

# How to Use CAPH WUI Widgets

Published 2014-10-27 | (Compatible with SDK 5.0,5.1 and 2013,2014 models)

This file will give some descriptions for using caph.wui widget.

Contents

[Prerequisites](#)

[Environment](#)

[Source File](#)

[How to Develop](#)

[Open Issues](#)

[Related Documents](#)

## Prerequisites

To create using caph.wui widget application, take highlight widget for example, you should include caph.wui dependencies, basic widget of UI, the base style of UI, The CSS style of caph.wui, highlight app source files in your application by putting the following code in the <head> and <body> sections of highlight.html, just as follows:

```
<head>
  <title>test_highlight</title>

  <!-- load caph-level1-unified.min.js, caph.css -->
  <script src="$CAPH/1.0.0/caph-level1-unified.min.js"></script>
  <link href="$CAPH/1.0.0/caph.css" rel="stylesheet" type="text/css">

  <!-- load the app -->
  <script src="highlight.js"></script>

</head>
<body onload="app.run();">
</body>
```

Source files mentioned above are explained in the table.

File	Description
caph-level1-unified.min.js	It is include all kinds of is basic widget of UI will be used. It includes external libraries also.
caph.css	The class style of caph.wui.
highlight.js	The highlight application source file.

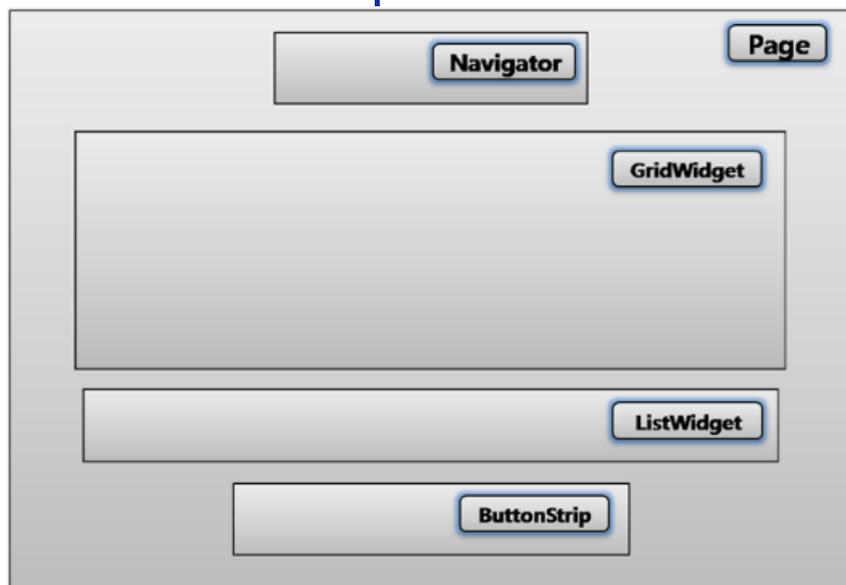
## Environment

Use Samsung Smart TV SDK and caph.wui to create applications that run on a TV screen, you need: Samsung TV connected to the Internet, SDK or a text editor for creating HTML, JavaScript and CSS files. You can also use the emulator provided with the SDK to debug and test the applications before uploading them in your TV.

# Source File

The files needed for the sample application are [here](#).

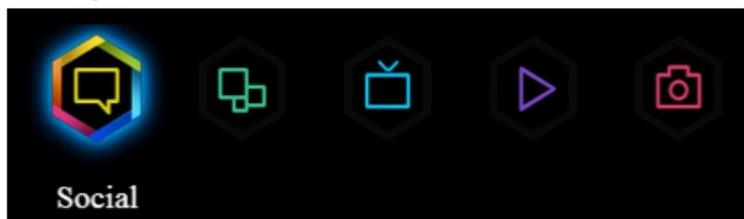
## How to Develop



In order to show the application with CAPH WUI Widgets on the screen, first it needs a thing to paint, it's a UIContext, which is the root canvas, in order to render CAPH WUI Widget objects on it, thus the objects can be displayed.

The file will introduce the highlight widget, it includes Navigator, GridWidget, ListWidget, Panel and Button strip widgets, so we will give some descriptions about these widgets.

Navigator is a sample widget with a layout in it. Each layout block has an image and a label. When the image is clicked, it will navigate to some other UIContext.



