Subject: Re: IDLgrAxis and scaling
Posted by Mark Hadfield on Wed, 19 Mar 2003 21:27:18 GMT
View Forum Message <> Reply to Message

"Thomas Gutzler" <tgutzler@ee.uwa.edu.au> wrote in message
news:3E783423.4080505@ee.uwa.edu.au...
> Hi,
>
> I tried to prevent the axis-text to be scaled while scaling a model with
> the axes added.

This is controlled by the RECOMPUTE_DIMENSIONS property of the axis-text IDLgrText objects. Default is 0; you should use 1 or 2.

It's easy to specify RECOMPUTE_DIMENSIONS for a text object you create yourself. But the tick-label objects are generated automatically, so you have to hunt them down. My MGHgrAxis object does this. The axis object keeps properties FONT and RECOMPUTE_DIMENSIONS, which it applies to all the text objects it manages. The code looks like this:

```
  self->GetProperty, TICKTEXT=oticktext, TITLE=otitle
  if obj_valid(otitle) then begin
    otitle->SetProperty, RECOMPUTE_DIMENSIONS=self.recompute_dimensions
    otitle->GetProperty, FONT=font
    if not obj_valid(font) then otitle->SetProperty, FONT=self.font
  endif
  for i=0,n_elements(oticktext)-1 do begin
    if obj_valid(oticktext[i]) then begin
      oticktext[i]->SetProperty, $
          RECOMPUTE_DIMENSIONS=self.recompute_dimensions
      oticktext[i]->GetProperty, FONT=font
      if not obj_valid(font) then $
          oticktext[i]->SetProperty, FONT=self.font
    endif
  endfor
```

To look at this code in context, see

 http://www.dfanning.com/hadfield/README.html

--
Mark Hadfield "Ka puwaha te tai nei, Hoea tatou"
m.hadfield@niwa.co.nz
National Institute for Water and Atmospheric Research (NIWA)

Subject: Re: IDLgrAxis and scaling

Posted by Thomas Gutzler on Thu, 20 Mar 2003 02:02:31 GMT
View Forum Message <> Reply to Message

Mark Hadfield wrote:
> "Thomas Gutzler" <tgutzler@ee.uwa.edu.au> wrote in message
> news:3E783423.4080505@ee.uwa.edu.au...
>
>> Hi,
>>
>> I tried to prevent the axis-text to be scaled while scaling a model with
>> the axes added.
>
>
> This is controlled by the RECOMPUTE_DIMENSIONS property of the axis-text
> IDLgrText objects. Default is 0; you should use 1 or 2

Yeah, that's what I thought, too.

Where is the difference between this:

```
>    self->GetProperty, TICKTEXT=oticktext, TITLE=otitle
>    for i=0,n_elements(oticktext)-1 do begin
>       if obj_valid(oticktext[i]) then begin
>          oticktext[i]->SetProperty, $
>              RECOMPUTE_DIMENSIONS=self.recompute_dimensions
>          oticktext[i]->GetProperty, FONT=font
>          if not obj_valid(font) then $
>              oticktext[i]->SetProperty, FONT=self.font
>       endif
>    endfor
```

and that:
```
zaxis->GetProperty, Ticktext=text
text->SetProperty, Recompute_Dimensions=1
```

I think there is none. The difference is somewhere else:
In MGHgrAxis::Init:
```
  self.recompute_dimensions = $
    n_elements(recompute_dimensions) gt 0 ? recompute_dimensions : 2
                                            ^
```
Setting Recompute_Dimensions=2 instead of 1 solves it.

Thanks for that.

Tom

Subject: Re: IDLgrAxis and scaling

Posted by David Fanning on Thu, 20 Mar 2003 02:09:30 GMT
View Forum Message <> Reply to Message

Thomas Gutzler (tgutzler@ee.uwa.edu.au) writes:

> I think there is none. The difference is somewhere else:
> In MGHgrAxis::Init:
> self.recompute_dimensions = $
> n_elements(recompute_dimensions) gt 0 ? recompute_dimensions : 2
> ^
> Setting Recompute_Dimensions=2 instead of 1 solves it.

I was going to recommend this method I use (especially
in object graphics) of trying every possible value for
every possible keyword, but it seemed, well, inelegant. :-(

Cheers,

David

P.S. Let's just say I'm pretty sure one of those funny
cigarettes was being passed around when that Recompute_Dimensions
documentation was being written.

