Subject: Re: Object Graphics: Black Box
Posted by David Fanning on Thu, 23 Aug 2001 19:09:36 GMT
View Forum Message <> Reply to Message

Adam Rankin (arankin@irus.rri.on.ca) writes:
>
> A fun one to throw at you...
> I load a widget_draw and give it the GRAPHICS_LEVEL=2 keyword...
> I make an IDLgrImage,IDLgrModel,IDLgrView and a IDLgrScene
> and add them appropriately using data retrieved by READ_DICOM
> (I tested the data with TVSCL and it works)
> but when I load it into this draw widget all that appears is a black box!
>
> Any suggestions?

Uh, you are looking at the first pixel in the image. :-)

But don't worry, this is almost inevitably
what new object graphics users look at
when they first start. (If they can see
anything in their window.)

You have to make the arbitrary coordinatee system
of the viewplane rectangle match the coordinate
system of the image. In your case, I would probably
try setting the viewplane rectangle like this:

```
oImageView = obj_new('IDLgrView', $
  VIEWPLANE_RECT=[0,0,128,128])
```

Assuming the image is 128 by 128, that will fill
the image up in the window. (Which may or may not
be what you want.) But at least you will see
something, which is always half the battle with
object graphics. :-)

If you want another example, see my XImage program
here:

http://www.dfanning.com/programs/ximage.pro

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting

Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Object Graphics: Black Box
Posted by Adam Rankin on Thu, 23 Aug 2001 19:15:43 GMT
View Forum Message <> Reply to Message

> Adam Rankin (arankin@irus.rri.on.ca) writes:
>>
>> A fun one to throw at you...
>> I load a widget_draw and give it the GRAPHICS_LEVEL=2 keyword...
>> I make an IDLgrImage,IDLgrModel,IDLgrView and a IDLgrScene
>> and add them appropriately using data retrieved by READ_DICOM
>> (I tested the data with TVSCL and it works)
>> but when I load it into this draw widget all that appears is a black box!
>>
>> Any suggestions?
>
> Uh, you are looking at the first pixel in the image. :-)
>
> But don't worry, this is almost inevitably
> what new object graphics users look at
> when they first start. (If they can see
> *anything* in their window.)
>
> You have to make the arbitrary coordinatee system
> of the viewplane rectangle match the coordinate
> system of the image. In your case, I would probably
> try setting the viewplane rectangle like this:
>
>   oImageView = obj_new('IDLgrView', $
>   VIEWPLANE_RECT=[0,0,128,128])
>
> Assuming the image is 128 by 128, that will fill
> the image up in the window. (Which may or may not
> be what you want.) But at least you will see
> *something*, which is always half the battle with
> object graphics. :-)
>
> If you want another example, see my XImage program
> here:
>
>   http://www.dfanning.com/programs/ximage.pro
>
> Cheers,
>

> David
>
> --
> David W. Fanning, Ph.D.
> Fanning Software Consulting
> Phone: 970-221-0438, E-mail: david@dfanning.com
> Coyote's Guide to IDL Programming: http://www.dfanning.com/
> Toll-Free IDL Book Orders: 1-888-461-0155

Once again, another problem solved.
Of course it worked and I can see the image wonderfully...
Thanks for your help.

-Adam

(oh and I got a good kick out of the fact that you can easily recognize a
newbie object graphic programmer.)~

Subject: Re: Object Graphics: Black Box
Posted by Rick Towler on Thu, 23 Aug 2001 19:39:22 GMT
View Forum Message <> Reply to Message

Hi Adam,

First off, I should probably do the requisite scolding for using COMMON
blocks. So there, you have been scolded.

> oScene = obj_new('IDLgrScene')

You don't need the scene. Remove it and just draw the view. KISS.

> oImageView = obj_new('IDLgrView', LOCATION=[-64,-64],
> DIMENSIONS=[128,128], $
> VIEWPLANE_RECT=[-1,-1,2,2], ZCLIP=[1, -1])

I think you are confusing some of the View object's keywords. LOCATION
determines the lower left hand point where the view will be projected in the
2d window (your oWindow) and DIMENSIONS specifies the width and height of
this projection. These settings are akin to your monitors position and size
settings where you can shrink or expand your desktop to fill the entire
screen or just some part of it. Unless you want to project 2 or more views
into a single window the default values are fine.

