Subject: Re: IDLgrAxis expanding
Posted by davidf on Wed, 14 Apr 1999 07:00:00 GMT
View Forum Message <> Reply to Message

I wrote after one too many beers:

> So here is how it is done. (It's a bit convoluted here,
> but it's a solution programmers will love.) I'm showing
> here just the calculations for the dependent data axis,
> but the other axes will be done in the same way:

Alright, I admit it. I was completely wrong. As Mark Hadfield points out, you can use the CRANGE keyword to get the real, autoscaled, range of the axis:

```
thisAxis->GetProperty, CRANGE=realRange
```

But this exercise wasn't completely pointless. I did come up with a true "programmers" solution, and I did learn a couple of things.

1. I learned something I didn't know about object axes and Lord knows I've plenty of room to learn more.

2. I learned that it doesn't matter if you have looked up the GetProperty method on an object 100 times and been told to go to the INIT method 100 times. If you look it up for the 101st time, you may find a keyword you *can't* find in the INIT section of the manual.

And I re-learned that most of the information I want *IS* in the IDL documentation, even if I don't know where to find it. :-)

But I am still confused about one thing: why "C"Range? What does that mean? I can understand "O"(output)Range, or "R"(real)Range, or "A"(autoscaled)Range. But I just don't get "C"Range. And I don't understand why I would be expected to know that to find what I want in the index of the on-line help. What I would prefer would be something like this:

```
Axis
  range
     setting
     obtaining the value of
```

I think even I could have found something like that. :-(

Cheers,

David

P.S. You still have to remember to apply the new axis
data range to the DATA, or your axis won't reflect your
true data values.

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

Subject: Re: IDLgrAxis expanding
Posted by davidf on Wed, 14 Apr 1999 07:00:00 GMT
View Forum Message <> Reply to Message

About a half hour ago I wrote this in response to a question
by Pavel Romashkin:

>> When an instance of IDLgrAxis is created without /EXACT keyword, it is
>> adjusted by IDL so that tick labels are nicely rounded. However, this
>> has a side effect of non-/EXACT axis extending past the limit of the
>> VIEWPLANE_RECTANGLE for the parent IDLgrView. Sometimes this expansion
>> is really large. Is there a way to make IDLgrAxis to round ticks but
>> lie inside the VIEWPLANE_RECTANGLE?
>
> If there is a way, it must be undocumented. :-(

Just after I finished this article (it being rather late) I said
to myself, "The hell with it, I'm going to have a beer". Of course,
I was out of beer. So on the drive to the liquor store the solution
came to me. :-)

I reasoned this way. *Someone* has to know the actual range of
the data, since it clearly gets drawn. Who is it who knows and
how can I get at that information? Using my best Sherlock
impersonation, I finally realized that it is the text object
that creates the labels that knows how long the axis is. So
how do I get at the labels? Humm. The STRINGS keyword! OK.

So here is how it is done. (It's a bit convoluted here,
but it's a solution programmers will love.) I'm showing
here just the calculations for the dependent data axis,

but the other axes will be done in the same way:

```
 ********************************************************

   ; Get the range of the data.

thisPlot->GetProperty, XRange=xrange, YRange=yrange

    ; Set up the scaling so that the axes for the plot and the
    ; plot data extends from 0->1 in the X and Y directions.

xs = Normalize(xrange)
ys = Normalize(yrange)

   ; Create the Y axis.

yAxis1 = Obj_New("IDLgrAxis", 1, Color=[255,255,0], Ticklen=0.025, $
    Minor=4, Title=ytitle, Range=yrange, YCoord_conv=ys, $
    Location=[0, 1000, 0])

   ; Get the text object containing the labels.

yAxis1->GetProperty, Ticktext=yAxisText

   ; Get the strings.

yAxisText->GetProperty, Strings=theseStrings

   ; Find the first and last of these strings. This is the REAL
   ; axis range.

nstrings = N_Elements(theseStrings)
minRange = Float(theseStrings[0])
maxRange = Float(theseStrings[nstrings-1])
newRange = [minRange, maxRange]

   ; Set the axis range to the REAL range.

yAxis1->SetProperty, Range=newRange

   ; Find new scaling values for this REAL range.

newYs = Normalize([minrange, maxrange])

   ; Re-scale both the axis AND the data into this new range.

yAxis1->SetProperty, Range=newRange, YCoord_Conv=newYs
thisPlot->SetProperty, YCoord_Conv=newYs
```

****

Works real good on my example program, even if it *is* a
bit of a hack. :-)

Cheers,

David

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

Subject: Re: IDLgrAxis expanding
Posted by davidf on Wed, 14 Apr 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Pavel Romashkin (promashkin@cmdl.noaa.gov) writes:

> When an instance of IDLgrAxis is created without /EXACT keyword, it is
> adjusted by IDL so that tick labels are nicely rounded. However, this
> has a side effect of non-/EXACT axis extending past the limit of the
> VIEWPLANE_RECTANGLE for the parent IDLgrView. Sometimes this expansion
> is really large. Is there a way to make IDLgrAxis to round ticks but
> lie inside the VIEWPLANE_RECTANGLE?

