

roCompositor

The roCompositor allows the composition and animation of multiple roBitmaps and roRegions.

Supported Interfaces

- ifCompositor

Description

The roCompositor can create and manage roSprites in a z-ordered list. The sprites can be of arbitrary size and can be thought of as planes. The compositor can manage collision detection between the sprites, support scrolling the sprite bitmap source, and support animated sprites (multi-frame sprites with frame-flipping animation). You may have multiple roCompositor components, and they can composite onto the same or separate bitmaps. That said, the most common scenario is to have a single roCompositor

Example: Scrolling a bitmap

```
Library "v30/bslCore.brs"
Function main()
    black=&hFF'RGBA
    screen=CreateObject("roScreen")
    compositor=CreateObject("roCompositor")
    compositor.SetDrawTo(screen, black)
    http = CreateObject("roUrlTransfer")
    http.SetMessagePort(CreateObject("roMessagePort"))
    http.SetUrl("http://rokudev.roku.com/rokudev/examples/scroll/VeryBigPng.png")
    http.AsyncGetToFile("tmp:/VeryBigPng.png")
    wait(0, http.GetPort())
    bigbm=CreateObject("roBitmap", "tmp:/VeryBigPng.png")
    region=CreateObject("roRegion", bigbm, 0, 0, 1280, 720)
    region.SetWrap(True)

    view_sprite=compositor.NewSprite(0, 0, region)
    compositor.draw()
    screen.SwapBuffers()
    msgport = CreateObject("roMessagePort")
    screen.SetMessagePort(msgport)
    codes = bslUniversalControlEventsCodes()
    While True
        msg=wait(0, msgport) ' wait for a button
        print "Msg: "; type(msg); " event: "; msg.GetInt()
        If type(msg)="roUniversalControlEvent" Then
            If msg.GetInt()=codes.BUTTON_UP_PRESSED Then
                Zip(screen, view_sprite, compositor, 0,-4) 'up
            Else If msg.GetInt()=codes.BUTTON_DOWN_PRESSED Then
                Zip(screen, view_sprite, compositor, 0,+4) ' down
            Else If msg.GetInt()=codes.BUTTON_RIGHT_PRESSED Then
                Zip(screen, view_sprite, compositor, +4,0) ' right
            Else If msg.GetInt()=codes.BUTTON_LEFT_PRESSED Then
                Zip(screen, view_sprite, compositor, -4, 0) ' left
            Else If msg.GetInt() = codes.BUTTON_BACK_PRESSED ' back button
                Exit While
            End If
        End If
    End While
End Function
Function Zip(screen, view_sprite, compositor, xd, yd)
    For x=1 To 60
        view_sprite.OffsetRegion(xd, yd, 0, 0)
        compositor.draw()
        screen.SwapBuffers()
    End For
End Function
```