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

Subject: Re: IDLgrAxis and scaling
Posted by Mark Hadfield on Thu, 20 Mar 2003 02:13:37 GMT
View Forum Message <> Reply to Message

"Thomas Gutzler" <tgutzler@ee.uwa.edu.au> wrote in message
news:3E792137.60009@ee.uwa.edu.au...
> Mark Hadfield wrote:
> Where is the difference between this:
>
>> self->GetProperty, TICKTEXT=oticktext, TITLE=otitle
>> for i=0,n_elements(oticktext)-1 do begin
>> if obj_valid(oticktext[i]) then begin
>> oticktext[i]->SetProperty, $
>> RECOMPUTE_DIMENSIONS=self.recompute_dimensions
>> oticktext[i]->GetProperty, FONT=font
>> if not obj_valid(font) then $

>>   oticktext[i]->SetProperty, FONT=self.font
>>   endif
>>   endfor
>
> and that:
> zaxis->GetProperty, Ticktext=text
> text->SetProperty, Recompute_Dimensions=1

Where does "that" come from? Did I miss a message?

The differences are:

- "that" is a lot shorter

- My code ("this") allows for the possibility that TICKTEXT is an array of IDLgrText object references. When IDL generates tick-text automatically it creates a single object, but it is possible for an array of object references to have been added via the axis's Init or SetProperty methods.

- My code also applies a default FONT property.

- As you noted elsewhere, my code allows self.recompute_dimensions to be set by the MGHgrAxis object's Init or SetProperty methods, but the default is 2. The IDL documentation describes in detail the difference between RECOMPUTE_DIMENSIONS = 0, 1, and 2, but I assumed you could read that for yourself. 2 see,s top do what I want.

--
Mark Hadfield "Ka puwaha te tai nei, Hoea tatou"
m.hadfield@niwa.co.nz
National Institute for Water and Atmospheric Research (NIWA)

Subject: Re: IDLgrAxis and scaling
Posted by Thomas Gutzler on Thu, 20 Mar 2003 04:13:06 GMT
View Forum Message <> Reply to Message

Mark Hadfield wrote:
> "Thomas Gutzler" <tgutzler@ee.uwa.edu.au> wrote in message
> news:3E792137.60009@ee.uwa.edu.au...
>
>> Mark Hadfield wrote:
>> Where is the difference between this:
>>
>>
>>>>   self->GetProperty, TICKTEXT=oticktext, TITLE=otitle
>>>>   for i=0,n_elements(oticktext)-1 do begin
>>>>   if obj_valid(oticktext[i]) then begin

```
>>>         oticktext[i]->SetProperty, $
>>>             RECOMPUTE_DIMENSIONS=self.recompute_dimensions
>>>         oticktext[i]->GetProperty, FONT=font
>>>         if not obj_valid(font) then $
>>>             oticktext[i]->SetProperty, FONT=self.font
>>>       endif
>>>    endfor
>>
>> and that:
>> zaxis->GetProperty, Ticktext=text
>> text->SetProperty, Recompute_Dimensions=1
>
>
>  Where does "that" come from? Did I miss a message?
```

Did my - or your - newsserver mess up my posting ?
I can't believe that it cuts random lines off. "that" comes from my very first posting of "this" thread :)
http://groups.google.com/groups?selm=3E783423.4080505%40ee.uwa.edu.au&oe=UTF-8&output=gplain
ACK?

Tom

Subject: Re: IDLgrAxis and scaling
Posted by Mark Hadfield on Thu, 20 Mar 2003 04:42:03 GMT
View Forum Message <> Reply to Message

"Thomas Gutzler" <tgutzler@ee.uwa.edu.au> wrote in message news:3E793FD2.5020900@ee.uwa.edu.au...
> Did my - or your - newsserver mess up my posting ?
> I can't believe that it cuts random lines off.

That's my story and I'm sticking to it :-)

The alternative (that I didn't read the entire thread carefully and remember everything I'd read) is too apalling to contemplate.

--
Mark Hadfield "Ka puwaha te tai nei, Hoea tatou"
m.hadfield@niwa.co.nz
National Institute for Water and Atmospheric Research (NIWA)
