Subject: Re: IDLgrLegend geometry
Posted by George Constantinides on Wed, 28 Feb 2001 04:32:25 GMT
View Forum Message <> Reply to Message

> In fact on my system legends do get resized as the size of the destination
> device changes! I wonder why they don't on yours, George. What version are
> you using? Do you have a line that looks like this in the CreateGlyphs
> method in idlgrlegend__define.pro?
>
> (*self.pTexts)[index] = OBJ_NEW('IDLgrText', $
> FONT = self.oFont, $
> COLOR = (*self.pText_Color), $
> STRINGS = (*self.pItem_Name)[index],$

> RECOMPUTE_DIMENSIONS = 2)

>> Now that I know it works for someone I'll go and have another look.
>
> It's puzzling that it works for me & not for you. It may be something to do
> with the fact that my test program has several layers of non-standard code
> between the IDLgrLegend and the IDLgrWindow.

I changed RECOMPUTE_DIMENSIONS=0 and recompiled IDLgrLegend__Define.pro
and it resized the way I expected (ie. the sizes of the legend text increased

proportionately to the new window size.) I will not use this as a permanent
fix though but it does advance my understanding of IDLgrLegend.

It appears that the default legend behavior is not consistent with other
objects such as IDLgrAxis, IDLgrText. I would have expected that, if you
are resizing a Window, all Text based objects either stay the same
size OR all Text objects resize proportionately to the new output object.
As it is, some resize and others don't even though I have not explicitly set
the RECOMPUTE_DIMENSIONS for any of them.

> Text characters are sized according to a text "box",
> whose width and height are given in the "data" units
> of your arbitrary coordinate system. (See the
> CHAR_DIMENSIONS keyword.) What RECOMPUTE_DIMENSIONS
> can do is tell you when to recompute the size of that
> text box. For example, if you change the data range,
> you will probably want to recompute your text box.
>
> But, and here is the point I was confused about, if
> you are just re-sizing the graphic there is no need
> to recompute the text box, since the data range

> doesn't change at all. In fact, in resizing windows
> you explicitly do NOT want to recompute dimensions.
>
> You can see this by downloading the Simple_Surface
> program from my web page:
>
> http://www.dfanning.com/programs/simple_surface.pro
>
> In this program, I have RECOMPUTE_DIMENSIONS set to
> 2. Notice when you resize the window that the text
> sizes remain the same size. (I still doesn't understand
> why this should be so, and I am looking for enlightenment
> on this point.) But if you change all the RECOMPUTE_DIMENSIONS=2
> to RECOMPUTE_DIMENSIONS=0 you will find that the text
> is size proportionally to the axes, the data, etc. This
> is the behaviour I want.

The IDL documentation is not clear on this issue but the
the use of RECOMPUTE_DIMENSIONS in "simple_surface.pro"
allows me to have another go at the interpretation:

<online help for IDLgrText>
;+
;RECOMPUTE_DIMENSIONS (Get, Set)
;
;Set this keyword to one of the following values to indicate when this text
;object's character dimensions (refer to the CHAR_DIMENSIONS property)
;are to be recomputed automatically:
;

;   the CHAR_DIMENSIONS property. If CHAR_DIMENSIONS is set to [0,0],
;   compute once and re-use the resulting dimensions until the
;   CHARACTER_DIMENSIONS are modified.

;-

When =0, Dave covers this pretty well in his post.
When =2, IDL recomputes the dimensions to maintain the size of the text when
applying the new transformation matrix. This is the size that was originally
set when the object was created or modified by the
oText->SetProperty Size=xxx method.

I hope, I am not stating the obvious here. Fill free to correct me if I have
completely missed the point.
--
Regards,
George Constantinides

Manly Hydraulics Laboratory
email: GeorgeC@mhl.nsw.gov.au
URL http://www.mhl.nsw.gov.au