Your real problem is with your VIEWPLANE_RECT. How big is the image you are
trying to display? Your view volume is set up to display a 2 pixel x 2

pixel image and since your image's coordinates run from 0 to n you will only see 1 of these pixels. You have two options, scale your image data to a range of 0..1 or better yet, enlarge your view volume to accommodate your entire image:

example for a 256x256 pixel image:

oImageView = obj_new('IDLgrView', VIEWPLANE_RECT=[0,0,256,256])

Give that a try. I think it should do the trick.

-Rick Towler

"Adam Rankin" <arankin@irus.rri.on.ca> wrote in message news:Pine.SUN.4.30.0108231327590.1754-100001@proxima.irus.rr i.on.ca...
> Hey guys,
>
> A fun one to throw at you...
> I load a widget_draw and give it the GRAPHICS_LEVEL=2 keyword...
> I make an IDLgrImage,IDLgrModel,IDLgrView and a IDLgrScene
> and add them appropriately using data retrieved by READ_DICOM
> (I tested the data with TVSCL and it works)
> but when I load it into this draw widget all that appears is a black box!
>
> Any suggestions?
>
> Also, I have been having trouble with my display so If anyone sees a
> normal image just say so and I'll kick my computer. =)
>
> Code:
>
> PRO loadimages, Ev
>
> COMMON graphics, ptrGraphStruct
> COMMON display, oWindow
>
> oPalette = obj_new('IDLgrPalette')
> oPalette -> LoadCT, 5
>
> oScene = obj_new('IDLgrScene')
> oImageView = obj_new('IDLgrView', LOCATION=[-64,-64],
> DIMENSIONS=[128,128], $
> VIEWPLANE_RECT=[-1,-1,2,2], ZCLIP=[1, -1])

```
>  oImageModel = obj_new('IDLgrModel', LIGHTING=2)
>
>  image =
>
 READ_DICOM('/home/irus/arankin/mrctbunny010523/2001may23/s27 _b1_TR1500/MR1.d
cm')
>
>  oImage = obj_new('IDLgrImage', image, PALETTE=oPalette)
>
>  oImageModel -> Add, oImage
>  oImageView -> Add, oImageModel
>  oScene -> Add, oImageView
>
>  ;create and load the new image(s)
>  WIDGET_CONTROL, (*ptrGraphStruct).draw, GET_VALUE=oWindow
>  oWindow -> SetProperty, GRAPHICS_TREE=oImageView
>  oWindow -> Draw, oScene
>
>  ;clear out the crap that is no longer needed
>  OBJ_DESTROY, oScene
>  OBJ_DESTROY, oPalette
>
>  ;make sure that everyone knows that the draw's are now drawn.
>  (*ptrGraphStruct).drawFlag='1'
>
>  END
>
>
 ;************************************************************ ****************
**
>  ;this simple procedure quits the application by calling /DESTROY to widget
>  control
>
 ;************************************************************ ****************
**
>
>  PRO quit, ev
>
>  COMMON graphics, ptrGraphStruct
>  COMMON display, oWindow
>
>  ;clear out that stuff
>  OBJ_DESTROY, oWindow
>  PTR_FREE, ptrGraphStruct
>  WIDGET_CONTROL, Ev.top, /DESTROY
>  END
>
>
```

```
;************************************************************ ****************
**
>  ;this is the main procedure, it loads the widgets and draws the
>  application
>
;************************************************************ ****************
**
>
>  PRO multidisplay, Ev
>
>  ;pass the common block
>  COMMON graphics, ptrGraphStruct
>
>  topBase = WIDGET_BASE(Title='Multiple Displays', /COLUMN,
>  MBAR=bar)
>  subBase = WIDGET_BASE(topBase, XSIZE=1024, YSIZE=425, /COLUMN)
>
>  fileMenu = WIDGET_BUTTON(bar, /MENU, value='File')
>  loadButtonMenu = WIDGET_BUTTON(fileMenu, Value='Load Images',
>  EVENT_PRO='loadimages')
>  quitButtonMenu = WIDGET_BUTTON(fileMenu, Value='Quit',
>  EVENT_PRO='quit')
>
>  draw = WIDGET_DRAW(subBase, GRAPHICS_LEVEL=2, XSIZE=1024,
>  YSIZE=400)
>
>  ;first time through, and their are no images so set flag to 0
>  drawFlag='0'
>
>  ;make a structure with all the widgets and pass the pointer
>  graphStruct = {topBase:topBase, subBase: subBase, draw:draw,
>  drawFlag:drawFlag}
>  ptrGraphStruct = PTR_NEW(graphStruct)
>
>
>  WIDGET_CONTROL, topBase, /REALIZE
>    XMANAGER, 'multidisplay', topBase
>
>  END
>
>  -Adam
>
```