
Subject: Re: Why the font is smaller in the IDLgrBuffer ?

Posted by [natha](#) on Fri, 27 Feb 2009 20:41:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Rick,

I wrote a small routine to aproximate the font size in the buffer to the original font size.

It's a crazy solution. See below...

With your idea all results more easy.

The only thing to do is set the RESOLUTION property on the IDLgrButer. Like that:

```
resolution=[1d/!D.X_PX_CM,1d/!D.Y_PX_CM]  
oBuffer->SetProperty, RESOLUTION=resolution)
```

The problem is that !D.XY_PX_CM only returns the approximate number of pixels per centimeter in the X and Y directions so the font size changes a little bit beacause the resolution is not exact.

There is another way to get the DPI ?

Thanks,

Bernat

;-----

```
PRO TEST_BUFFER_FONT, font_size, dimensions, $      ;;input  
      final_font_size      ;;output
```

```
tt=SYSTIME(/SEC)
```

```
oView=OBJ_NEW('IDLgrView', VIEWPLANE_RECT=[0,0,dimensions  
[0],dimensions[1]])
```

```
oModel=OBJ_NEW('IDLgrModel')  
oModel->SetProperty, /DEPTH_WRITE_DISABLE  
oView->Add, oModel
```

```
buff_font=OBJ_NEW('IDLgrFont', 'Helvetica', SIZE=font_size, THICK=2)  
buff_text=OBJ_NEW('IDLgrText', STRING='null text', FONT=buff_font,  
RECOMPUTE_DIMENSIONS=2)  
oModel->Add, buff_text
```

```
oBuffer=OBJ_NEW('IDLgrBuffer', DIMENSIONS=dimensions)  
oBuffer->Draw, oView
```

```
buff_text_dim=oBuffer->GetTextDimensions(buff_text)
```

```
count=1.
WHILE buff_text_dim[1] LT font_size DO BEGIN
    buff_font->SetProperty, SIZE=font_size+count
    oBuffer->Draw, oView
    buff_text_dim=oBuffer->GetTextDimensions(buff_text)
    count=count+1.
ENDWHILE

PRINT, 'Font size / size after buffer ', font_size, buff_text_dim[1]

final_font_size=font_size+count

OBJ_DESTROY, [oView, oModel, oBuffer, buff_font, buff_text]

PRINT, 'TEST_BUFER_FONT TIME ', SYSTIME(/SEC)-tt
END
```
