Subject: Re: object graphics, exploding axis text, how to fix by explicitly setting char_dimens?
Posted by lecacheux.alain on Wed, 11 Dec 2013 14:01:00 GMT
View Forum Message <> Reply to Message

Le mercredi 11 décembre 2013 12:59:27 UTC+1, AJAS a écrit :
> On Tuesday, 10 December 2013 20:35:48 UTC, Starbuck wrote:
>
>> On Friday, November 29, 2013 2:01:04 PM UTC-7, jkj wrote:
>
>>
>
>>> Hi,
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> The researchers I work for [San Antonio, Southwest Research Institute, Space Science & Engineering] are using object graphics for the interactivity it provides.
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> We work with a lot of very small-valued data (electron precipitation measurements). In certain units the mean of our data will typically be 1.e-14. We consider 1.d-25 to be a reasonable measure of zero.

>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> There are many quirks presented by IDL when data values become very small. One such quirk is "exploding text", as seen in this random selection of line plots, two showing well-behaved text (goodExample*.png) and the other two showing exploded text (badExample*.png):
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> http://safaripass.com/goodExample.png
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>

>
>>> http://safaripass.com/badExample.png
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> http://safaripass.com/goodExample2.png
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> http://safaripass.com/badExample2.png
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>

>
>>> I set recompute_dimensions=2 and allow IDL to calculate the size of the text but have also tried values of '0' and '1'. It does not, however, appear that this approach (IDL internal calculation of tick text dimensions) will work reliably with our data sets. We are working with IDL 8.1, I see exactly the same behaviour in IDL 5.5, no change between those versions.
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>

>>> I can make "most of the plots" well-behaved "most of the time" by paying careful attention to the yrange (or whatever range is very-small-valued) of the data. For example, if the data does not cover a full decade in log space then I need to increase the min/max for the range until a full decade is represented. I also need to avoid ranges with values like "1.129e-14", choosing something like "1.e-14" or "1.1e-14". This means I can not choose arbitrary ranges for fear of text explosion... so we try then choose to sacrifice the best view of the data for well-behaved text.
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>

>>> By and large I can "whack the range around" and get the tick text to behave well. It seems to me that the real solution is to calculate my own character dimensions and I think I understand how to do that but simple examples with large-valued data work but when I switch to our very-small data the approaches taken with the larger data values then fail and I get the same exploded text as when character dimensions are calculated internally by IDL.
>
>>

>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> That leaves me wondering if there are good examples of explicitly computing character dimensions laying around somewhere... and it also leaves me wondering if those of us who work with very-small data values are left at the mercy of something internal to IDL that ignores the possibility of such small-valued data sets.
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> Any thoughts would be appreciated. If IDL is simply internally deficient with respect to very small data then I should switch to some other method of putting up text. I am able to reliably display text summaries of ranges as a 'plot title' and those 'idlgrtext' summaries have yet to behave poorly, but the tick text behaviour is really unacceptable.
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>

>
>>
>
>>> Thanks,
>
>>
>
>>>
>
>>
>
>>> -Kevin
>
>>
>
>>
>
>>
>
>> I was able to reproduce the problem with the following code:
>
>>
>
>>
>
>>
>
>> h = double(HANNING(100,100)*2.3e-13)
>
>>
>
>> s = surface(h,COLOR='black', style=1, CLIP=0)
>
>>
>
>>
>
>>
>
>> I have filed a bug report IDL-68998 about the issue.
>
>>
>
>>
>
>>
>
>> -Starbuck

>
>
>
> I also get strange behaviour in contour.
>
>
>
> IDL> c = CONTOUR(h,C_LABEL_SHOW=1,XRANGE=[60,80])
>
> IDL> HELP, !VERSION
>
> ** Structure !VERSION, 8 tags, length=104, data length=100:
>
> ARCH STRING 'x86_64'
>
> OS STRING 'linux'
>
> OS_FAMILY STRING 'unix'
>
> OS_NAME STRING 'linux'
>
> RELEASE STRING '8.2'
>
> BUILD_DATE STRING 'Apr 10 2012'
>
> MEMORY_BITS INT 64
>
> FILE_OFFSET_BITS
>
> INT 64
>
>
>
>
>
>
> Andy

It seems to be a scaling problem. With c.ASPECT_RATIO=1, the labels are well formed, then get unreadable when c.ASPECT_RATIO goes down to 0 (the default).
alx.
