69

# MarkerWidget2D

**Name:** MarkerWidget2D

**Description:** Marks a node in the scene to help other widgets or components with finding the node they need for a specific purpose. For example the camera to be used for a transition or the adorning to be used during dragging of a button. Use the static method MarkerWidget2D::findMarker to search for a marker widget. See the WidgetUserGuide for more details.

**Category:** Common

### 69.1 Property list

Name	Bindable	Type	Description	Test Scope
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
Id1	True	UInt	1st identifier	
Id2	True	UInt	2nd identifier	
Id3	True	UInt	3rd identifier	
Id4	True	UInt	4th identifier	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

69.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 70

# MeshWidget2D

**Name:** MeshWidget2D

**Description:** Performs a perspective projection of a 3D rotated effect (e.g. Bitmap or SolidColor effects) based on the configured origin, rotation and field of view.

**Category:** Common

### 70.1 Property list

Name	Bindable	Type	Description	Test Scope
CameraDistance	True	Float	Defines a normalized camera distance (the projection is based on a normalized geometry of width and height 1) with a default value of 1.	TC_W17_02
CustomHorizontalOrigin	True	Float	Defines a normalized (bitmap resolution independent) custom origin that can not be defined with Left, Center or Right.	TC_W17_03
CustomVerticalOrigin	True	Float	Defines a normalized (bitmap resolution independent) custom origin that can not be defined with Top, Center or Bottom.	TC_W17_04
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing.
FieldOfView	False	Float	The area on the Y axis view field in degrees. It influences the perspective distortion of the projection. Higher angles will result in more distorted projection. Lower angles will result in less perspective distorted projections. The angle has to be higher than 0 and less than 180.	

FitStrategy	False	Enum	Defines how the perspective projected mesh should fit into the original boundaries.	
FitToMaximumSizeRotation	False	Float	Defines the rotation angle in degree that will be uses as maximum for the FitToMaximumSize fit strategy. Default is an angle of 90 degree.	
HorizontalOrigin	False	Enum	The origin influences where the rotation is performed and how the perspective projection looks. It is defined as a normalized (bitmap resolution independent) origin on the y axis as Left (0), Center (0.5), Right (1.0) or Custom (see CustomHorizontalOrigin).	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property has not been supported for testing.
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OriginOffset	False	Vector2	In addition to the configured normalized origin this pixel based origin offset is added.	
Rotation	True	Float	Defines the rotation angle in degree around configured rotation axis at the configured origin.	TC_W17_01
RotationAxis	False	Vector3	Defines the axis (default is the y axis) for the rotation as a 3D direction vector (the vector will be internally normalized).	
VerticalOrigin	False	Enum	The origin influences where the rotation is performed and how the perspective projection looks. It is defined as a normalized (bitmap resolution independent) origin on the y axis as Top (0), Center (0.5), Bottom (1.0) or Custom (see CustomVerticalOrigin).	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W17_09

VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
----------------	-------	------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

70.2 Message list

<i>Name</i>	<i>Subscribes</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	-------------------	--------------------	----------------	---------------------	-------------------

## Chapter 71

# MeterWidget2D

**Name:** MeterWidget2D

**Description:** Widget which shows the progress in a circular direction

**Category:** Extra

### 71.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
CurrentValue	True	Float	Indicates the Ongoing operation received from external component	TC_W43_01
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W43_03
IndicatorNode	False	Node2D	Node used as a pointer to rotate to show progress	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
MaximumAngle	False	Float	Angle of the Indicator to represent the maximum value	
MaximumValue	False	Float	End Value of the ProgressBar. Progress ends at this value.	



MinimumAngle	False	Float	Angle of the Indicator to represent the minimum value	
MinimumValue	False	Float	Start Value of the ProgressBar. Progress starts from this value.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
RotationType	False	Enum	Decides the direction the progress should happen - clockwise or anticlockwise direction	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W43_02
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 71.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 72

# MultiSliderHelperWidget3D

**Name:** MultiSliderHelperWidget3D

**Description:** Implementation of MultiSliderHelperWidget3D

**Category:**

### 72.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
ColorBottom	False	Color	Defines the color if center of polygon area moves to bottom.	
ColorTopLeft	False	Color	Defines the color if center of polygon area moves to top left.	
ColorTopRight	False	Color	Defines the color if center of polygon area moves to top right.	
Enabled	False	Bool	Enabled: Enable or disable the widget	
Name	False	CharArray	The name of the widget instance	
Node	False	Node3D	The associated node of the widget.	
NonLinearFactor	False	Float	Factor for non-linear ratio between value and depth (0.001..10). 0.001 means linear, higher values define more exponential ratio.	
RadiusOffset	False	Float	Value > 0 creates a gap on the edges of the polygon.	
Scale	False	Bool	The texture will be cut off on top edge (false) or is scaled down (true).	
ValueMin	False	Float	To show some content even if the controller value is 0, a minimum value for the view can be defined.	

### 72.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 73

# MultiSliderWidget2D

**Name:** MultiSliderWidget2D

**Description:** Widget to change the importance of a property and visualize the same in the meaning of a unique profile representation.

**Category:** Range

### 73.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.

Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.

GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
HandleNode	False	Node2D	Pointer to the node that represents a slider handle. This node must not be a child node of the widget associated node and rendering of that should be disabled.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
MultiSliderView	False	Widget	The multi slider view widget for graphical representation.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RadiusCenter	False	Float	Radius for center button (sets all values to minimum)	
RadiusMax	False	Float	Radius for maximum values in pixel	
RadiusMin	False	Float	Radius for minimum values in pixel	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
StartAngle	False	Float	Defines the angle in degrees of the first value. 0° is 12 o'clock position.	

Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Values	True	Float	Array of float values to be controlled by the widget. The number of values also defines the number of billboard to be shown by MultiSliderHelperWidget3D.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 73.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 74

# NodeControlWidget2D

**Name:** NodeControlWidget2D

**Description:** Controls the properties of a node allowing to expose them outside composites or to change them using data binding sources.

**Category:** Common

### 74.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AlphaValue	True	Float	See Node's property AlphaValue	TC_W77_01
AlphaValueEnabled	False	Bool	Enables configuration of the property AlphaValue	
BoundingBox	True	Rectangle	See Node's property BoundingBox	TC_W77_02
BoundingBoxEnabled	False	Bool	Enables configuration of the property BoundingBox	
Collapsible	True	Bool	See Node's property Collapsible	Ignored: This prop is only used to improve the performance, has no visual effect.
CollapsibleEnabled	False	Bool	Enables configuration of the property Collapsible	
Column	True	Byte	See Node's property Column	TC_W77_04
ColumnEnabled	False	Bool	Enables configuration of the property Column	
ColumnSpan	True	Byte	See Node's property ColumnSpan	TC_W77_03
ColumnSpanEnabled	False	Bool	Enables configuration of the property ColumnSpan	

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This prop is inherited but not specified for this widget.
HorizontalAlignment	True	Enum	See Node's property HorizontalAlignment	TC_W77_06
HorizontalAlignmentEnabled	False	Bool	Enables configuration of the property HorizontalAlignment	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property has not been fully developed.
LayoutDirection	True	Enum	See Node's property LayoutDirection	TC_W77_20
LayoutDirectionEnabled	False	Bool	Enables configuration of the property LayoutDirection	
Margin	True	Margin	See Node's property Margin	TC_W77_19
MarginEnabled	False	Bool	Enables configuration of the property Margin	
MaximumSize	True	Vector2	See Node's property MaximumSize	TC_W77_15
MaximumSizeEnabled	False	Bool	Enables configuration of the property MaximumSize	
MinimumSize	True	Vector2	See Node's property MinimumSize	TC_W77_14
MinimumSizeEnabled	False	Bool	Enables configuration of the property MinimumSize	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PivotOffset	True	Vector2	See Node's property PivotOffset	Ignored: This property because this property is deprecated. Replaced and tested by property PivotPoint.
PivotOffsetEnabled	False	Bool	Enables configuration of the property PivotOffset	
PivotPoint	True	Vector2	See Node's property PivotPoint	TC_W77_18
PivotPointEnabled	False	Bool	Enables configuration of the property PivotPoint	
Position	True	Vector2	See Node's property Position	TC_W77_12
PositionEnabled	False	Bool	Enables configuration of the property Position	
RenderOrderRank	True	Int	See Node's property RenderOrderRank	TC_W77_11
RenderOrderRankEnabled	False	Bool	Enables configuration of the property RenderOrderRank	



RenderingEnabled	True	Bool	DEPRECATED. Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. The same node property is configured also by the widget property Visible which is available in all widgets. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This property is deprecated.
RenderingEnabledEnabled	False	Bool	DEPRECATED. Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. The same node property is configured also by the widget property Visible which is available in all widgets. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
Rotation	True	Float	See Node's property Rotation	TC_W77_18
RotationEnabled	False	Bool	Enables configuration of the property Rotation	
Row	True	Byte	See Node's property Row	TC_W77_05
RowEnabled	False	Bool	Enables configuration of the property Row	
RowSpan	True	Byte	See Node's property RowSpan	TC_W77_03
RowSpanEnabled	False	Bool	Enables configuration of the property RowSpan	
Scale	True	Vector2	See Node's property Scale	TC_W77_17
ScaleEnabled	False	Bool	Enables configuration of the property Scale	
ScopeMask	True	UInt	See Node's property ScopeMask	TC_W77_21
ScopeMaskEnabled	False	Bool	Enables configuration of the property ScopeMask	
Size	True	Vector2	See Node's property Size	TC_W77_13
SizeEnabled	False	Bool	Enables configuration of the property Size	
StretchBehavior	True	Enum	See Node's property StretchBehavior	TC_W77_16
StretchBehaviorEnabled	False	Bool	Enables configuration of the property StretchBehavior	
VerticalAlignment	True	Enum	See Node's property VerticalAlignment	TC_W77_10
VerticalAlignmentEnabled	False	Bool	Enables configuration of the property VerticalAlignment	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W77_09

VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
----------------	-------	------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

74.2 Message list

<i>Name</i>	<i>Subscribes</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	-------------------	--------------------	----------------	---------------------	-------------------

## Chapter 75

# OPSWidget2D

**Name:** OPSWidget2D

**Description:** OPS (Optical parking system) widget is used to display the vehicle graphics together with sensor field displays (SFD) around the car

**Category:** Extra

### 75.1 Property list

Name	Bindable	Type	Description	Test Scope
BlinkInterval	True	custom://List	List each of the sensor fields: Blinking time interval in msec.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
LeftRTADistanceValue	True	Byte	Left RTA distance value: (0x1,0x02,0x03)	
LeftRTAStatuslevel	True	Byte	Left RTA status: 0->No sensor, other values->Sensor available	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OPSDisplayMode	True	Byte	OPSDisplayMode: FrontAndRear or Around360 or RearOnly	TC_W44_03

OPSMini	False	Bool	If TRUE, Mini mode will be used	
OPSWidgetType	True	Enum	OPSWidgetType: BySector or BySegment	
RightRTADistanceValue	True	Byte	Right RTA distance value: (0x1,0x02,0x03)	
RightRTAStatuslevel	True	Byte	Right RTA status: 0->No sensor, other values->Sensor available	
SegmentPreDefinedDistanceValues	True	custom://List	Pre defined segment values upon which the actual segment values to be compared	
SensorFieldBlinking	True	custom://List	If true, the sensor field is blinking depending on the BlinkInterval	
SensorFieldHatched	True	custom://List	True if field is hatched, for each of the sensor fields.	
SensorFieldScanned	True	custom://List	True if field is scanned, for flanks.	
SensorsAvailable	True	custom://List	Availability of the sensors around the car	
SensorsSegmentValues	True	custom://List	Array of sensors which represent its availability: True or False	
StatusValues	True	custom://List	Array of status values to find the segment color for all sectors	
TrailerConnected	True	Bool	Flag to identify Trailer is Connected or not	TC_W44_03
VehicleType	True	UShort	Type of the vehicle	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 75.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
------	--------------	-------------	---------	--------------	------------

## Chapter 76

# OffscreenTouchProxyWidget2D

**Name:** OffscreenTouchProxyWidget2D

**Description:** Used to map the touch coordinates for offscreen render targets

**Category:** Under construction

### 76.1 Property list

Name	Bindable	Type	Description	Test Scope
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
Camera	False	Camera2D	Camera : Candra::Camera2D*	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: Not tested because this property is inherited and not specific to this Widget.

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W78_08
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	

Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTF is simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget.
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OffscreenScene	False	Node2D	OffscreenScene : Candra::Node2D*	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.

RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W78_01
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.



Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W78_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

76.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 77

# OverlayLayoutWidget2D

**Name:** OverlayLayoutWidget2D

**Description:** Sets an overlay layouter on a composite node. Add the widget to the composite and associate it to the composite itself. Don't use the nested level because it can create confusion and lead to errors. Also don't use this widget to set an overlay layouter on a normal group because it is more efficient to set it from the properties panel of that group.

**Category:** Layout

### 77.1 Property list

Name	Bindable	Type	Description	Test Scope
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	

NestedLevel	True	UInt	DEPRECATED: This property was required in the past when it was not possible to associate a widget directly to the composite. The value of this property represents the ancestor's level, relative to the associated node, which will be targeted by this widget. This property should not be used because it creates confusion and can lead to errors. Just associate the widget directly to the targeted composite.	
Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 77.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 78

# PageEditWidget2D

**Name:** PageEditWidget2D

**Description:** Provides support to arrange items inside a grid. See the widget user guide for details regarding how to configure it.

**Category:** Gesture

### 78.1 Property list

Name	Bindable	Type	Description	Test Scope
AlbumId	True	UInt	Editing is possible only within the pages from the same album.	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ArrangeItems	True	Bool	If enabled then the items will be arranged based on ItemCells value.	TC_W38_01
AutoArrange	True	Bool	If enabled then the items will be auto arranged on item drag.	TC_W38_02
CellSize	True	Vector2	Size in pixels of a cell.	TC_W38_03
ColumnCount	True	UShort	Number of columns.	TC_W38_04
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
EditingCellsNode	False	Node2D	If set to a resizable node then the editing cells will be marked. This node is visible only during editing.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing.
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	

Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTF is simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property has not been supported for testing.
ItemCells	True	Rectangle	Specifies the top left cell and the cell span for each item. If span is 0 (width or height is 0) then the item will be hidden.	TC_W38_06
ItemLocks	True	Bool	Provides support to lock items preventing them from being moved during editing.	TC_W38_07
ItemsNode	False	Node2D	If set then the items will be searched inside this node otherwise the widget node will be used. ItemsNode must be the same as Node or a descendent of Node.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.

Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
RowCount	True	UShort	Number of rows.	TC_W38_05
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W38_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 78.2 Message list

Name	Subscribers	Description	Members	Distribution Scope
------	-------------	-------------	---------	--------------------

PageEditUpdMsg	Model		AlbumId: AlbumIdType.	sequential	
			Data: PageEdit- Data::SharedPointer.		
			Status: enEdit- Status::Enum. +		



## Chapter 79

# PanelSlideWidget2D

**Name:** PanelSlideWidget2D

**Description:** Supports to slide the panel from 4 directions(HorizontalLeft, HorizontalRight, VeriticalTop and VerticalBottom) using drag and swipe gestures. It also can be opened and closed with PanelButton.

**Category:** Gesture

### 79.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
Direction	False	Enum	Direction in which panel should be opened	
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragThreshold	False	Float	It is a minimum limit to show or hide the panel in case of drag(value should be in percentage ex: 0.5 for 1/2 of bg length/width)	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	

Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTF is simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PanelButton	False	Widget	The button will open and close the Panel. This is an optional button, Panel can also be opened and closed with swipe/drag gestures.	
PanelNode	False	Node2D	Node which will contain the childrens of the panel. This will be visible when panel is opened and invisible when panel is closed	
PanelOpen	True	Bool	Panel will be in Opened state if it's true and Closed if it's false	TC_W14_14
PanelSize	False	Float	Size of the panel (i.e. panel width in case of horizontal direction and panel height in case of vertical direction)	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.

Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 79.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
PanelSlideReqMsgView			PanelAction: enPanelAction. +	sequential	

PanelStateMsg	Model, View, Con- troller		IsOpen: bool. +	sequential	
---------------	------------------------------------	--	-----------------	------------	--

## Chapter 80

# ProgressBarWidget2D

**Name:** ProgressBarWidget2D

**Description:** ProgressBarWidget2D implements the Non Interactive functionality.

**Category:** Range

### 80.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
BufferMaskNode	False	Node2D	This property holds the Mask Node which is used to selectively render parts of the BufferNode	
BufferValue	True	Float	This property holds the buffer's position with respect to the TotalValue	TC_W13_01
CurrentValue	True	Float	This property holds the current position of progress with respect to the TotalValue	TC_W13_02
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing for this widget
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property is dependent on EnableGroupWidget2D, so it will be tested in EnableGroupWidget2D section.

MaskNode	False	Node2D	This property holds the Mask Node which is used to selectively render parts of the FillNode	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
ProgressBarOrientationType	False	Enum	Select the orientation of the ProgressBar. Horizontal(Left, right), Vertical(Top, Bottom)	
ProgressBufferNode	False	Node2D	The buffer node is the indicator of the current buffer value of the progressbarWidget	
ProgressFillNode	False	Node2D	The fill node is the indicator of the current value of the progressbarWidget	
ProgressFillType	False	Enum	The strategy used to fill the Progressbar which can be either based on Value or a Timer	
RestartTimer	True	Bool	This property controls when to Restart the internal timer which will update the Progressbar current position	TC_W13_05
StartTimer	True	Bool	This property controls when to start the internal timer which will update the Progressbar current position	TC_W13_04
TimerStepCount	False	UShort	Total number of steps for Progressbar to fill completely for the given TimerValue	
TimerValue	False	UInt	The timer duration in which the ProgressBar will be filled completely based on an internal timer value. The internal timer is started based on the StartTimer property value	
TotalValue	True	Float	This property holds the progress bar's maximum value	TC_W13_03
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W13_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 80.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
------	--------------	-------------	---------	--------------	------------

## Chapter 81

# RichTextHighlightWidget2D

**Name:** RichTextHighlightWidget2D

**Description:** Provides highlight support for rich text. See Widget User Guide for details on how to use this widget.

**Category:** RichText

### 81.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AllOccurrences	False	Bool	If set to true, all occurrences of the string defined in property 'Highlight Text' are highlighted. False highlights only the first occurrence.	
BackgroundColor	True	Color	The background color to be set on the highlighted text.	TC_W48_02
BackgroundColorEnabled	False	Bool	If this property is set to true, the color for the highlighted background can be defined property 'Background Color'.	
CaseSensitive	False	Bool	If set to true, property 'Highlight Text' is case sensitive. False searches case insensitive.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W48_08



HighlightMode	False	Enum	Defines how to interpret string property HighlightText.  - Mask All occurrences of the string defined in property 'Highlight Text' are highlighted.  - Range Defines character index ranges to be highlighted.  Example to highlight characters at index 3, 7, 8, 9 and 15: '3,7-9,15'	
HighlightText	True	custom://String	Criteria to highlight Text. See property 'Highlight Mode'.	TC_W48_01
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and not specific to this Widget..
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
TextColor	True	Color	The color to be set on the highlighted text.	TC_W48_03
TextColorEnabled	False	Bool	If this property is set to true, the color for the highlighted text can be defined with property 'Text Color'.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W48_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 81.2 Message list

Name	Subscription	Description	Members	Distribution	Test Scope
------	--------------	-------------	---------	--------------	------------

## Chapter 82

# RichTextMarqueeWidget2D

**Name:** RichTextMarqueeWidget2D

**Description:** Provides horizontal scrolling animation support for rich text. See Widget User Guide for details on how to use this widget.

