Subject: Re: scripted object graphics question
Posted by David Fanning on Fri, 25 Feb 2011 13:43:28 GMT
View Forum Message <> Reply to Message

George Millward writes:

> I am trying to convert a bunch of old direct graphics programs to
> objects.
> These programs are all called by a cronjob/script so work in the
> background (no interactivity or windows produced).
>
> In essence they use this:
>
> set_plot,'z'
> write_png .....
>
> I now realize that I've never seen an example of object graphics
> working in a similar way - all object
> examples seem to be embedded in widgets and the like - based on human
> interactivity.
>
> I assume I can do this in object graphics ? Does anyone have a simple
> example of how this is done ?

The object graphics system, of course, *is* the Z-graphics
buffer. So, I presume the equivalent of a "hidden" window
would be the IDLgrClipboard.

I've never tried to run an object graphics program
(well, *any* program, truthfully!) in a cron job,
so I don't know about that. OpenGL seems inherently
tied to the graphics device, so I would guess you
would also need to have software rendering turned on
for the object graphics system.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: scripted object graphics question
Posted by George Millward on Fri, 25 Feb 2011 14:17:10 GMT
View Forum Message <> Reply to Message

On Feb 25, 6:43 am, David Fanning <n...@idlcoyote.com> wrote:
> George Millward writes:
>> I am trying to convert a bunch of old direct graphics programs to
>> objects.
>> These programs are all called by a cronjob/script so work in the
>> background (no interactivity or windows produced).
>
>> In essence they use this:
>
>> set_plot,'z'
>> write_png .....
>
>> I now realize that I've never seen an example of object graphics
>> working in a similar way - all object
>> examples seem to be embedded in widgets and the like - based on human
>> interactivity.
>
>> I assume I can do this in object graphics ? Does anyone have a simple
>> example of how this is done ?
>
> The object graphics system, of course, *is* the Z-graphics
> buffer. So, I presume the equivalent of a "hidden" window
> would be the IDLgrClipboard.
>
> I've never tried to run an object graphics program
> (well, *any* program, truthfully!) in a cron job,
> so I don't know about that. OpenGL seems inherently
> tied to the graphics device, so I would guess you
> would also need to have software rendering turned on
> for the object graphics system.
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

The cron part isn't important - that's easy - it's the whole idea of
creating graphics without
any windows or interactivity. We have a ton of IDL stuff - all direct
graphics - that runs

(say) every minute in the background. Satellite data comes in every minute, a script runs all of the
idl progs which generate .png files which are then immediately available in a web browser.
With direct graphics this is very solid and is the basis for almost 100% of our operational graphics output.

My problem is that I've never seen any examples of such use with object graphics. Software rendering should be fine.
Surely this is possible - or is object graphics always considered within the context of an actual window generated on
the screen ?

Cheers

George.

Subject: Re: scripted object graphics question
Posted by Michael Galloy on Fri, 25 Feb 2011 16:20:29 GMT
View Forum Message <> Reply to Message

On 2/25/11 6:32 am, George Millward wrote:
> I am trying to convert a bunch of old direct graphics programs to
> objects.
> These programs are all called by a cronjob/script so work in the
> background (no interactivity or windows produced).
>
> In essence they use this:
>
> set_plot,'z'
> write_png .....
>
> I now realize that I've never seen an example of object graphics
> working in a similar way - all object
> examples seem to be embedded in widgets and the like - based on human
> interactivity.
>
> I assume I can do this in object graphics ? Does anyone have a simple
> example of how this is done ?

Create your object graphics hierarchy as you would for screen display and then draw it with an IDLgrBuffer (for raster) or IDLgrClipboard (vector) object for output. If you can draw something to the screen, you only have to modify a line or two to draw it to IDLgrBuffer/IDLgrClipboard.

Mike
--

www.michaelgalloy.com
Research Mathematician
Tech-X Corporation

Subject: Re: scripted object graphics question
Posted by David Fanning on Fri, 25 Feb 2011 16:27:23 GMT
View Forum Message <> Reply to Message

Michael Galloy writes:

> Create your object graphics hierarchy as you would for screen display
> and then draw it with an IDLgrBuffer (for raster) or IDLgrClipboard
> (vector) object for output. If you can draw something to the screen, you
> only have to modify a line or two to draw it to IDLgrBuffer/IDLgrClipboard.

Right, and I would try this in a cron job sooner,
rather than later. It just seems to me there might be
a lot of "hidden" connections to a graphics device
in object graphics. :-(

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: scripted object graphics question
Posted by George Millward on Fri, 25 Feb 2011 16:30:01 GMT
View Forum Message <> Reply to Message

On Feb 25, 9:20 am, Michael Galloy <mgal...@gmail.com> wrote:
> On 2/25/11 6:32 am, George Millward wrote:
>
>
>
>> I am trying to convert a bunch of old direct graphics programs to
>> objects.
>> These programs are all called by a cronjob/script so work in the
>> background (no interactivity or windows produced).