If there is a way, it must be undocumented. :-(

This is a maddening "feature" of the object axes. For example,
if you set the axis range from 0 to 12.6, IDL actually creates
an axis that has a range from 0 to 14. But if you ask the axis
what its range is:

```
  thisAxis->GetProperty, Range=thisRange
```

It reports that its range is 0 to 12.6. Aaaughh!

Of course, it is impossible to scale an axis like this into
any sort of coordinate system, so you have two choices. You
can always use exact axis scaling (which is what I do because
I'm fairly anal and I like my axes pairs to be at nice right
angles to one another and where I put them, for God's sake).
Or, you can have IDL scale them haphazardly and have axes

end points sticking out every which way and never meeting
at a point.

You may have noticed that none of the IDL object graphic
examples ever use box-style axes. This is the reason why.

In object graphic's defense, it is quite easy (well, alright,
once you figure it out, it is quite easy) to write your programs
so that the user can drag the axes around and put them wherever
he or she likes. I suppose you could always "nudge" your axes
back together once they appeared if you were picky about
appearance, but this does seem a little over the top for
nice looking axes. Of course, this does nothing to solve
the problem of axes extending outside the "plot space".

> Any suggestions?

Yes, several. If axes are going to autoscale, at least
have them report the values they autoscale to. Failing this,
publish the axis autoscaling algorithms so that we can create
better guesses as to what our axes are really going to do.

But, frankly, if line plots are what you want to do, and
you want object-like properties, it probably makes more
sense to write your own "plot object" in direct graphics.
This has the advantage of simplicity and familiarity.
Plot objects are powerful and work nicely. And they
take 5 seconds to print rather than 5 minutes. Anyone
who has taken one of my object-oriented programming
classes could build one in about a third the time it would
take to build an object graphics plot. :-)

Cheers,

David

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

Subject: Re: IDLgrAxis expanding
Posted by Pavel Romashkin on Thu, 15 Apr 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Hi David,
Well, you didn't have to look up the CRANGE keyword in the on-line help - I mentioned that keyword in my initial message! Doing taxes is really bad for you, we all know that :-)
Thank you - the solution surely will be to re-normalize and rescale data and axes into the range returned by CRANGE. This didn't come to my mind since repeated rescaling is not something you think is totally normal. Why in the world IDL's rounding expands axes PAST the viewplane rect. limits?!
So, you made smaller and simplier objects for plots? I should consider attending your class - F. Collins is not so far from where I live (Boulder). Unfortunately, mastering IDL is not quite what I am getting paid for - papers that result from IDL use are. Travel and class fees are not among the things most bosses like the best. Oh well, we'll see.
Thank you again and anjoy that beer!
Pavel

Subject: Re: IDLgrAxis expanding
Posted by davidf on Thu, 15 Apr 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Gwyn Fireman (fireman@mcst.gsfc.nasa.gov) writes:

> : But I am still confused about one thing: why "C"Range?
>
> I always thought it was "current" range.
> It may have been documented as such long ago, but I can't find it now.

No, I can't find it either, but I've already demonstrated my sleuthing ability. Stein Vidar thinks "coordinate" range. But either way, "current" or "coordinate", it comes up empty in a search of the index. I've even been searching on "B"Range and "D"Range without luck this morning. :-(

Cheers,

David

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

Subject: Re: IDLgrAxis expanding

Posted by fireman on Thu, 15 Apr 1999 07:00:00 GMT
View Forum Message <> Reply to Message

David Fanning (davidf@dfanning.com) wrote:
: But I am still confused about one thing: why "C"Range?

I always thought it was "current" range.
It may have been documented as such long ago, but I can't find it now.

--
-- Gwyn F. Fireman
-- General Sciences Corporation / MODIS Characterization Support Team
-- Gwyn.Fireman@gsfc.nasa.gov 301-352-2118