**Category:** RichText

### 82.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Bounce	False	Bool	If set, the animation reverts the direction at the end (text runs out to the right again).	
DelayEnd	False	UInt	Defines a delay in milliseconds at the end of the marquee animation.	
DelayStart	False	UInt	Defines a delay in milliseconds at the start of the marquee animation.	
Direction	False	Enum	The animation direction is defined by one of these options:  - Text: The first strong character of the text defines the animation direction. - Culture: The current culture defines the animation direction. - Node: The node effective layout direction is used. - LeftToRight: Animation always runs left-to-right (usually for RTL text). - RightToLeft: Animation always runs right-to-left (usually for LTR text).	

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
EndEmpty	False	Bool	If set, the animation ends with an empty field, which means that all text is moved to the left.  If not set, the animation ends as soon as the last character on the right side is completely visible.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
RepeatCount	False	UInt	Number of animation sequences to run, zero means infinite.	
Speed	False	UInt	Animation speed in pixel per second.	
StartEmpty	False	Bool	If set, the animation starts with an empty field and all text is moved in from the right.  If not set, the visible part of the text is displayed from the beginning and the hidden part is moved in from the right.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 82.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 83

# RichTextWidget2D

**Name:** RichTextWidget2D

**Description:** Renders formatted and styled text elements. See Widget User Guide for details on how to use this widget.

**Category:** RichText

### 83.1 Property list

Name	Bindable	Type	Description	Test Scope
AsynchronousRendering	False	Bool	Enables asynchronous parsing and rendering.	
BaselineOffset	False	Int	If set to a value $\geq 0$ this value defines the vertical distance in pixel between node position and the baseline position of the first rendered text line.	
CachedRendering	False	Bool	Enables fast rendering of big content by creating cache file. Example: For oss_license file	
Color	True	Color		TC_W40_01
ColorEnabled	False	Bool	If this property is set to true, the text color can be defined with property 'Color'.	
CultureDependentAlignment	False	Bool	Defines whether the horizontal alignment depends on the culture or not.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W40_08
HorizontalAlignment	False	Enum		
HorizontalAlignmentEnabled	False	Bool	If this property is set to true, the horizontal alignment can be defined with property 'Horizontal Alignment'.	

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
MaxNumberOfLines	False	UShort	If set to a value greater than zero, text truncation happens at last specified line. If truncation should show an ellipsis ('...') at the end, 'text-overflow:ellipsis' has to be defined as style.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
SliceHeight	False	UShort	The pixel height for all slices. Value 0 defines a slice height that is automatically set to the viewport height.	
SlicedRendering	False	Bool	Enables rendering into separate bitmap slices.	
Source	True	custom://	<p>Depending on property 'Source Type' the</p> <ul style="list-style-type: none"> <li>- String This can either be plain text or a string including HTML tags like &lt;p style='font-size:2em;color:blue'&gt;This is large blue text.&lt;/p&gt;</li> <li>- URL URL to a resource that holds the HTML formatted text. See property 'Source Type' for supported URL formats.</li> </ul>	TC_W40_03

SourceType	False	Enum	<p>Defines how to interpret string property 'Source'.</p> <ul style="list-style-type: none"> <li>- String Source defines the HTML formatted string.</li> <li>- URL Source is an URL to a resource that holds the HTML formatted string. Format for URL is [scheme]:[authority]/[path].</li> </ul> <p>Currently supported URLs are</p> <ul style="list-style-type: none"> <li>- asset:path/[path] the path to the source in format shown as Cander Name in property grid. Example: 'asset:path/Resources#sample.html'</li> <li>- asset:id/[id] the asset id as decimal value as defined with property 'Id'. Example: 'asset:id/1000'</li> </ul>	
Style	True	custom://string	<p>String user style that overrides the style properties defined by property 'Style Sheet Url'. Example:</p>	TC_W40_04

StyleSheetUrl	False	custom:// <del>StyleSheet</del>	<p>The URL in format [scheme]:[authority]/[path] to the HTML style sheet document. See property 'Source Type' for URL examples.</p> <p>The following CSS properties are supported:</p> <ul style="list-style-type: none"> <li>- background-color</li> <li>- color</li> <li>- display</li> <li>- font-family</li> <li>- font-size</li> <li>- height</li> <li>- line-height</li> <li>- margin-bottom</li> <li>- margin-left</li> <li>- margin-right</li> <li>- margin-top</li> <li>- margin</li> <li>- overflow</li> <li>- padding-bottom</li> <li>- padding-left</li> <li>- padding-right</li> <li>- padding-top</li> <li>- padding</li> <li>- position</li> <li>- text-align</li> <li>- text-overflow</li> <li>- vertical-align</li> <li>- white-space</li> <li>- width</li> </ul>	
TextIdsList	False	custom:// <del>TextIdList</del>	TextIdList	
TextStyle	True	TextStyle	<p>This text style is taken if not defined in HTML style.</p> <p>To define a text style the following CSS properties has to be used (example):</p> <ul style="list-style-type: none"> <li>- text-family:font-family:Resources#Fonts#Arial</li> <li>- text-size:20px</li> </ul>	TC_W40_02
Visible	True	Bool	<p>Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.</p>	TC_W40_09
VisibleEnabled	False	Bool	<p>Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.</p>	



83.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 84

# ScopeWidget2D

**Name:** ScopeWidget2D

**Description:** Sets the scope mask of a node and applies it also to its descendents. Scoping is described in detail in the CGI Studio documentation.

**Category:** Common

### 84.1 Property list

Name	Bindable	Type	Description	Test Scope
ApplyOnDescendents	True	Bool	Applies the node scope mask recursively to all the descendents.	TC_W82_02
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

ScopeMask	True	UInt	Scope mask to be applied on the associated node and, if configured so, also on descendents. Use a calculator to determine the decimal value containing the required scopes. For example, the decimal value 9 enables Scope 1 and Scope 4.	TC_W82_01
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W82_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 84.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 85

# ScrollBarButtonWidget2D

**Name:** ScrollBarButtonWidget2D

**Description:** It triggers scrolling actions for a ScrollBarWidget2D. Its node must be a child of the node of the ScrollBarWidget2D it should controll

**Category:** Range

### 85.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Amount	False	Int	The amount for the change type. For example, if set to 2 and ChangeType is set to Previous, it will request an action to go 2 items back.	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ChangeType	False	Enum	The type of action that will be issued to the scrollbar.	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
EditorFocused	False	Bool	Sets the focused flag in order to preview the appearance in SceneComposer. It has no effect in the simulation or on the target.	
EditorPressed	False	Bool	Sets the pressed flag in order to preview the appearance in SceneComposer. It has no effect in the simulation or on the target.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.

FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTF is simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
IsActive	True	Bool	Indicates if the button is active. For toggle/radio button active means checked/selected. For normal button active means that the option/action is active (for example track is playing, station is tuned, etc).	TC_W04_02
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OnPress	False	Bool	If set to true, then the action is done when pressing the button, otherwise it's done when releasing the button.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PostButtonReactionMsg	False	Bool	ButtonReactionMsg messages will be posted only if this property is true.	
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	

RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TimerConfiguration	False	Enum	Configures the timer.	
TouchHandler	True	Int	Specifies the id of the touch handler for this button. A touch handler can provide a custom sensitive area for buttons (for example circle, triangle, etc). Touch handlers are registered at application start-up.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
TouchableArea	False	Rectangle	Configures the touchable area of the button. X and Y are coordinates relative to the upper left corner of the node effective bounding rectangle. If width or height are zero or negative the effective bounding rectangle of the node will be used as touchable area (X and Y will also be ignored).	TC_W04_03
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

85.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------



## Chapter 86

# ScrollBarWidget2D

**Name:** ScrollBarWidget2D

**Description:** Provides support to scroll through scrollable elements (e.g. a list).

**Category:** Range

### 86.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AdditionalPageBackwardNode	False	Node2D	An auxiliary node acting as a page backward button. This node is not part of the knob movement area.	
AdditionalPageButtons	False	Bool	Use additional page buttons that are not resized and are not part of the knob movement area.	
AdditionalPageForwardNode	False	Node2D	An auxiliary node acting as a page forward button. This node is not part of the knob movement area.	
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
BackwardNode	False	Node2D	The node acting as back button.	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time

DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: Not tested because this property is inherited and not specific to this Widget.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: ScrollBar-Widget2D is implemented to consume only Legacy-Touch gesture events. No reaction on touching the nodes when Drag is enabled.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DynamicSliderSize	False	Bool	Whether or not the slider size shall reflect the number of items in the list, if set to true the specified slider size will represent the minimum size.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W83_08

FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
ForwardNode	False	Node2D	The node acting as forward button.	
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
KnobDragMode	False	Enum	The mode the knob/slider) moves when dragged: ListPosition follows the list current position relative to its whole content; PointerPosition follows the pointer.	TC_W30_01
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

OverscrollSize	True	Float	The percentage of one of the overscroll areas relative to the size of the slider back. There are two regions of overscroll: one at the begining of the scrollbar and one at the end.	TC_W83_02
PageBackwardNode	False	Node2D	The node acting as a page back button; the part between the knob and BackwardNode.	
PageForwardNode	False	Node2D	The node acting as a page forward button; the part between the knob and ForwardNode.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	Ignored: ScrollBar-Widget2D is implemented to consume only Legacy-Touch gesture events. No reaction on touching the nodes when PressHold is enabled.
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: ScrollBar-Widget2D is implemented to consume only Legacy-Touch gesture events. No reaction on touching the nodes when PressRepeat is enabled.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.