To initialize a Navigator widget, use this code:

```
var UIContext = caph.wui.widget.UIContext;
var Navigator = caph.wui.widget.Navigator;
var uiContext = new UIContext();
var gator = new Navigator();
```

To set the main options of Navigator object, including width, height, CSS of label of Navigator item, maximum size and cycle, using the following codes:

```
var Options = {
  'frame' : {
    'width' : 450,
    'height' : 90,
    'labelCls' : 'bb',
    'maxsize' : 5,
    'cycle' : false,
  }
};
```

The following codes set the position of Navigator showing on the screen:

```

var position = {
    'x' : 1000,
    'y' : 100,
    'z' : 1
};

```

```

gator.setPosition(position.x, position.y, position.z);

```

Add one item to the Navigator object as follows, you can add other items like this, it defines CSS styles of 4 kinds of states and the text content on the label:

```

var state0 = {
    'normal' : './img/social_normal.png',
    'focus' : './img/social_focus2.png',
    'select' : './img/social_selected.png',
    'hover' : './img/social_over.png',
    'text' : 'Social'
};

```

```

gator.addItem(0, state0);

```

How can user draw the Navigator widget on the UIContext of screen is like:

```

gator.render(uiContext);
uiContext.show();

```

GridWidget is a widget group that displays widgets in a two-dimensional, you can add some caph.wui objects on it.



To initialize a GridWidget widget, using these codes:

```

var UIContext = caph.wui.widget.UIContext;
var GridWidget = caph.wui.widget.GridWidget;
var uiContext = new UIContext();
var gridwidget = new GridWidget();

```

To set the main options of GridWidget object, including width, height, layout, cycle, the width and height of each item and margin parameters and so on, using the following codes:

```

var grid_options = {
  'frame': {
    'width': 1000,
    'height': 900,
    'layout': {
      'row': 4,
      'column': 4
    },
    'cycle': false,
  },
  'item': {
    'item_width': 223,
    'item_height': 198,
    'margin': '10 10'
  }
};

```

Take adding one panel item to the GridWidget object for example as follows, you can add other items like this, Import Panel source codes, new the Panel object, set the image and text content to Panel object, add the Panel object to GridWidget widget.

```

var Panel = caph.wui.widget.Panel;
var panel1 = new Panel(options);

panel1.setURL('img/2.jpg');
panel1.setText('Panel1');
gridwidget.addItem(panel1);

```

How can user draw the ``GridWidget`` widget on the ``UIContext`` of screen is like:

```

gridwidget.render(uiContext);
uiContext.show();

```

ListWidget is a widget list group that displays widgets in one dimensional, it's a special example of GridWidget, you can add some caph.wui widgets on it.



To initialize a ListWidget widget, using these codes:

```

var UIContext = caph.wui.widget.UIContext;
var ListWidget = caph.wui.widget.ListWidget;
var uiContext = new UIContext();
var listW = new ListWidget();

```

To set the main options of ListWidget object, including width, height, cycle, maximum size, default it can focus, the width, height, margin left, margin right, margin top and margin bottom of each item parameters and so on, using the following codes:

```

var options = {
  'frame': {
    'width': 3000,
    'height': 400,
    'cycle': false,
  },
  'maxSize': 10,
  'disableFocus': false,
  'item': {
    'item_width': 100,
    'item_height': 200,
    'margin_left': 0,
    'margin_right': 0,
    'margin_top': 0,
    'margin_bottom' : 0,
  }
};

```

The following code sets the position of ListWidget showing on the screen:

```

var position = {
  'x': 1500,
  'y': 400,
  'z': 0
};

```

```
listW.setPosition(position.x, position.y, position.z);
```

Take adding one panel item to the ListWidget object for example as follows, you can add other items like this, Import Panel source codes, define the options about width, height, textHeight, margin and labelScroll (whether the text on Panel label can scroll), new the Panel object, set the image and text content to Panel object, add the Panel object to ListWidget widget.

```

var Panel = caph.wui.widget.Panel;
var optionsB = {
  'width': 285,
  'height': 385,
  text: {
    'textHeight': 45,
    'margin':'25 0'
  },
  'labelScroll': false
};
var panel0 = new Panel(optionsB);

```

```

panel0.setURL('images/2.png');
panel0.setText('NormalType');
listW.addItem(0, panel0);

```

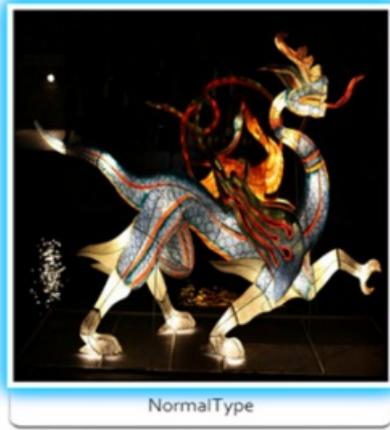
How can user draw the ListWidget widget on the UIContext of screen is like:

```

listW.render(uiContext);
uiContext.show();

```

Panel is a widget that consists with Image UI and Label UI. When you click it, the image can zoom in.