>
>> In essence they use this:
>
>> set_plot,'z'
>> write_png .....
>
>> I now realize that I've never seen an example of object graphics
>> working in a similar way - all object
>> examples seem to be embedded in widgets and the like - based on human
>> interactivity.
>
>> I assume I can do this in object graphics ? Does anyone have a simple
>> example of how this is done ?
>
> Create your object graphics hierarchy as you would for screen display
> and then draw it with an IDLgrBuffer (for raster) or IDLgrClipboard
> (vector) object for output. If you can draw something to the screen, you
> only have to modify a line or two to draw it to IDLgrBuffer/IDLgrClipboard.
>
> Mike
> --www.michaelgalloy.com
> Research Mathematician
> Tech-X Corporation

Mike,

Thanks for the info - I've never played with IDLgrBuffer or
IDLgrclipboard
- I'll give it a go.

Cheers

George.

Subject: Re: scripted object graphics question
Posted by Karl[1] on Fri, 25 Feb 2011 17:36:58 GMT
View Forum Message <> Reply to Message

On Feb 25, 9:30 am, George Millward <george.millw...@noaa.gov> wrote:
> On Feb 25, 9:20 am, Michael Galloy <mgal...@gmail.com> wrote:
>
>
>
>> On 2/25/11 6:32 am, George Millward wrote:
>
>>> I am trying to convert a bunch of old direct graphics programs to
>>> objects.

>>> These programs are all called by a cronjob/script so work in the
>>> background (no interactivity or windows produced).
>
>>> In essence they use this:
>
>>> set_plot,'z'
>>> write_png .....
>
>>> I now realize that I've never seen an example of object graphics
>>> working in a similar way - all object
>>> examples seem to be embedded in widgets and the like - based on human
>>> interactivity.
>
>>> I assume I can do this in object graphics ? Does anyone have a simple
>>> example of how this is done ?
>
>> Create your object graphics hierarchy as you would for screen display
>> and then draw it with an IDLgrBuffer (for raster) or IDLgrClipboard
>> (vector) object for output. If you can draw something to the screen, you
>> only have to modify a line or two to draw it to IDLgrBuffer/IDLgrClipboard.
>
>> Mike
>> --www.michaelgalloy.com
>> Research Mathematician
>> Tech-X Corporation
>
> Mike,
>
> Thanks for the info - I've never played with IDLgrBuffer or
> IDLgrclipboard
> - I'll give it a go.
>
> Cheers
>
> George.

IDL still might try to make a connection to an X server, even if you try to keep it from doing so by choosing your graphics carefully. If that is the case, do a search on Xvfb here in this group. The idea is that you can write a script to start a virtual X server with Xvfb, sleep a bit to let it start, set your DISPLAY to point to the virtual server, start and run IDL, and then kill the virtual server. You'll be able to run this script from a cron job. It seems like a pretty common technique.

Subject: Re: scripted object graphics question

On Feb 25, 6:32 am, George Millward <george.millw...@noaa.gov> wrote:
> Hi there,
>
> I am trying to convert a bunch of old direct graphics programs to
> objects.
> These programs are all called by a cronjob/script so work in the
> background (no interactivity or windows produced).
>
> In essence they use this:
>
> set_plot,'z'
> write_png .....
>
> I now realize that I've never seen an example of object graphics
> working in a similar way - all object
> examples seem to be embedded in widgets and the like - based on human
> interactivity.
>
> I assume I can do this in object graphics ? Does anyone have a simple
> example of how this is done ?
>
> Cheers
>
> George.

Below is an example using IDLgrBuffer.

Kelly Dean
Milliken, CO

```
;+
;
; @file_comments
;  <P>Use object graphics to draw polygon with IDLgrBuffer as the
;  destination object.
;
 ;---------------------------------------------------------- -----------
PRO BufferOBJ

; Prepare a polygon (Hypocycloid)

scale  = 2.0
cusp   = 3.0
direct = 45.0 * !DTOR

xrad = INDGEN(360) * !DTOR
```

```
yrad = INDGEN(360) * !DTOR

x = scale * ( ( ( cusp - 1.0 ) * COS( xrad ) - direct * $
              COS( ( cusp - 1.0 ) * xrad ) ) )
y = scale * ( ( ( cusp - 1.0 ) * SIN( xrad ) + direct * $
              SIN( ( cusp - 1.0 ) * xrad ) ) )

deltoid      = FLTarr( 2, 360 )
deltoid[0,*] = TEMPORARY(y)
deltoid[1,*] = TEMPORARY(x)

; Add polygon to object graphics.

vp_rec = [ -15.0, -15.0, 30.0, 30.0 ]
navy = [0,0,128]
dims = [80,80]

oDest = OBJ_NEW( 'IDLgrBuffer' )
oDest -> SetProperty, DIMENSIONS = dims
oView = OBJ_NEW( 'IDLgrView', VIEWPLANE_REC = vp_rec )
oModel = OBJ_NEW( 'IDLgrModel' )
oEvolute = OBJ_NEW( 'IDLgrPolygon', COLOR = navy )
oEvolute -> SetProperty, DATA = deltoid
oModel -> ADD, oEvolute
oView -> ADD, oModel
oDest -> DRAW, oView

; Retrieve image from destination object.

oDest -> GetProperty, IMAGE_DATA = img3

; Heap variable clean up.

OBJ_DESTROY, oEvolute
OBJ_DESTROY, oModel
OBJ_DESTROY, oView
OBJ_DESTROY, oDest

; Save image as JPEG

outfile = 'c:\temp\cycloid.jpg'
WRITE_JPEG, outfile, img3, TRUE=1, QUALITY=100

PRINT, STRING( outfile, FORMAT ='( " -- All done : ", A0 )' )

END
```