RepeatMessage	False	UInt	The time between triggering two consecutive events(back, forward, pageBack or pageForward) when the corresponding button is kept pressed.	
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollableNode	False	Node2D	The node containing a FlexScrollable widget.	
SliderNode	False	Node2D	The node that will act as a slider and can be dragged. Its parent must be the ancestor of the PageBackwardNode and PageForwardNode.	
SliderSize	True	Float	The percentage of the slider size relative to the size of the slider back.	TC_W83_01
Swipe	True	Bool	Enable swipe gesture detection	Ignored: ScrollBar-Widget2D is implemented to consume only Legacy-Touch gesture events. No reaction on touching the nodes when Swipe is enabled.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: ScrollBar-Widget2D is implemented to consume only Legacy-Touch gesture events. No reaction on touching the nodes when Tap is enabled.

TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W83_08
TrackPressMode	False	Enum	Specifies the scrollbar behavior when it is touched either above or below the knob - PageScroll: The list is scrolled one page up / down, JumpToTouchCoordinate: Causes the knob to jump to the touch coordinate	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W83_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 86.2 Message list

Name	Subscribers	Description	Members	Distribution	Test Scope
ScrollBarBubbleUpMsg	ScrollBar, DataModel		Event: enBubbleEvent::Enum.  RegionIndex: FeatStd::UInt.  Text: FeatStd::String. +	sequential	

## Chapter 87

# ScrollableRichTextWidget2D

**Name:** ScrollableRichTextWidget2D

**Description:** Provides vertical scrolling support for rich text. See Widget User Guide for details on how to use this widget.

**Category:** RichText

### 87.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: This property is inherited and only works for Button and Slider widgets.

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	TC_W39_01
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W39_08
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.



GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	TC_W39_04
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	TC_W39_06
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: This property is inherited and not specific to this Widget.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollId	True	UInt	A unique identifier be used to communicate with e.g. the data model.	TC_W39_10
Swipe	True	Bool	Enable swipe gesture detection	TC_W39_02

SwipeDeceleration	False	UInt	Scroll deceleration for swipe gesture in [pixel / s <sup>2</sup>	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
SwipeMaxDistance	False	UInt	If swipe velocity and configured deceleration would result in a scroll distance that is greater than this value, the deceleration is increased to limit the scroll distance.  A value of 0 means no limit.	
Tap	True	Bool	Enable press and tap gesture detection	TC_W39_05
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W39_07
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W39_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled.  Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 87.2 Message list

Name	Subscribers	Description	Members	Distribution	Test Scope
------	-------------	-------------	---------	--------------	------------

ScrollableRichText	UpdateModel, View, Model	CurrentScrollPosition: Feat-Std::UInt32.  MaxScrollPosition: Feat-Std::UInt32. +	Sequential	IC_W39_03
--------------------	--------------------------	-------------------------------------------------------------------------------------------	------------	-----------

## Chapter 88

# ScrollableTextWidget2D

**Name:** ScrollableTextWidget2D

**Description:** Allows the display of formatted, multiline texts.

**Category:** Text

### 88.1 Property list

Name	Bindable	Type	Description	Test Scope
AlphaValue	True	Float	Sets the alpha value to of the associated node.	TC_W20_03
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ColorTable	False	custom://	Contains all colors which will be used whenever a '\cxx' is found in the text.	TC_W20_06
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
CultureDependentAlignment	False	Bool	If disabled then HLeft and HRight will be inverted for cultures with right to left text direction	

DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: This widget property is deprecated and has been replaced in ScrollableRichTextWidget.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: This widget property is deprecated and has been replaced in ScrollableRichTextWidget.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W20_04

FileName	True	custom://	File name of the text that will be displayed if the TextMode is set to SourceFile.	Ignored: Not tested because this property is having a reported problem but the fix is not available. Kindly see in RTC_668420.
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: This widget property is deprecated and has been replaced by ScrollableRichTextWidget.
HorizontalAlignment	False	Enum	Horizontal alignment of the text. RT_Leading: Aligns text to the left side for LTR and right side for RTL texts. RT_Middle: Aligned to the center. RT_Trailing: Aligns text to the right side for LTR and left side for RTL texts.	

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget property is deprecated and has been replaced by ScrollableRichTextWidget.
LinespacingFactor	False	Float	Line spacing used in mutli line source string.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NormalTextColor	False	Color	Default color to be used to display the text.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PostUpdateMessage	False	Bool	Decision on posting ScrollableTextUpdMsg when there is a change in current and maximum scroll position value is decided based on this property	
PressHold	True	Bool	Enable hold gesture detection	Ignored: This widget property is deprecated and has been replaced in ScrollableRichTextWidget.
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: This widget property is deprecated and has been replaced in ScrollableRichTextWidget.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.

Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollableTextId	False	UInt	Unique identifier of the scrollable text	
Style	False	TextStyle	Default style to be used for text rendering.	
Swipe	True	Bool	Enable swipe gesture detection	Ignored: This widget property is deprecated and has been replaced in ScrollableRichTextWidget.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
SwipeFriction	False	Int	Controls the intensity of swipe. Increasing the friction decreases the swipe intensity	
TabStopTable	False	custom://	Contains all tab stop positions which will be used whenever a '\txx' is found in the text.	TC_W20_06
Tap	True	Bool	Enable press and tap gesture detection	Ignored: This widget property is deprecated and has been replaced in ScrollableRichTextWidget.
Text	True	custom://	String to be displayed.	TC_W20_01
TextAreaSize	True	Vector2	The text area size to be used for truncation and alignment calculations.	TC_W20_02
TextId	True	UInt	Id of the text that will be displayed if the TextMode is set to SourceTextId.	Ignored: This widget property is deprecated and has been replaced in ScrollableRichTextWidget.
TextIdsList	False	custom://	TextIdList of text ids used by this widget	
TextSource	True	Enum	Controls which source should be used as input.	TC_W20_01



TextWrapMode	False	Enum	Controls how to long lines should be wrapped. RT_WrapNone: No wrapping happens. RT_WrapWords: Whole words will be wrapped. If the word doesn't fit inside an empty line RT_WrapCharacters will be used. RT_WrapCharacters: Individual characters will be wrapped. If no character fits an empty line at least one will be drawn.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	Ignored: This widget property is deprecated and has been replaced in ScrollableRichTextWidget.
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
VerticalAlignment	False	Enum	Vertical alignment of the text. RT_Leading: Aligns text to the top side for TTB and bottom side for BTT texts. RT_Middle: Aligned to the center. RT_Trailing: Aligns text to the bottom side for TTB and top side for BTT texts.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W20_05
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

88.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
ScrollableTextUpdate	Controller, View, Model		CurrentScrollPosition: Courier::UInt32.  MaxScrollPosition: Courier::UInt32. +	Sequential	

## Chapter 89

# SimpleButtonWidget3D

**Name:** SimpleButtonWidget3D

**Description:** Provide a simple 3D button, this button should link a kind of 3D note (billboard, mesh. . . ), it also contains several states (enabled/disabled, pressed), all these states can be configured with a bitmap

**Category:** Button3D

### 89.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledBitmap	True	Bitmap	Bitmap used inside the texture of Apperance when Button is disabled	
Enabled	False	Bool	Enabled: Enable or disable the widget	
Name	False	CharArray	The name of the widget instance	
Node	False	Node3D	The associated node of the widget.	
NormalBitmap	True	Bitmap	Bitmap used inside the texture of Apperance when Button is enabled, not pressed	
PressedBitmap	True	Bitmap	Bitmap used inside the texture of Apperance when Button is pressed, enabled	

TouchPriority	False	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.

## 89.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 90

# SizeWidget2D

**Name:** SizeWidget2D

**Description:** Controls the size property of a node.

**Category:** Layout

### 90.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and not specific to this Widget.
Name	False	CharArray	The name of the widget instance	
NestedLevel	True	UInt	The ancestor's level, relative to the associated node, that will have the Size property modified.	TC_W84_02
Node	False	Node2D	The associated node of the widget.	
Size	True	Vector2	The value to set for the Size property of the ancestor node.	TC_W84_01

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W84_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

90.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 91

# SliderWidget2D

**Name:** SliderWidget2D

**Description:** The slider is the widget for controlling a value by positioning a knob / thumb in a legal range.

**Category:** Range

### 91.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
BitmapFillNode	False	Node2D	Node to be used for the bitmap corresponding to the current value.	
ContinuousUpdateOnDrag	False	Bool	Set to true if value updates are required while dragging, set to false if update is required only at the end of drag.	
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
CurrentValue	True	Float	Current value of the slider marker, should be in the range between MinVal and MaxVal.	TC_W14_04
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	TC_W14_01

DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W14_03
FillerPositionIsKnobCenter	False	Bool	Filler starts at center of marker, for example round knob.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	



Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTF is simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: Not tested because this property is inherited and not specific to this Widget.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
MarkerMovement	False	Enum	Defines the way the slider marker moves on tap or drag. For TimedStepMovement the marker will move one step on Press and will keep moving one step for every Repeat, this means that PressRepeat needs to be enabled and Drag needs to be disabled to prevent jumping on Drag.	
MarkerNode	False	Node2D	The Scene Tree Node that supports a bitmap used for the slider marker (Knob, Thumb).	
MarkerNormalBitmap	False	UInt	DEPRECATED! Use the appearance concept to change bitmaps based on state.	
MarkerSelectedBitmap	False	UInt	DEPRECATED! Use the appearance concept to change bitmaps based on state.	
MaxVal	True	Float	Maximum value for slider at last marker (e.g Tuner 108.0 Mhz)	TC_W14_07
MinVal	True	Float	Minimum value for slider at first marker (e.g Tuner 88.0 Mhz)	TC_W14_06
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NumOfSteps	True	UShort	Total number steps for the marker movement. The stepsize will be calculated based on given range and number of steps. (e.g. range 0..20, steps=10 -> stepsize 2)	TC_W14_05
PaddingMaxVal	False	Float	Maximum graphical padding value on right hand, offset in pixel to the last valid marker for maximum value (calculated from the end of the background scale bitmap)	

PaddingMinVal	False	Float	Minimum graphical padding value on left hand, offset in pixel to the first valid marker for minimum value (calculated from the beginning of the background scale bitmap).	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SliderBackGround	False	Node2D	The Scene Tree Node that renders sliders background grid with scale markers on top.	
SliderOrientation	False	Enum	Defines the orientation of the slider marker movement and the slider bar.	
StepSize	True	Float	Defines the stepsize for movement of slidermarker.	TC_W14_05
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	

Tap	True	Bool	Enable press and tap gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	TC_W14_01
UseNumofSteps	False	Bool	If this property is set to true NumOfSteps will be visible and used, else StepSize will be visible and used.	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W14_02
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 91.2 Message list

Name	Subscribers	Description	Members	Distribution	Test Scope
------	-------------	-------------	---------	--------------	------------

## Chapter 92

# SolidColorEffectWidget2D

**Name:** SolidColorEffectWidget2D

**Description:** Manipulates the Size and the FillColor properties of the existing SolidColorBrush effect of a RenderNode.