To initialize a Panel widget, using these codes:

```
var UIContext = caph.wui.widget.UIContext;
var Panel = caph.wui.widget.Panel;
var uiContext = new UIContext();
var panel0 = new Panel();
```

To set the main options of Panel object, including width, height, embedded style, text content, text height, text scroll, image, focusScaleX and focusScaleY properties, using the following codes:

```
var option1 = {
  frame: {
    'width': 400,
    'height': 450,
    'style': 'normal'
  },
  text: {
    'data': 'NormalType',
    'text-height': 50,
    'text-scroll': true,
    'margin': '5% 50%'
  },
  image: {
    'url': 'img/1.png'
  },
  focusScaleX : 1.05,
  focusScaleY : 1.05
};
```

How can user draw the Panel widget on the UIContext of screen is like:

```
panel0.render(uiContext);
uiContext.show();
```

Button strip is a widget, with a layout in it. Each layout block has a label. When the label is choose, it will highlight.



To initialize a Button strip widget, using these codes:

```
var UIContext = caph.wui.widget.UIContext;
var Strip = caph.wui.widget.Strip;
var uiContext = new UIContext();
var strip = new Strip();
```

To set the main options of Button strip object, including width, height, cycle, focusHighlightCLs, focusCIsTargetWidth, focusCIsTargetHeight and CSS style of Button strip parameters, using the following codes:

```

var strip_options = {
  'frame': {
    'width': 900,
    'height': 100,
    'cycle': false,
    'focusHighlightCLs': 'hh',
    'focusClsTargetWidth': 150,
    'focusClsTargetHeight': 60
  },
  cls: 'strip'
};

```

Take adding one label item to the Button strip object for example as follows, adding CSS style to Button strip Object, you can add other items like this:

```

strip.addItem(0,'What');
strip.addCls('bb');

```

How can user draw the Panel widget on the UIContext of screen is like:

```

strip.render(uiContext);
uiContext.show();

```

The following parts will be in order to show how to use caph.wui.widget, so first we will import the source files about these widgets we used about UIContext, GridWidget, Image and so on, at the beginning of the highlight.js file, just as follows:

```

1      /**
2       * buttonstrip sample
3       */
4      var app = app || {};
5
6      app.run = function() {
7          caph._setBasePath('$CAPH');
8          app.handleLoad();
9
10         var UIContext = caph.wui.widget.UIContext;
11         var GridWidget = caph.wui.widget.GridWidget;
12         var Image = caph.wui.widget.Image;
13         var HighLight = caph.wui.widget.HighlightHelper;
14         var Panel = caph.wui.widget.Panel;
15         var ListWidget = caph.wui.widget.ListWidget;
16         var Navigator = caph.wui.widget.Navigator;
17         var Strip = caph.wui.widget.Strip;

```

The codes explain as follows:

Line 10	Import the UIContext` widget will be used. The ``UIContext widget is the root canvas, in order to render object on it.
Line 11	Import the GridWidget widget will be used. The GridWidget object is widget group that displays widgets in a two-dimensional.
Line 12	Import the Image widget will be used. Image is the object that contains the image.
Line 13	Import the highlight widget will be used. The HighlightHelper is a selector. Its alternative terms could be cursor or focus that can be generally controlled by user's remote control. It moves over widgets or UI templates. It always knows where it should move when any input events like up, down, left, and right happen.
Line 14	Import the Panel widget will be used. The Panel object is a widget that consists with Image UI and Label UI.
Line 15	Import the ListWidget widget will be used. The object is widget list.
Line 16	Import the Navigator widget will be used. Navigator is a class with a layout in it. Each layout block has an image and a label. When the image is clicked, it will navigate to some other UIContext.