Subject: Re: scripted object graphics question
Posted by George Millward on Fri, 25 Feb 2011 22:59:36 GMT
View Forum Message <> Reply to Message

On Feb 25, 12:31 pm, kBob <krd...@gmail.com> wrote:
> On Feb 25, 6:32 am, George Millward <george.millw...@noaa.gov> wrote:
>
>
>
>> Hi there,
>
>> I am trying to convert a bunch of old direct graphics programs to
>> objects.
>> These programs are all called by a cronjob/script so work in the
>> background (no interactivity or windows produced).
>
>> In essence they use this:
>
>> set_plot,'z'
>> write_png .....
>
>> I now realize that I've never seen an example of object graphics
>> working in a similar way - all object
>> examples seem to be embedded in widgets and the like - based on human
>> interactivity.
>
>> I assume I can do this in object graphics ? Does anyone have a simple
>> example of how this is done ?
>
>> Cheers
>
>> George.
>
> Below is an example using IDLgrBuffer.
>
> Kelly Dean
> Milliken, CO
>
> ;+
> ;
> ; @file_comments
> ; <P>Use object graphics to draw polygon with IDLgrBuffer as the
> ; destination object.
> ;
> ;-- -----------
> PRO BufferOBJ
>
> ; Prepare a polygon (Hypocycloid)
>

```
>  scale  = 2.0
>  cusp   = 3.0
>  direct = 45.0 * !DTOR
>
>  xrad = INDGEN(360) * !DTOR
>  yrad = INDGEN(360) * !DTOR
>
>  x = scale * ( ( ( cusp - 1.0 ) * COS( xrad ) - direct * $
>                  COS( ( cusp - 1.0 ) * xrad ) ) )
>  y = scale * ( ( ( cusp - 1.0 ) * SIN( xrad ) + direct * $
>                  SIN( ( cusp - 1.0 ) * xrad ) ) )
>
>  deltoid      = FLTarr( 2, 360 )
>  deltoid[0,*] = TEMPORARY(y)
>  deltoid[1,*] = TEMPORARY(x)
>
>  ; Add polygon to object graphics.
>
>  vp_rec = [ -15.0, -15.0, 30.0, 30.0 ]
>  navy = [0,0,128]
>  dims = [80,80]
>
>  oDest = OBJ_NEW( 'IDLgrBuffer' )
>  oDest -> SetProperty, DIMENSIONS = dims
>  oView = OBJ_NEW( 'IDLgrView', VIEWPLANE_REC = vp_rec )
>  oModel = OBJ_NEW( 'IDLgrModel' )
>  oEvolute = OBJ_NEW( 'IDLgrPolygon', COLOR = navy )
>  oEvolute -> SetProperty, DATA = deltoid
>  oModel -> ADD, oEvolute
>  oView -> ADD, oModel
>  oDest -> DRAW, oView
>
>  ; Retrieve image from destination object.
>
>  oDest -> GetProperty, IMAGE_DATA = img3
>
>  ; Heap variable clean up.
>
>  OBJ_DESTROY, oEvolute
>  OBJ_DESTROY, oModel
>  OBJ_DESTROY, oView
>  OBJ_DESTROY, oDest
>
>  ; Save image as JPEG
>
>  outfile = 'c:\temp\cycloid.jpg'
>  WRITE_JPEG, outfile, img3, TRUE=1, QUALITY=100
>
```

> PRINT, STRING(outfile, FORMAT ='(" -- All done : ", A0)')
>
> END

Wow - nailed it !!

I used the oDest = OBJ_NEW('IDLgrBuffer') as detailed above by Kelly.....
There was still an X-windows problem coming from TVLCT so I was thinking of the Xvfb solution
from Karl - but I realized that I didn't need TVLCT .... once you've decided on an r, g and b then it's just:
thisPalette=Obj_New('IDLgrPalette')
thisPalette -> setProperty, red_values = r
thisPalette -> setProperty, green_values = g
thisPalette -> setProperty, blue_values = b
... and that doesn't seem to need any X-window

...and it works !!!

Thanks to everyone for their help. Apologies if I'm a little over-exuberant but you know how it is
when it comes together!... Like I said above - our basic business is IDL graphics running in the background on LINUX...
and I'd been telling everyone that objects was the way to go... so I was getting a little anxious ;o) - and now it's just about beer-oclock on a Friday - perfect timing !!

Cheers

George.