**Category:** Common

### 92.1 Property list

Name	Bindable	Type	Description	Test Scope
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: No test needed for this property because this prop is inherited but not specified for this widget.
FillColor	True	Color	Fill color to be set on the SolidColorBrush effect inside the RenderNode.	TC_W96_01
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: No test needed for this property because this property has not been fully developed.
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

Size	True	Vector2	Size to be set on the SolidColorBrush effect inside the RenderNode. The property value is not used if the X field or the Y field is less than zero.	TC_W96_02
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W96_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 92.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 93

# SpellerWidget2D

**Name:** SpellerWidget2D

**Description:** SpellerWidget2D provides an on-screen keyboard to interact with user input

**Category:** Input

### 93.1 Property list

Name	Bindable	Type	Description	Test Scope
ActivateLayoutIndex	True	UInt	The Layout Index which has to be shown currently	TC_W15_09
ActiveCountry	True	UInt	Name for the clock face image.	TC_W15_04
ActiveLanguage	True	UInt	Currently active language	TC_W15_03
AllCharLayoutDisable	True	Bool	Disable all Char layout buttons	TC_W15_06
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
AutoLayoutToggle	True	Bool	Auto Toggle Layout Based on ValidCharSet and layout sequence in Match Mode	TC_W15_07
ConfigureFocusGroup	True	Bool	Focus group is configured explicitly or uses default configuration.	Ignored: The TTFis simulation for the gesture action is not ready yet.
ConfigureSpellerScenes	False	Bool	Replace the render target of the cameras in the speller layout scenes with the one associated to the first camera in current scene.	

ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It can't be changed during run time
DefaultFocusOrder	True	Short	Focus order of the element which should become focused when this group becomes active.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: inherited property.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: inherited property.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W15_08

FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusLayer	True	Short	Specifies to which focus layer this group belongs. There should be distinct layers for main surface screens, subspeller, popups to prevent focusing of elements in the main screen when a popup is displayed. Only elements in the groups with the highest layer will be focused.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
FocusWrapAround	True	Bool	Focus wraps around to first/last element.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: inherited property.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: inherited property.



KeypadFormat	True	Enum	Currently active keypad format - e.g. QWERTY or QBC.	TC_W15_02
Mode	False	Enum	The SpellerMode - either FreeMode or MatchMode.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PreserveFocus	True	Bool	Current focused element is preserved when group becomes inactive.	Ignored: The TTF is simulation for the gesture action is not ready yet.
PressHold	True	Bool	Enable hold gesture detection	Ignored: inherited property.
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: inherited property.
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SpellerKeyPressed	True	custom://string	Update EditField when Speller is in FreeMode.	TC_W15_01
SpellerScenesPath	False	custom://string	Speller layout scenes should be in this Candra path.	
Swipe	True	Bool	Enable swipe gesture detection	Ignored: inherited property.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: inherited property.

TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	Ignored: inherited property.
Type	False	Enum	Speller type.	
UseOnlyButtonsWithText	False	Bool	If enabled only the buttons which have a text widget will be used for the speller layout (legacy behaviour), otherwise all the buttons will be used allowing to use non text buttons for control/delete/space	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
UserDefinedSequence	True	custom://\$id\$	User Defined Sequence	TC_W15_05
ValidCharSet	True	custom://\$id\$	Valid char set, requested by match mode	TC_W15_10
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This property is intentionally not handled.
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
ZeroKeyLongPressed	False	Bool	To update zero long press with '+' sign	

## 93.2 Message list

Name	Subscribers	Description	Members	Distribution	Test Scope
------	-------------	-------------	---------	--------------	------------

SpellerCtrlKeyPressed	Control Model View	<p>Courier Message from Speller to notify the subscribers about the Control key pressed</p> <ul style="list-style-type: none"> <li>- This message is applicable for both FreeMode and MatchMode Speller</li> <li>- ViewId: corresponds to the view from where the message is fired</li> <li>- Sender: corresponds to the widget instance which fires the message</li> <li>- CtrlKeyChar: corresponds to the UCS4(UTF-32) encoded code point Control character which is pressed</li> </ul>	CtrlKeyChar: Courier::UInt32. +		TC_W15_11
SpellerKeyPressed	Model	<p>Courier Message from Speller to notify the subscribers about the key pressed</p> <ul style="list-style-type: none"> <li>- This message is fired only in Match mode. In Free mode, the character pressed is updated via DataBinding</li> <li>- ViewId: corresponds to the view from where the message is fired</li> <li>- Sender: corresponds to the widget instance which fires the message</li> <li>- KeyChar: corresponds to the character which is pressed (will be changed to UCS4(UTF-32) encoded code point in future)</li> </ul>	KeyChar: Can-dera::String. +		TC_W15_12
SpellerStatusChanged	Widget Model	<p>Courier Message which Speller expects when its child-button is explicitly disabled by external source dynamically.</p> <ul style="list-style-type: none"> <li>- This message notifies Speller to re-evaluate the focus handling (if required).</li> <li>- Do note that this message is applicable only for FreeMode Speller and should not never be fired in MatchMode Speller -</li> <li>- as the focus handling in MatchMode will always be re-evaluated on receiving the update for ValidCharSet.</li> </ul>			TC_W15_13
SubSpellerStatusUpdated	Widget Model	<p>Courier Message to notify the status of a Sub Speller to the subscribers for the key pressed</p> <ul style="list-style-type: none"> <li>- ViewId: corresponds to the view from where the message is fired</li> <li>- Sender: corresponds to the widget instance which fires the message</li> <li>- KeyChar: corresponds to the character which is pressed (will be changed to UCS4(UTF-32) encoded code point in future)</li> <li>- Status: Sub Speller status like sub Speller shown, hidden etc</li> </ul>	<p>KeyChar: Can-dera::String.</p> <p>Status: hmibase::widget::speller::enSubSpellerStatus::En +</p>	sequential	TC_W15_14

## Chapter 94

# StepAnimationWidget2D

**Name:** StepAnimationWidget2D

**Description:** Provides support to animate properties with discrete values. This has the advantage that update is not performed continuously resulting in reduced CPU usage. It is suited for wait animations, blink animations and other situations when animations have to run for a longer time. The widget uses a timer to trigger the updates and this timer can be shared between multiple instances so they all update in the same time.

**Category:** Animation

### 94.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AnimatedWidget	False	Widget	Widget to be animated.	
ChannelCount	False	Byte	Channel count specifies how many values are required for each keyframe. Depending on the type of the animated property the number of values for each keyframe is between 1 and 4. 1 value is required for simple types (bool, int, float); 2 values are required for Vector2; 3 values are required for Vector3; 4 values are required for Color, Rectangle and Margin; For example if it is required to animate a property of type Vector2, the ChannelCount must be specified as 2 so that 2 values will be required for each keyframe (values at positions 0 and 1 are for the first keyframe and so on).	

Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing for this widget
EndKeyframeIndex	True	Int	Index of the keyframe which will be set when the animation is stopped. Useful for blink animations where the final node state should be visible.	TC_W86_02
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property is dependent on EnableGroupWidget2D, so it will be tested in EnableGroupWidget2D section.
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PropertyAnimation	False	Enum	Identifies the property which has to be animated. Additional information is required for widget properties.	
SharedTimerInstance	True	Byte	Identifies the shared instance of the internal timer. Within the same scene all widgets can use the same timer instance so that they are updated in the same time.	
ShouldRun	True	Bool	Animation runs when this property is enabled.	TC_W86_01
TimerDelay	True	UInt	Timeout used by the timer. If multiple widgets share the same timer, only the timeout value of the first started animation will be used.	TC_W86_04
Values	False	Float	Stores the keyframe values. Depending on the type of the animated property the number of values for each keyframe is between 1 and 4. 1 value is required for simple types (bool, int, float); 2 values are required for Vector2; 3 values are required for Vector3; 4 values are required for Color, Rectangle and Margin;	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W86_09

VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
WidgetProperty	False	custom://Widget	Widget property to be animated.	