Line 17	Import the button strip widget will be used. Button Strip is a class, with a layout in it. Each layout block has a label. When the label is chosen, it will highlight.
------------	--

Second, it will define some options about width, height, labelCIs and so on parameters, and position parameters for these widgets about Navigator, GridWidget, ListWidget and button Strip, we should define before the special operations, the codes just as follows:

```

1      var navigator_options = {
2          width: 450,
3          height: 90,
4          labelCls: 'navigator-label',
5          focusCls: 'u-navigator-focus-d',
6          selectCls: 'u-navigator-select-d',
7          cls: 'u-navigator-normal-d'
8      };
9
10     var navigator_Position = {
11         'x': 950,
12         'y': 60,
13         'z': 1
14     };
15
16     var gridwidget_option = {
17         'width': 900,
18         'height': 550,
19         'cls': 'boxCls'
20     };
21
22     var gridw_position = {
23         'x': 950,
24         'y': 415,
25         'z': 1
26     };
27
28     var listwidget_options = {
29         'width': 900,
30         'height': 110,
31         'maxSize': 6,
32         'cycle': false,
33         'disableFocus': false
34     };
35
36     var list_Position = {
37         'x': 950,
38         'y': 775,
39         'z': 0
40     };
41
42     //strip
43     var strip_Position = {
44         'x': 950,
45         'y': 915,
46         'z': 5
47     };

```

The codes explain as follows:

Lines 1-8	Define the options of navigator, including width, height and all class styles, including focus, select and normal types.
-----------	--

Lines 10-14	Define the position of navigator, decide the location showing on the screen.
Lines 16-20	Define the options of GridWidget, including width, height, class style.
Lines 22-26	Define the position of GridWidget, decide the location showing on the screen.
Lines 28-34	Define the options of ListWidget, including width, height, maxSize, whether display cycle and default focus.
Lines 36-40	Define the position of ListWidget, decide the location showing on the screen.
Lines 43-47	Define the position of button strip, decide the location showing on the screen.

Third, rendering the navigator widget object to the UIContext object, it should new the UIContext and navigator object, define the options about width, height, cycle and so on parameters for navigator, add 5 items to navigator object, set the position for showing the navigator object and render it on the UIContext object.

```

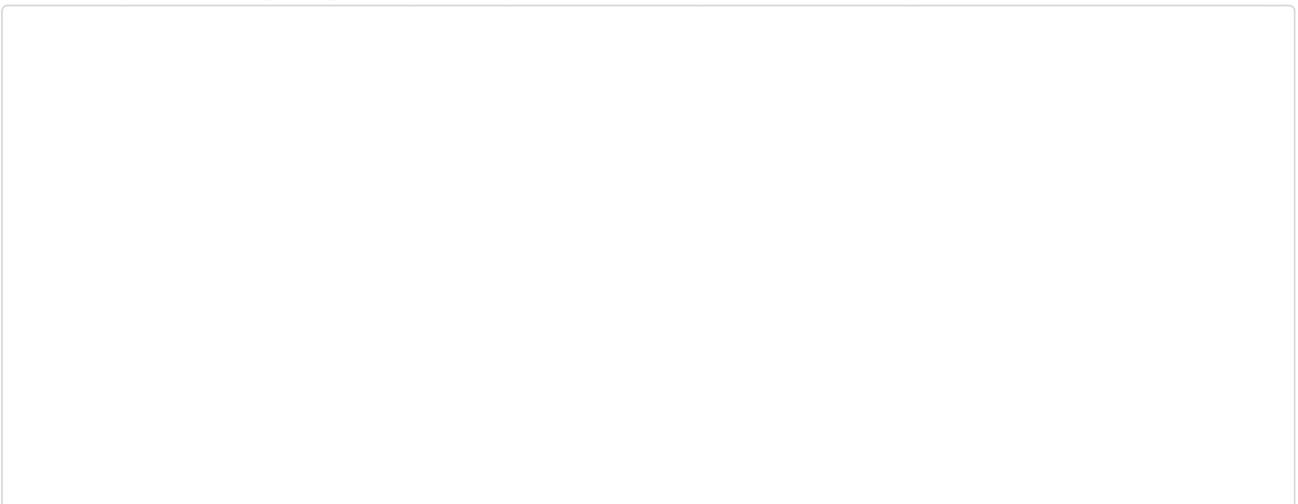
1  var uiContext = new UIContext();
2
3  var options0 = {
4      width: 450,
5      height: 90,
6      cycle: true,
7      labelCls: 'bb',
8      focusCls: '',
9      focusClsTargetWidth: 90,
10     focusClsTargetHeight: 90
11 };
12
13 var gator = new Navigator(options0);
14
15 gator.addItem(0, './images/social_normal.png', 'Social', state0);
16 gator.addItem(1, './images/apps_normal.png', 'Apps', state1);
17 gator.addItem(2, './images/smartguide_normal.png', 'On TV', state2);
18 gator.addItem(3, './images/movies_normal.png', 'Movie & TV Shows', state3);
19 gator.addItem(4, './images/photo_normal.png', 'Photos, Videos & Music', state4);
20
21 gator.setPosition(navigator_Position.x, navigator_Position.y, navigator_Position.z);
22 gator.render(uiContext);

```

The codes explain as follows:

Line 1	New the UIContext object. UIContext is a canvas, all the caph.wui widgets should be rendered on it.
Line 13	New the navigator object. navigator is a class with a layout in it. Each layout block has an image and a label. When the image is clicked, it will navigator to some other UIContext.
Lines 15-19	Add some items on the navigator widget, they has image, text content and state.
Line 21	Set the navigator object position. If you don't do this the navigator will display on the top left corner of your screen.
Line 22	Render navigator object to UIContext. Then the navigator object will be show on your screen.

