Subject: Re: Object Graphics newbie question
Posted by Dick Jackson on Fri, 25 May 2001 21:37:39 GMT
View Forum Message <> Reply to Message

[sorry, that first message slipped out before I was done!]

Hi Arend,

"Arend Sluis" <sluis@physics.rutgers.edu> wrote in message
news:3B0E7558.8EA29250@physics.rutgers.edu...
> I want to make a 2D plot of, say, 10k data points, and each data point
> has associated with it a specific color and a set of secondary data
> points (an emission line spectrum). Since I want to be able to zoom in
> on and move around the data in order to make selection of individual
> data points easier, I thought that Object Graphics would be the way to
> go. However, it is 10k data points, so I clearly need to be careful in
> order to have efficient code.
>
> My first question is: should I be using Object Graphics

It may be overkill for the project as you've described, but the XObjView example program is a great starting point for a 3D viewing/inspecting application. If there's a useful way to turn your data into a 3D display of some kind, this may be especially helpful. Your mention that each point in 2D has a spectrum associated with it suggests that some 3D representation of the whole dataset at once may be possible. (or it may be just a mess)

Here's an example of how easy XObjView is to use (code examples are attached to this message as .pro files... hope it works for you):

=====
```
m=obj_new('IDLgrSurface', Dist(30), Color=[255,0,255], Style=2, $
        Shading=1, Name='Wide Magenta Surface')
g=obj_new('IDLgrSurface', Dist(10)*5, Color=[0,255,0], Style=2, $
        Shading=1, Name='Tall Green Surface')
xobjview,[m,g]
```
=====

You've got viewing controls, and a select tool which presents the name of whatever object you click on. If you want to do something else when an object is clicked, you might modify the code in idlexobjviewwid__define.pro around line 406. (I've done a bit of work extending these classes for my own purposes, but be warned: they are not documented, and I believe they are subject to change without notice.)

Now, a reality check shows that 10,000 separate objects (so that each is selectable with the Select tool), even if they are just points, is a lot for Object Graphics to handle.

```
=====
a=objArr(10000)
oSymbol=obj_New('idlgrsymbol',3)
for i=0,9999 do a[i]=obj_new('idlgrpolyline', $
                        RandomU(seed,3,1)*100, $
                        symbol=oSymbol, $
                        name=StrTrim(i,2))
xobjview,a
=====
```

(mine took 17 seconds to redraw while rotating)

But one IDLgrPolyline object with 10,000 points is a breeze!

```
=====
oSymbol=obj_New('idlgrsymbol',3)
b=obj_new('idlgrpolyline', $
        RandomU(seed,3,10000)*100, $
        Symbol=oSymbol, $
        Linestyle=6,Name='One big object')
xobjview,b
=====
```

This is where you might want to customize what happens when you click on the object, perhaps to report the location of the point you clicked on (add this in idlexobjviewwid__define.pro before the 'endif' on line 409):

```
                        pickedOK = self.oWindow-> $
                          PickData(oCurrent, oSelected[0], $
                                [event.x, event.y], pickXYZ)
                        Print, pickXYZ
```

Recompile it, click on a point and see the output log. A bit crude, but you get the idea.

If black points aren't enough, each datum can of course be represented by a polygon, a smoothly colored line, or whatever you please. The graphics classes really give you a lot of options. A colored point scatterplot can be done as follows:

```
=====
oSymbol=obj_New('idlgrsymbol',3)
c=obj_new('idlgrsurface', $
        DataX=RandomU(seed,100,100), $
        DataY=RandomU(seed,100,100), $
        DataZ=RandomU(seed,100,100), $
        Style=0,Vert_Colors=RandomU(seed,3,10000)*256, $
```

```
        Name='One big object')
xobjview,c
=====
```

(yes, we'd prefer to use a vector of 10000 for DataX|Y|Z, but IDLgrSurface
really wants a 2D array)

Well, that was quite a ramble. Hope it's of some help.

Cheers,
--
-Dick

Dick Jackson / dick@d-jackson.com
D-Jackson Software Consulting / http://www.d-jackson.com
Calgary, Alberta, Canada / +1-403-242-7398 / Fax: 241-7392

```
begin 666 ex0.pro
M;3UO8FI?;F5W*"=)1$QG<E-U<F9A8V4G+"!$:7-T*#,P*2P@0V]L;W(]6S(U
M-2PP+#(U-5TL(%-T>6QE/3(L("0-"B @(" @(" @("!3:&%D:6YG/3$L($YA
M;64])U=I9&4@36%G96YT82!3=7)F86-E)RD-"F<];V)J7VYE=R@G241,9W)3
M=7)F86-E)RP@1&ES="@Q,"DJ-2P@0V]L;W(]6S L,C4U+#!=+"!3='EL93TR
M+" D#0H@(" @(" @(" @4VAA9&EN9STQ+"!.86UE/2=486QL($=R965N(%-U
@<F9A8V4G*0T*>&]B:G9I97<L6VTL9UT-"@T*96YD#0H`
`
end

begin 666 ex1.pro
M83UO8FI!<G(H,3 P,# I#0IO4WEM8F]L/6]B:E].97<H)VED;&=R<WEM8F]L
M)RPS*0T*9F]R(&D],"PY.3DY(&1O(&%;:5T];V)J7VYE=R@G:61L9W)P;VQY
M;&EN92<L("0-"B @(" @(" @(" @(" @(" @(" @(" @(" @(" @4F%N9&]M
M52AS965D+#,L,2DJ,3 P+" D#0H@(" @(" @(" @(" @(" @(" @(" @(" @
M(" @('-Y;6)O;#UO4WEM8F]L+" D#0H@(" @(" @(" @(" @(" @(" @(" @
M(" @(" @(&YA;64]4W1R5')I;2AI+#(I*0T*>&]B:G9I97<L80T*#0IE;F0-
!"@``
`
end

begin 666 ex2.pro
M;U-Y;6)O;#UO8FI?3F5W*"=I9&QG<G-Y;6)O;"<L,RD-"F(];V)J7VYE=R@G
M:61L9W)P;VQY;&EN92<L("0-"B @(" @(" @("!286YD;VU5*'-E960L,RPQ
M,# P,"DJ,3 P+" D#0H@(" @(" @(" @4WEM8F]L/6]3>6UB;VPL("0-"B @
M(" @(" @("!,:6YE<W1Y;&4]-BQ.86UE/2=/;F4@8FEG(&]B:F5C="<I#0IX
2;V)J=FEE=RQB#0H-"F5N9 T*
`
end
```

```
begin 666 ex3.pro
M;U-Y;6)O;#UO8FI?3F5W*"=I9&QG<G-Y;6)O;"<L,RD-"F,];V)J7VYE=R@G
M:61L9W)S=7)F86-E)RP@) T*(" @(" @(" @($1A=&%8/5)A;F1O;54H<V5E
M9"PQ,# L,3 P*2P@) T*(" @(" @(" @($1A=&%9/5)A;F1O;54H<V5E9"PQ
M,# L,3 P*2P@) T*(" @(" @(" @($1A=&%:/5)A;F1O;54H<V5E9"PQ,# L
M,3 P*2P@) T*(" @(" @(" @(%-T>6QE/3 L5F5R=%]#;VQO<G,]4F%N9&]M
M52AS965D+#,L,3 P,# I*C(U-BP@) T*(" @(" @(" @($YA;64])T]N92!B
@:6<@;V)J96-T)RD-"GAO8FIV:65W+&,-"@T*96YD#0H`
`
end
```