## 94.2 Message list

<i>Name</i>	<i>Subscribable</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 95

# SurfaceInputRegionWidget2D

**Name:** SurfaceInputRegionWidget2D

**Description:** Marks a node in the scene as Wayland input region as required for Gen4.

**Category:** Input

### 95.1 Property list

Name	Bindable	Type	Description	Test Scope
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Operation	True	Enum	Configures how this input region is used: Ignore - not used Add - used as a touchable rectangle on the surface Subtract - used as non touchable rectangle on the surface	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

95.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------



## Chapter 96

# SurfaceRegionWidget2D

**Name:** SurfaceRegionWidget2D

**Description:** Marks a node in the scene as Wayland input or opaque region as required for Gen4.

**Category:** Common

### 96.1 Property list

Name	Bindable	Type	Description	Test Scope
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
InputOperation	True	Enum	Configures the input region: Ignore - not used Add - used as a touchable rectangle on the surface Subtract - used as non touchable rectangle on the surface	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

OpaqueOperation	True	Enum	Configures the opaque region: Ignore - not used Add - used as an opaque rectangle on the surface Subtract - used as non opaque rectangle on the surface	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 96.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 97

# SwitchWidget2D

**Name:** SwitchWidget2D

**Description:** Sets the RenderingEnabled property of is children nodes based on the index property (allowing to switch between nodes by a bindable widget property).

**Category:** Common

### 97.1 Property list

Name	Bindable	Type	Description	Test Scope
BitmaskEnabled	True	Bool	If set to true, the Index will be considered as bitmask and each bit will toggle the visibility of a node. Bit 0 from Index will control the first child node, bit 1 the second child node, etc. Up to 32 nodes can be controlled this way.	TC_W87_02
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This derived property is not used in Switch widget.
Index	True	UInt	Index of the associated node's child that will be visible.	TC_W87_01
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property has no implementation in Switch widget.
Name	False	CharArray	The name of the widget instance	

Node	False	Node2D	The associated node of the widget.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W87_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 97.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 98

# TextAreaWidget2D

**Name:** `TextAreaWidget2D`

**Description:** Renders formatted text into a given area.

**Category:** Text

### 98.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
ActiveTextColor	True	Color	Color to be used to display the text in active state.	TC_W19_02
AlphaValue	True	Float	Sets the alpha value to of the associated node.	TC_W19_01
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTextColor	True	Color	Color to be used to display the text in disabled state.	TC_W19_03

DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	Ignored: Not tested because this property is inherited and not specific to this Widget.
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTF is simulation for the gesture action is not ready yet.
Drag	True	Bool	Enable drag gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTF is simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: Not tested because this property is inherited and not specific to this Widget.
EnableDynamicText	True	Bool	Enables dynamic text processing.	Ignored: Not tested because this property is inherited and not specific to this Widget.

Enabled	True	Bool	Sets of the text to Enabled/Disabled state.	Ignored: Not tested because this property is inherited and not specific to this Widget.
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.
GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	Ignored: Not tested because this property is inherited and not specific to this Widget.
HighlightCount	True	UShort	The number of the characters which are highlighted. This property is ignored if HighlightText is set.	TC_W19_06
HighlightMode	True	Enum	Switch between start/count and string based highlighting.	TC_W19_04
HighlightStartIndex	True	UShort	The index of the character where the highlighting starts. This property is ignored if HighlightText is set.	TC_W19_06
HighlightText	True	custom://string	String of the set text which should be highlighted. Only the first occurrence will be highlighted.	TC_W19_07
HighlightTextColor	True	Color	Color for highlighted text characters.	TC_W19_05

HorizontalAlignment	False	Enum	Horizontal alignment of the text. RT_Leading: Aligns text to the left side for LTR and right side for RTL texts. RT_Middle: Aligned to the center. RT_Trailing: Aligns text to the right side for LTR and left side for RTL texts.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
ItemOrder	False	Byte	Item order for widget which would be useful inside a list.	
LinespacingFactor	False	Float	LineSpacing used e.g. buttons with two lines, a value of one is the font height	
MaximumNumberOfLines	False	UShort	Maximum Number Of Lines, default is one.	
MultiLineLayouting	False	Bool	Sets the property of MultiLineLayouting to show content in multiple lines	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NormalTextColor	False	Color	Default color to be used to display the text.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressRepeat	True	Bool	Enable repeat gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.



RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.
Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
ScrollEndDelay	True	UShort	Delay after the scrolling reached the end position in milliseconds before jumping to the normal position.	Ignored: Not tested because this property is not currently activated in the test app
ScrollSpeed	True	UShort	Scroll speed in pixel per seconds.	Ignored: Not tested because this property is not currently activated in the test app
ScrollStartDelay	True	UShort	Delay before the scrolling is started in milliseconds after receiving the start event.	Ignored: Not tested because this property is not currently activated in the test app
Style	False	TextStyle	Default style to be used for text rendering.	
Swipe	True	Bool	Enable swipe gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.

Text	True	custom://String	String to be displayed.	Ignored: This widget property is deprecated and has been replaced in RichTextWidget.
TextAreaSize	True	Vector2	The text area size to be used for truncation and alignment calculations.	Ignored: This widget property is deprecated and has been replaced in RichTextWidget.
TextIdsList	False	custom://TextIdList	TextIdList ids used by this widget	
TextParameter1	True	custom://String	This text will replace any '%1' found in the text string. Use '%x' to write '%x'.	Ignored: This widget property is deprecated and has been replaced in TextExtensionWidget.
TextParameter2	True	custom://String	This text will replace any '%2' found in the text string. Use '%x' to write '%x'.	Ignored: This widget property is deprecated and has been replaced in TextExtensionWidget.
TextParameter3	True	custom://String	This text will replace any '%3' found in the text string. Use '%x' to write '%x'.	Ignored: This widget property is deprecated and has been replaced in TextExtensionWidget.
TextParameter4	True	custom://String	This text will replace any '%4' found in the text string. Use '%x' to write '%x'.	Ignored: This widget property is deprecated and has been replaced in TextExtensionWidget.

TextParameter5	True	custom://	This text will replace any '%5' found in the text string. Use '%x' to write '%x'.	Ignored: This widget property is deprecated and has been replaced in TextExtensionWidget.
TextParameter6	True	custom://	This text will replace any '%6' found in the text string. Use '%x' to write '%x'.	Ignored: This widget property is deprecated and has been replaced in TextExtensionWidget.
TextWidgetMode	True	Enum	The widget mode Normal - default mode, Invisible - the text box is not displayed, ColorFill - TextBox is filled with Color, Active - ActiveColor is used	Ignored: This widget property is deprecated and has been replaced in RichTextWidget.
TextWrapMode	False	Enum	Controls how to long lines should be wrapped. RT_WrapNone: No wrapping happens. RT_WrapWords: Whole words will be wrapped. If the word doesn't fit inside an empty line RT_WrapCharacters will be used. RT_WrapCharacters: Individual characters will be wrapped. If no character fits an empty line at least one will be drawn.	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	Ignored: Not tested because this property is inherited and not specific to this Widget.
TruncationMode	False	Enum	Text truncation method. Truncation is only active if the text size is bigger than 0. HardTruncation: Characters are cut off hard on text size boundaries. SoftTruncation: Only completely fitting characters are displayed. TextTruncation: None fitting characters are substituted with the truncation text.	

TruncationText	False	custom::String	Truncation text to be rendered.	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
VerticalAlignment	False	Enum	Vertical alignment of the text. RT_Leading: Aligns text to the top side for TTB and bottom side for BTT texts. RT_Middle: Aligned to the center. RT_Trailing: Aligns text to the bottom side for TTB and top side for BTT texts.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget property is deprecated and has been replaced in RichTextWidget.
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 98.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	---------------------	--------------------	----------------	---------------------	-------------------

## Chapter 99

# TextBaseLineOffsetWidget2D

**Name:** TextBaseLineOffsetWidget2D

**Description:** Attach this widget to a text RenderNode in order to control the BaseLineOffset property of the parent Group depending on the number of text lines which are rendered. The height of the text box is compared to the font line height to determine if the text is single line or multiple lines.

**Category:** Text

### 99.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: No test needed for this property because this prop is inherited but not specified for this widget.
FontLineHeight	True	Short	DEPRECATED	Ignored: No test needed for this property because this property is deprecated.

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: No test needed for this property because this property has not been fully developed.
MultipleLinesOffset	True	Float	Offset to be used when the text is rendered on multiple lines.	TC_W21_01
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
SingleLineOffset	True	Float	Offset to be used when the text is rendered on a single line.	TC_W21_01
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: No test needed for this property because this prop is inherited but not specified for this widget.
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 99.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 100

# TextColorWidget2D

**Name:** TextColorWidget2D

**Description:** Changes the color of a text depending on the enabled and active properties. This extension widget uses a text widget in the same scene on which it will set the text color depending on the Enabled property of the text widget and Active property of this widget. The text widget should be linked to a node which is ancestor of the this node widget. A particular situation would be when the text widget and the text color widget are linked to the same node.

**Category:** Text

### 100.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Active	True	Bool	Indicates that the text is active.	TC_W08_02
ActiveTextColor	False	Color	Text color used when the enabled and active.	
DisabledTextColor	False	Color	Text color used when the state is disabled (active info will be ignored).	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W08_01
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget

Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
NormalTextColor	False	Color	Text color used when the state is enabled and not active.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W08_03
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 100.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>



## Chapter 101

# TextExtensionWidget2D

**Name:** TextExtensionWidget2D

**Description:** TextExtensionWidget2D is a widget extension, that is capable to set special text configurations(like 'printf' style or animated strings) to a standard text widget. This extension widget uses the TextWidget on same node on which it will set the text depending on the Method property(TEXTCONCAT, TEXTSWITCH, TEXTWITHARGS). The TextWidget should be linked to a node which is ancestor of the this node widget, in most cases the TextWidget and the TextExtensionWidget are linked to the same node. A simple sample for the ARGS feature could be: 'Hello \$(2:) and \$(1:)?'