Fourth, rendering the GridWidget widget to the UIContext, we should new the GridWidget object, set the layout style, add some item objects including image and panel objects to it, set the position for showing and render on the UIContext object.



```

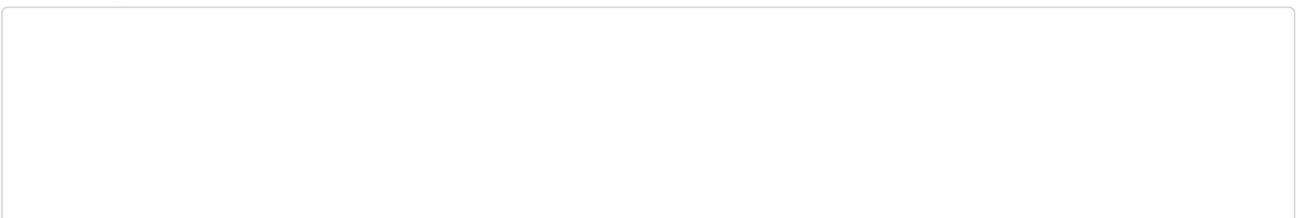
1  var gridwidget = new GridWidget(gridwidget_option);
2  gridwidget.setPosition(gridw_position.x, gridw_position.y, gridw_position.z);
3
4  var itemObjList = [];
5  var image;
6  for (var i = 0; i < imgoptions.length; i++) {
7      image = new Image(imgoptions[i]);
8      itemObjList.push(image);
9      image.addCls('caph-nav-item');
10 }
11
12 var layout = {
13     'row': 3,
14     'column': 4
15 };
16
17 gridwidget.setLayOut(layout);
18 gridwidget.addItemList(itemObjList);
19
20 var panel = new Panel(imgoptions[0]);
21 panel.setURL('images/10.jpg');
22 panel.setText('Testa');
23 gridwidget.addItem(panel, 9);
24 gridwidget.render(uiContext);

```

The codes explain as follows:

Line 1	New the GridWidget object. The object is widget group that displays widgets in a two-dimensional.
Line 2	Set the GridWidget object position. If you don't do this the GridWidget will display on the top left corner of your screen.
Lines 4-10	Add some image objects to item list. The images have added the class style.
Lines 12-15	Set the row and column layout style of GridWidget.
Lines 17-18	Set the layout of GridWidget object and add some items on it.
Lines 20-22	New the panel object. Add the image and text on the panel object.
Lines 23	Add panel objects on GridWidget widget.
Lines 24	Render GridWidget object to UIContext. Then the GridWidget object will be show on your screen.

Fifth, rendering the ListWidget and button strip widgets to the UIContext, we should new the ListWidget and button strip objects firstly, then add some item objects to them, set the position for showing and render on the UIContext object, add the highlight effect on UIContext and set the key operations to document body. Thus, we finish the introduce about how to use caph.wui widget.



```

1    var listwidget = new ListWidget(listwidget_options);
2    for (var i = 0; i < 6; i++) {
3        var imageObj = new Image(imgOption);
4        imageObj.addCls('caph-nav-item');
5        listwidget.addItem(i, imageObj);
6    }
7
8    listwidget.setPosition(list_Position.x, list_Position.y, list_Position.z);
9    listwidget.render(uiContext);
10
11   var strip = new Strip({
12       height: 50,
13       width: 600
14   });
15
16   strip.addItem(0, 'What');
17   strip.addItem(1, 'Who');
18   strip.addItem(2, 'Why');
19   strip.addItem(3, 'Where');
20   strip.addItem(4, 'How');
21   strip.render(uiContext);
22   strip.setPosition(strip_Position.x, strip_Position.y, strip_Position.z);
23
24   HighLight.init(uiContext);
25
26   uiContext.show();
27
28   caph.wui.widget.Keycontrol.init();

