

roCanvasWidget is a mix of roCanvasImage from Roku and our own widget code. It is created with an roRectangle in the same way as any other widget to set it's size and position on the screen.

The functions available are:

- Hide() to hide widget
- Show() to show widget
- SetLayer(object content, int z-level) sets the contents of a layer within the widget. The lowest z-level is drawn first, and the highest last. The content is described below.
- ClearLayer(int z-level) clears the layer specified.
- Clear() clears all of the layers
- EnableAutoRedraw(bool enable) disable and enables the automatic redrawing of the widget. When enabled, each call to SetLayer, ClearLayer or Clear results in a redraw. If multiple layers are to be changed then it's wise to disable auto redraw whilst calling the SetLayer calls and then it will be redrawn when it is reenabled. The redraw happens in a separate thread so EnableAutoRedraw returns pretty much immediately.

The content specified to each layer can consist of 1 or more objects each defined by an roAssociativeArray. If more than 1 object, then they are put into an roArray prior to passing to SetLayer. The three object types that currently exist are:

- Background color
 - color - #rrggbb hex value
 - targetRect - target rectangle which is another roAssociativeArray consisting of x, y, w and h values. These are relative to the top left corner of the widget.
- Text
 - text - A string to display
 - targetRect - the rectangle in which to show the text
 - textAttrs - an roAssociativeArray containing attributes to be applied to the text. These can be any of:
 - font - small/medium/large/huge the same as the Roku code
 - fontSize - a point size that is used directly when creating the font. If set to 0, then the font autosizes to fit the targetRect
 - fontFile - a filename for a non-system font to use
 - hAlign - left/center/right alignment of the text on a line
 - vAlign - top/center/bottom alignment of the text perpendicular to the line
 - rotation - 0/90/180/270 degrees rotation of text
 - color - #rrggbb hex value of text
- Image
 - filename - filename of image
 - targetRect - the rectangle in which to show the image. Image is stretched to fit
 - sourceRect - source rectangle to clip from source image
 - compositionMode - source or source_over. The latter alpha blends with underlying objects, the former replaces the underlying values completely.

Here's an example containing most of the features:

```

rect=CreateObject("roRectangle", 0, 0, 1920, 1080)
cw=CreateObject("roCanvasWidget", rect)

aa=CreateObject("roAssociativeArray")
aa["text"] = "Primal Scream"
aa["targetRect"] = { x: 280, y: 180, w: 500, h: 30 }
aa["textAttrs"] = { Color:"#AAAAAA", font:"Medium", HAlign:"Left", VAlign:"Top"}

aal=CreateObject("roAssociativeArray")
aal["text"] = "Movin' on up, followed by something else, followed by something else, followed by something else, followed by something else"
aal["targetRect"] = { x: 282, y: 215, w: 80, h: 500 }
aal["textAttrs"] = { Color:"#ffffff", font:"Large", fontfile:"usb1:/GiddyupStd.otf", HAlign:"Left", VAlign:"Top", rotation:"90"}

array=CreateObject("roArray", 10, false)
array.Push({ color: "5c5d5f" })
array.Push({ filename: "transparent-balls.png" })
array.Push(aa)

aa2=CreateObject("roAssociativeArray")
aa2["filename"] = "transparent-balls.png"
aa2["CompositionMode"] = "source_over"
aa2["targetRect"] = { x: 400, y: 200, w: 200, h: 200 }

cw.Show()
cw.EnableAutoRedraw(0)
cw.SetLayer(array, 0)
cw.SetLayer(aal, 1)
cw.SetLayer(aa2, 2)
cw.EnableAutoRedraw(1)

cw.ClearLayer(0)

```

/var/www/dokuwiki/data/pages/rocanvaswidget.txt · Last modified: 2011/11/02 15:39 by dcraig