**Category:** Text

### 101.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AnimationTime	False	UShort	Speed for switching between the text parts	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing
ExtensionMethod	False	Enum	ExtensionMethod	
FormatText	True	custom://String	String with containing format identifier for insertion. e.g.	TC_W89_02
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This property has not been fully developed
Name	False	CharArray	The name of the widget instance	

Node	False	Node2D	The associated node of the widget.	
Text1	True	custom://String	String1 to be displayed on the label	TC_W89_01
Text2	True	custom://String	String2 to be displayed on the label	TC_W89_01
Text3	True	custom://String	String3 to be displayed on the label	TC_W89_01
Text4	True	custom://String	String4 to be displayed on the label	TC_W89_01
Text5	True	custom://String	String5 to be displayed on the label	TC_W89_01
TextIdsList	False	custom://TextIdList	TextIdList ids used by this widget	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W89_09
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 101.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 102

# TextHighlightWidget2D

**Name:** TextHighlightWidget2D

**Description:** UNDER DEVELOPMENT! Don't use this widget yet. Highlights segments from a text using the specified color. Attach this widget to a text render node.

**Category:** Under construction

### 102.1 Property list

Name	Bindable	Type	Description	Test Scope
Color	True	Color	Color used for text highlight.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Mode	False	Enum	Mode : enHighlightMode::Enum	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
Text	True	custom://string	String figures text to be highlight.	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

102.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 103

# TextWaitAnimationWidget2D

**Name:** TextWaitAnimationWidget2D

**Description:** TextWaitAnimationWidget2D is used set the mask node position to play the animation using animation widget, TextWaitAnimationWidget2D and animation widget are associated to the mask node. It uses the text widget which should be associated with GLBitmapBrushMaskBlend.

**Category:** Text

### 103.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AnimationText	False	custom://String	String to be added to animate	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
TextWidget	False	Widget	Textwidget which should be animated	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

103.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 104

# TextWidget2D

**Name:** TextWidget2D

**Description:** Label widget capable for rendering text in 2D scenes, single line and multiline. TextWidget2DV2 is bindable to a RenderNode or TextNode2D. In case of TextNode2D most of the properties are provided by the Node, otherwise use the properties of the widget.

**Category:** Text

### 104.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Alpha	True	Float	Represents the Alpha value to be set to the associated node, default: 1.0	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
Bold	True	Bool	This will enable Bold Text	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.

BracesCorrection	False	Bool	This property is used to inform text widget to process the given string to find the braces and add a special unicode to handle them correctly in RTL culture. Default is false and shall be enabled ONLY based on use case. Even if it is set to true, the processing happens only in RTL cultures.	
CultureDependentAlignment	False	Bool	If enabled then HLeft and HRight will be inverted for cultures with right to left text direction	
Direction	True	Enum	Utilized for text flow direction. Legacy : old behavior, Implicit: depends on character, LeftToRight: As LTR, RightToLeft: As RTL, Culture: As culture direction, Node: As Layout Direction	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
FixedTextBoxSize	True	Bool	If set to True, Text Box Size will be equal to maximum size else Text Box size should be based on given text	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
HorizontalAlignment	True	Enum	Text horizontal alignment	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.



Italic	True	Bool	This will enable Italic Text	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
LineSpacingFactor	True	Float	LineSpacing used e.g. buttons with two lines, a value of one is the font height	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
LineSpacingInPixels	True	Float	LineSpacing between the lines to be used in pixels	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
MaximumNumberOfLines	True	UShort	Maximum Number Of Lines, default is one.	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
MaximumSize	True	Vector2	Window size for truncation and multiline mode.	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
MultiLineHorizontalAlignment	False	Enum	Text horizontal alignment in case text is rendered for multiple lines	
MultiLineLayouting	True	Bool	Sets the property of MultiLineLayouting for the TextBrush	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

OutlineColor	True	Color	Sets the outline color	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
OutlineWidth	True	Byte	Sets the outline width	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
ScrollBehavior	False	Enum	Behavior of scrolling, circular endless or scrolling to the end of line and return after a short remaining time.	
ScrollCharCount	False	UInt	Pixels scrolled is equal to (Number of characters * default font width), 0 indicates legacy behavior	
ScrollSpeed	False	UShort	Speed for scrolling in msec. 300 is a good value and default.	
ScrollStartDelay	False	UShort	Delay Required before starting scrolling in msec	
ScrollStartMode	False	Enum	How scrolling feature was enabled : 'OnTouch' will start the scrolling whenever someone touches the widget, 'Auto' starts the scrolling feature after init of the scene. 'Extern' starts scrolling after it was called from an ancestor widget.	
ShrinkMethod	False	Enum	This will enable to choose the extended textstyle condensed and shrinked for shrink feature	
Style	True	TextStyle	The text style for the label with size and font information	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
Text	True	custom://string	String text, that has to be displayed. Language dependent string has to be taken from the translation table, use the choice button on the right side to select a string	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.

TextColor	True	Color	Color to be used to display the text; For state handling see als TextColorWidget2D.	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
TextIdsList	False	custom://TextIdList	Text IDs used by this widget	
TransRefs	False	custom://TransRefs	String separated TEXTID reference list, only used for translation topics. As of now you have to add manually the ID's devided with a comma.	
TruncationDirection	True	Enum	Truncation Direction add right side or left side of the text	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
TruncationMethod	False	Enum	Text truncation method; Hard cuts pixelwise, Soft cuts charwise, Text adds the 'TruncationText' ... and Shrink fits the string to the available space.	
TruncationText	False	custom://TruncationText	Truncation text to be rendered in UTF-8 encoding, in most cases '...'	
VerticalAlignment	True	Enum	Text vertical alignment	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

WordWrap	True	Bool	Sets the property of WordWrap for the TextBrush	Ignored: This widget property is deprecated and has been replaced in TextWidget2DV2.
----------	------	------	-------------------------------------------------	-----------------------------------------------------------------------------------------

104.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 105

# TextWidget2DV2

**Name:** TextWidget2DV2

**Description:** Label widget capable for rendering text in 2D scenes, single line and multiline. TextWidget2DV2 is bindable to a RenderNode or TextNode2D. In case of TextNode2D most of the properties are provided by the Node, otherwise use the properties of the widget.

**Category:** Under construction

### 105.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Alpha	True	Float	Represents the Alpha value to be set to the associated node, default: 1.0	TC_W18_03
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
Bold	True	Bool	This will enable Bold Text	TC_W18_06
BracesCorrection	False	Bool	This property is used to inform text widget to process the given string to find the braces and add a special unicode to handle them correctly in RTL culture. Default is false and shall be enabled ONLY based on use case. Even if it is set to true, the processing happens only in RTL cultures.	
CultureDependentAlignment	False	Bool	If enabled then HLeft and HRight will be inverted for cultures with right to left text direction	

Direction	True	Enum	Utilized for text flow direction.Legacy : old behavior, Implicit: depends on character, LeftToRight: As LTR, RightToLeft: As RTL, Culture: As culture direction, Node: As Layout Direction	TC_W18_15
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	TC_W18_04
FixedTextBoxSize	True	Bool	If set to True, Text Box Size will be equal to maximum size else Text Box size should be based on given text	
HorizontalAlignment	True	Enum	Text horizontal alignment	TC_W18_18
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget
Italic	True	Bool	This will enable Italic Text	TC_W18_07
LineSpacingFactor	True	Float	LineSpacing used e.g. buttons with two lines, a value of one is the font height	TC_W18_13
LineSpacingInPixels	True	Float	LineSpacing between the lines to be used in pixels	TC_W18_14
MaximumNumberOfLines	True	UShort	Maximum Number Of Lines, default is one.	TC_W18_10
MaximumSize	True	Vector2	Window size for truncation and multiline mode.	TC_W18_09
MultiLineHorizontalAlignment	False	Enum	Text horizontal alignment in case text is rendered for multiple lines	
MultiLineLayouting	True	Bool	Sets the property of MultiLineLayouting for the TextBrush	TC_W18_12
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
OutlineColor	True	Color	Sets the outline color	TC_W18_08
OutlineWidth	True	Byte	Sets the outline width	TC_W18_08
ScrollBehavior	False	Enum	Behavior of scrolling, circular endless or scrolling to the end of line and return after a short remaining time.	TC_W18_19
ScrollCharCount	False	UInt	Pixels scrolled is equal to (Number of characters * default font width), 0 indicates legacy behavior	
ScrollSpeed	False	UShort	Speed for scrolling in msec. 300 is a good value and default.	
ScrollStartDelay	False	UShort	Delay Required before starting scrolling in msec	

ScrollStartMode	False	Enum	How scrolling feature was enabled : 'OnTouch' will start the scrolling whenever someone touches the widget, 'Auto' starts the scrolling feature after init of the scene. 'Extern' starts scrolling after it was called from an ancestor widget.	
ShrinkMethod	False	Enum	This will enable to choose the extended textstyle condensed and shrinked for shrink feature	
Style	True	TextStyle	The text style for the label with size and font information	
Text	True	custom://String	String text, that has to be displayed. Language dependent string has to be taken from the translation table, use the choice button on the right side to select a string	TC_W18_01
TextColor	True	Color	Color to be used to display the text; For state handling see als TextColorWidget2D.	TC_W18_02
TextIdsList	False	custom://TextIdList	TextIdList ids used by this widget	
TransRefs	False	custom://String	Stringseparated TEXTID reference list, only used for translation topics. As of now you have to add manually the ID's devided with a comma.	
TruncationDirection	True	Enum	Truncation Direction add right side or left side of the text	TC_W18_16
TruncationMethod	False	Enum	Text truncation method; Hard cuts pixelwise, Soft cuts charwise, Text adds the 'TruncationText' ... and Shrink fits the string to the available space.	
TruncationText	False	custom://String	Truncation text to be rendered in UTF-8 encoding, in most cases '...'	
VerticalAlignment	True	Enum	Text vertical alignment	TC_W18_17
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	TC_W18_05
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
WordWrap	True	Bool	Sets the property of WordWrap for the TextBrush	TC_W18_11

## 105.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------



## Chapter 106

# TextureImageWidget3D

**Name:** TextureImageWidget3D

**Description:** Controls the Texture property of the existing bitmap brush of a render node.

**Category:** Image

### 106.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Bitmap	True	Bitmap	Bitmap to be set on the BitmapBrushEffect inside the RenderNode.	
Enabled	False	Bool	Enabled: Enable or disable the widget	
Name	False	CharArray	The name of the widget instance	
Node	False	Node3D	The associated node of the widget.	