```

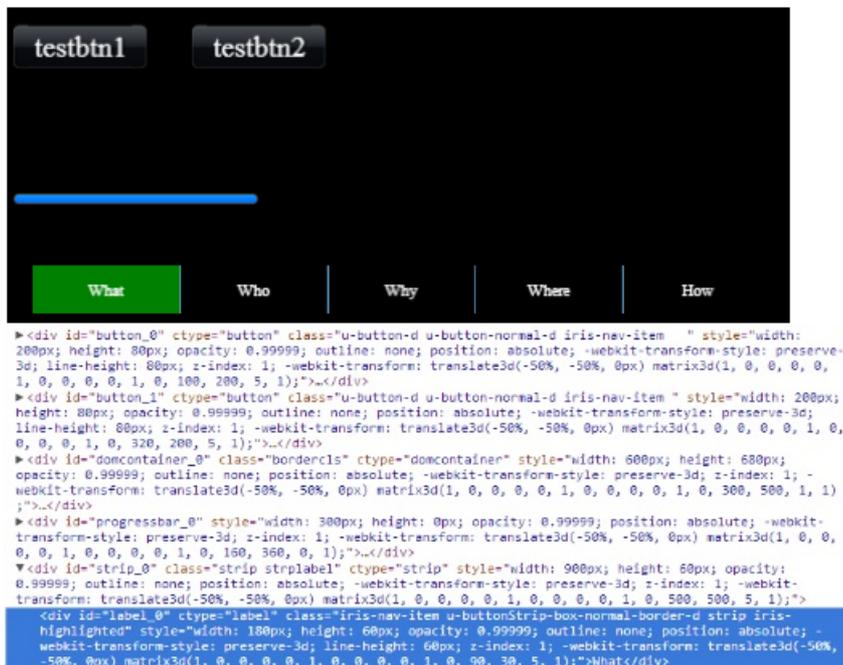
The codes explain as follows:

Lines 1-6	New the ListWidget object. The object is widget list. Add some image objects to item list. The images have added the class style.
Lines 8-9	Set the ListWidget object position. If you don't do this the ListWidget will display on the top left corner of your screen. And render ListWidget object to UIContext. Then the object will be show on your screen.
Lines 16-22	Add some items on the button strip widget, render button strip object to UIContext. Then the object will be show on your screen. Set the button strip object position. If you don't do this the button strip will display on the top left corner of your screen.
Line 24	Add the highlight effect on UIContext.
Line 26- 31s	Show the UIContext and define the key operations to document body.

The example picture is about above widgets shows on your screen is like, the second item of GridWidget has highlight effect:



When moving the highlight effect among CAPH WUI Widgets, you can set the `caph_nav_item` class style to the CAPH WUI Widgets, in order to find up, down, left and right position of widgets with the highlight effect to the current widget, if you set, it can easily find the exact widgets, if you don't set, even though the widget near the current widget in the up, down, left and right directions, it won't find the widgets. You can look at the following pictures.



Above two pictures, the two buttons and the items of button strip have the `caph_nav_item` class style, the progressbar widget hasn't this class style, when user wants to press the "up" key to move the highlight effect to progressbar, but the `button_0` has the highlight effect, this case could well explain this question, you can look at the following pictures.



```

<div id="button_0" ctype="button" class="u-button-d u-button-normal-d iris-nav-item u-button-normal-focus-d iris-highlighted" style="width: 200px; height: 80px; opacity: 0.99999; outline: none; position: absolute; -webkit-transform-style: preserve-3d; line-height: 80px; z-index: 1; -webkit-transform: translate3d(-50%, -50%, 0px) matrix3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 100, 200, 5, 1);"></div>
<div id="button_1" ctype="button" class="u-button-d u-button-normal-d iris-nav-item " style="width: 200px; height: 80px; opacity: 0.99999; outline: none; position: absolute; -webkit-transform-style: preserve-3d; line-height: 80px; z-index: 1; -webkit-transform: translate3d(-50%, -50%, 0px) matrix3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 320, 200, 5, 1);"></div>
<div id="domcontainer_0" class="bordercls" ctype="domcontainer" style="width: 600px; height: 680px; opacity: 0.99999; outline: none; position: absolute; -webkit-transform-style: preserve-3d; z-index: 1; -webkit-transform: translate3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 300, 500, 1, 1);"></div>
<div id="progressbar_0" style="width: 300px; height: 0px; opacity: 0.99999; position: absolute; -webkit-transform-style: preserve-3d; z-index: 1; -webkit-transform: translate3d(-50%, -50%, 0px) matrix3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 160, 360, 0, 1);"></div>
<div id="strip_0" class="strip strplabel" ctype="strip" style="width: 900px; height: 60px; opacity: 0.99999; outline: none; position: absolute; -webkit-transform-style: preserve-3d; z-index: 1; -webkit-transform: translate3d(-50%, -50%, 0px) matrix3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 500, 500, 5, 1);">
  <div id="label_0" ctype="label" class="iris-nav-item u-buttonStrip-box-normal-border-d strip" style="width: 180px; height: 60px; opacity: 0.99999; outline: none; position: absolute; -webkit-transform-style: preserve-3d; line-height: 60px; z-index: 1; -webkit-transform: translate3d(-50%, -50%, 0px) matrix3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 90, 30, 5, 1);">What</div>