### 106.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>

## Chapter 107

# TimerWidget2D

**Name:** TimerWidget2D

**Description:** TimerWidget2D widget

**Category:** Common

### 107.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AutoRestartEnabled	False	Bool	When this property is Enabled, the timer will run recursively until deactivation of the scene.	
AutoStartEnabled	False	Bool	When this is Enabled, the timer will be started automatically on the render activation of the scene.	
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	

PauseTrigger	True	Bool	This is an explicit trigger to pause the timer. When set to true a running timer(if any) is paused. When set to false a paused timer(if any) continues to run.	
StartStopTrigger	True	Bool	This is an explicit trigger to start or stop the timer. When set to false the timer is stopped, always. On setting the StartStopTrigger property to true it causes a restart of the timer.	
TimerDuration	False	UInt	This property holds the timeout value in milli seconds.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 107.2 Message list

<i>Name</i>	<i>Subscription</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
TimerWidgetExpiryMsg	Model, View, Controller		ElapsedTime: Courier::UInt32.  Stopped: bool. +	sequential	

## Chapter 108

# ToggleWidget2D

**Name:** ToggleWidget2D

**Description:** Controls the RenderingEnabled property of a node (allowing to toggle a node by a bindable widget property).

**Category:** Common

### 108.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
Enabled	True	Bool	DEPRECATED. Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. The same node property is configured also by the widget property Visible which is available in all widgets. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
UseDeprecatedEnabled	False	Bool	Enables the usage of the deprecated property Enabled. The default value of this property is true for legacy reasons (notice that also default VisibleEnabled is true for the same reasons). It is recommended to set this property to false and to control the node visibility by using the Visible property which is available in all widgets.	
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 108.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
-------------	--------------------	--------------------	----------------	---------------------	-------------------

## Chapter 109

# TwoDimensionSliderWidget2D

**Name:** TwoDimensionSliderWidget2D

**Description:** TwoDimensionSliderWidget2D is used to get the touched position values based on the horizontal and vertical values set by the user. Add a group node, add three bitmap node for the crosshair, first for the knob, 2nd for horizontal line and 3rd for vertical line.

**Category:** Range

### 109.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AppearanceId	True	UInt	UNDER DEVELOPMENT! Specifies the id of the appearance responsible to change, based on widget state (enabled, pressed, active, focused), the images and the colors for the widget node and the descendant nodes. The appearances are registered at start-up.	Ignored: Under development
ControllerId	True	Short	Identifies the controller attached to this widget. -1 no controller is attached; 0 default controller for the class is attached (used for derived classes); 1..n id of a controller registered at start-up;	Ignored: Id of a controller registered at start-up. It cant be changed during run time
DisabledTouching	True	Bool	Widget can be touched also when it is disabled.	
DoubleTap	True	Bool	Enable double tap gesture detection	Ignored: The TTFis simulation for the gesture action is not ready yet.

Drag	True	Bool	Enable drag gesture detection	
DragDirection	False	Enum	Direction in which drag should be detected	
DragDropDestinationEnabled	True	Bool	Enables the widget to be used as the target of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
DragDropSourceEnabled	True	Bool	Enables the widget to be used as the source of a drag and drop operation.	Ignored: The TTFis simulation for the gesture action is not ready yet.
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	
FocusControllerSet	True	Short	The application can associate a list of focus controllers (ControllerSet) to a numerical id. Those controllers can be used for a widget based on the same id.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusOrder	True	Short	Focus order. Zero has the highest priority.	Ignored: The TTFis simulation for the gesture action is not ready yet.
FocusParentNode	False	Node2D	Node of the parent focus group. If it is not specified a search will be performed to find a focus group linked to the closest ancestor node.	
Focusable	True	Bool	Widget can gain the focus.	Ignored: The TTFis simulation for the gesture action is not ready yet.

GestureConfigId	True	UInt	Identifies the gesture configuration used for this widget. Gesture configurations are registered at start-up and attached to widgets using numerical ids (0 is used for the default configuration defined in the widget). For more information please read the gesture configuration chapter in the widget user guide.	
HorizontalCurValue	True	Int	Current horizontal value of marker based on the range (int).	
HorizontalMovementLocked	True	Bool	Only vertical movement allowed, e.g for Audio Fader	
HorizontalPermittedValueMax	True	Int	Permitted range value, allowed valid area inside given range (int).	
HorizontalPermittedValueMin	True	Int	Permitted range value, allowed valid area inside given range (int).	
HorizontalRangeMax	False	Int	Max scale value for horizontal marker (int).	
HorizontalRangeMin	True	Int	Min scale value for horizontal marker (int).	
HorizontalStepValue	False	UInt	Horizontal marker movement step value given in unsigned integer for move messages, default=1	
InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
PinchSpread	True	Bool	Enable pinch and spread gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
PressHold	True	Bool	Enable hold gesture detection	
PressRepeat	True	Bool	Enable repeat gesture detection	
RawTouch	True	Bool	Enable raw touch coordinate routing (mainly for hand writing recognition)	Ignored: Not tested because this property is inherited and not specific to this Widget.



Rotate	True	Bool	Enable rotate gesture detection	Ignored: Not tested because this property is inherited and not specific to this Widget.
SkinNode	False	Node2D	The node of the background skin.	
SliderArea	False	Rectangle	Configures the touchable area of the slider. X and Y are coordinates relative to the upper left corner of the node effective bounding rectangle. If width or height are zero or negative the effective bounding rectangle of the node will be used as touchable area (X and Y will also be ignored).	
Swipe	True	Bool	Enable swipe gesture detection	
SwipeDirection	False	Enum	Direction in which swipe should be detected	
Tap	True	Bool	Enable press and tap gesture detection	
TouchPriority	True	UInt	Increase this priority to handle touch message for this widget before widgets with a lower priority	Ignored: Not tested because this property is inherited and not specific to this Widget.
Touchable	True	Bool	Widget is Touchable or not	
UsePermittedValues	True	Bool	Make use of permitted values inside of given range values	
UserData	True	UInt	Together with the view and the widget identifier, this user data is a parameter to many messages posted by the widgets which can be used in the state machine or in the data model. Use data binding to change this value dynamically and store extra information in the widgets.	Ignored: The infrastructure is in the base widget, No extra information to store here.
ValueUpdateOnDrag	False	Bool	Set to true if value updates are required while dragging, set to false if update is required only at the end of the drag operation, .	
VerticalCurValue	True	Int	Current vertical value of marker based on the range (int).	
VerticalMovementLocked	True	Bool	Only vhorizontal movement allowed, e.g for Audio Fader	
VerticalPermittedValueMax	True	Int	Permitted range value, allowed valid area inside given range (int).	
VerticalPermittedValueMin	True	Int	Permitted range value, allowed valid area inside given range (int).	
VerticalRangeMax	False	Int	Max scale value for vertical marker (int).	
VerticalRangeMin	True	Int	Min scale value for vertical marker (int).	
VerticalStepValue	False	UInt	Vertical marker movement step value given in unsigned integer for move messages, default=1	

Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 109.2 Message list

<i>Name</i>	<i>Subscriber</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>
MoveMsg	View		WidgetName: Can-dera::String.  MoveType: Wid-gets::TwoDimensionSlider::MoveType.  Value: ::Courier::Int32. +		
UpdMsg	Model		WidgetName: Can-dera::String.  ValueHor: Courier::Int32.  ValueVer: Courier::Int32.  UpdType: Wid-gets::TwoDimensionSlider::UpdType. +		

## Chapter 110

# ViewSwitchWidget2D

**Name:** ViewSwitchWidget2D

**Description:** Manages the load/unload and activate/deactivate of some views which are considered as children of the widget's parent view (this scene).

**Category:** Common

### 110.1 Property list

<i>Name</i>	<i>Bindable</i>	<i>Type</i>	<i>Description</i>	<i>Test Scope</i>
AddInvalidationDependency	False	Bool	If set to true then two-way invalidation dependency will be set between the widget's parent view (this scene) and the selected view.	
CacheViews	True	Bool	If set to true then the unselected views will be deactivated else they will be unloaded (destroyed).	TC_W95_04
Enable	True	Bool	Enables this widget. Some widgets (for example button) use this property while others ignore it. If InheritEnabled is true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true.	Ignored: This property has not been supported for testing.
Index	True	UInt	Index of the view which should be visible.	TC_W95_01

InheritEnabled	True	Bool	If true then this widget is considered effectively enabled only if both local Enabled and the value inherited from the ancestor EnableGroupWidget2D are true. If this widget has no EnableGroupWidget2D ancestor or InheritEnabled is false then only the local Enabled is used.	Ignored: Not tested because this property is inherited and will be tested with Enabled-GroupWidget.
Name	False	CharArray	The name of the widget instance	
Node	False	Node2D	The associated node of the widget.	
ViewIds	True	custom://Name	Names of the views managed by this widget. The view full name will be obtained by concatenating ViewPrefix and ViewId[Index].	TC_W95_03
ViewPrefix	True	custom://Prefix	Prefix to be inserted before view id.	TC_W95_02
Visible	True	Bool	Configures the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	Ignored: Not tested because this property is inherited and not specific to this Widget.
VisibleEnabled	False	Bool	Enables the configuration of the node property EnableRendering which is used to determine if the node is rendered or not. A node is effectively rendered if it and all its ancestors have rendering enabled. Please notice that if the same property of a node is set from multiple sources then the result is unpredictable.	

## 110.2 Message list

<i>Name</i>	<i>Subscribers</i>	<i>Description</i>	<i>Members</i>	<i>Distribution</i>	<i>Test Scope</i>