```

There are some differences between setScale and setSize methods, when setScale method is called, the display width and height of widget will be changed, but the value of height and width properties will not be modified, but when setSize method is called, the value of height and width properties are changed. Besides, setScale method can modify the x, y, z coordinate for scale position of the widget, but setSize method can only modify the x, y coordinate of the widget. About setScale method you can look at the following pictures, the panel widget is in the normal state, the width and height are 180px and 200px, the focusScaleX and focusScaleY are all 1 by matrix3d(1,0,0,0,0,1,0,0,0,0,1,0,1200,360,1,1).



NormalType

```

<div id="panel_1" ctype="panel" class="iris-nav-item u-panel-label-normal-d" style="width: 180px; height: 200px; opacity: 0.99999; outline: none; position: absolute; -webkit-transform-style: preserve-3d; background-image: url(file:///E:/tmp_fengk/iris_rc_003/iris/samples/ui/amd/panel/img/2.png); background-size: 180px 170px; line-height: 360px; text-indent: 0%; z-index: 1; -webkit-transform: translate3d(-50%, -50%, 0px) matrix3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1200, 360, 1, 1); background-position: 0px 0px; background-repeat: no-repeat no-repeat;">PiercedType</div>

```

By the setScale method you can look at the following pictures, the panel widget is in the magnify state, the width and height are also 180px and 200px, the focusScaleX and focusScaleY are all 1.25 by matrix3d(1.25,0,0,0,0,1.25,0,0,0,0,1,0,1200,360,1,1).



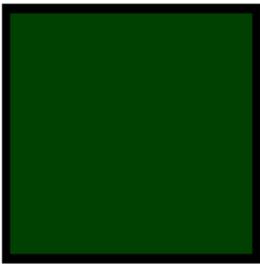
NormalType

```

<div id="panel_1" ctype="panel" class="iris-nav-item bb iris-highlighted" style="width: 180px; height: 200px; opacity: 0.99999; outline: none; position: absolute; -webkit-transform-style: preserve-3d; background-image: url(file:///E:/tmp_fengk/iris_rc_003/iris/samples/ui/amd/panel/img/2.png); background-size: 180px 170px; line-height: 360px; text-indent: 0%; z-index: 1; -webkit-transform: translate3d(-50%, -50%, 0px) matrix3d(1.25, 0, 0, 0, 0, 1.25, 0, 0, 0, 0, 1, 0, 1200, 360, 1, 1); background-position: 0px 0px; background-repeat: no-repeat no-repeat;">PiercedType</div>

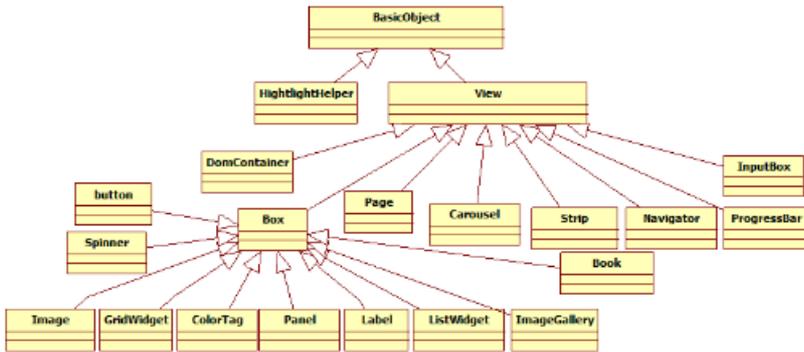
```

About setSize method, you can look at the following pictures, you can paint a box on the UIContext object with width and height are 300px and 300px, If user doesn't call setSize method, the width and height are 0px, and there won't see a box on the UIContext object.

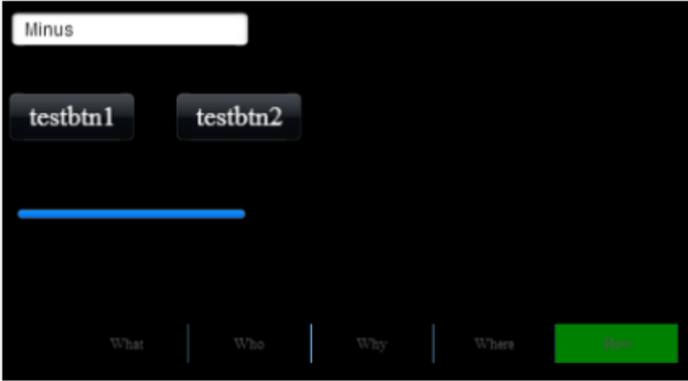


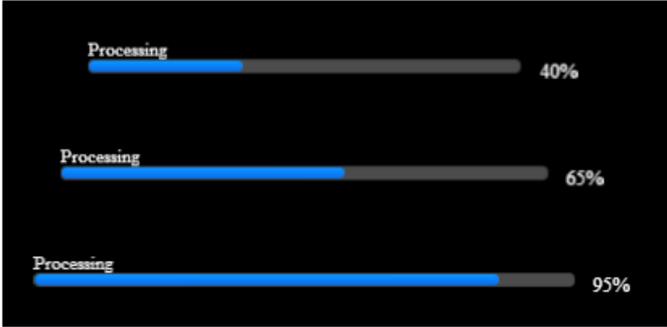
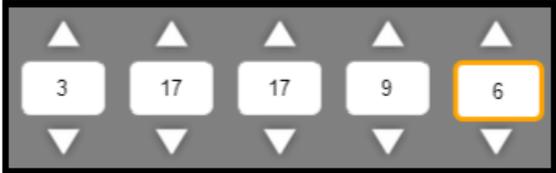
```
<div id="box_0" class="boxcls" ctype="box" style="width: 300px; height: 300px; opacity: 0.5; outline: none; position: absolute; -webkit-transform-style: preserve-3d; z-index: 1; -webkit-transform: translate3d(-50%, -50%, 0px) matrix3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 300, 300, 0, 1);"></div>
```

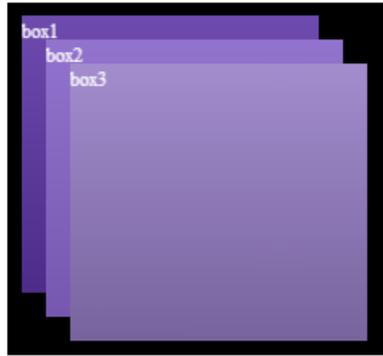
The structure of CAPH UI widget as follows.

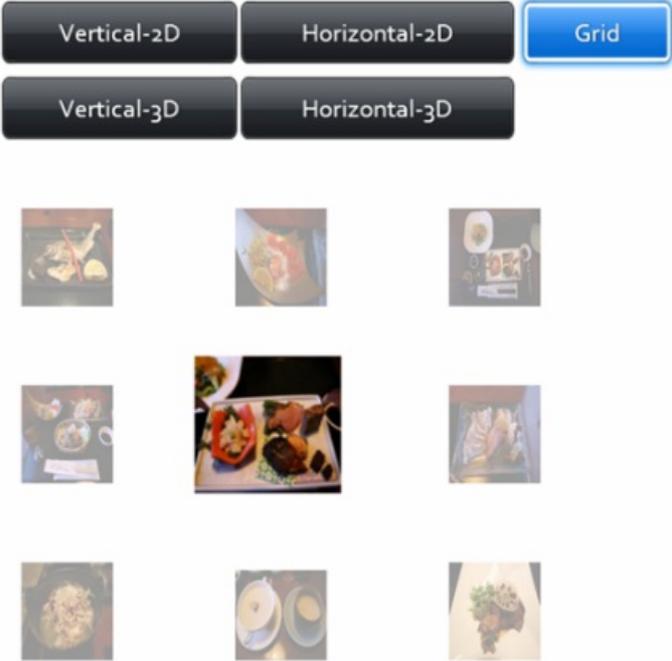


The all widgets list in the following form.

Component	Example photo
Box	
DomContainer	
Carousel	

Component	Example photo
Button Strip	
Navigator	
Progress Bar	
Spinner	
Image	
Gridwidget	

Component	Example photo
ListWidget	
ColorTag	
Panel	
Label	
InputBox	

Component	Example photo
Button	
Book	
ImageGallery	

## Open Issues

None

## Related Documents

[How To Customize CAPH WUI Widgets](#)

[How To Integrate CAPH WUI Widgets With jQuery Library](#)

[How To Use Animation With CAPH WUI Widgets](#)

[How To Use CAPH WUI Widgets With HTML](#)

[How To Create CAPH Application](#)

[CAPH Framework Overview](#)