Subject: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
Posted by Paul Van Delst[1] on Wed, 12 Dec 2012 17:06:34 GMT
View Forum Message <> Reply to Message

Hello,

I'm creating a graphics window and storing the id in a hash like so:

```
gref[tag] = WINDOW( WINDOW_TITLE=tag, $
                    DIMENSIONS = [600,800] )
```

I then do a bunch of plotting and legend-ing in that window using the /CURRENT keyword.

The "gref" variable is passed out of my procedure so that, if I decide to modify the plots (e.g. zoom, add arrows, text, whatever) interactively, I can still save it. I've done this in the past and it has always worked fine.

Now when I try to do it I get an error:

```
IDL> gref['afn'].save,'afn.png',height=800
% SAVE: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'.
% Execution halted at: $MAIN$
```

Has anyone seen this error before? Or explain it to me? I can find anything in the IDL help about either the object (IDLITSYMBOL) or the method (IS3D)

Thanks for any info.

cheers,

paulv

p.s. IDL> print, !version
{ x86 linux unix linux 8.2 Apr 10 2012 32 64}

Subject: Re: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
Posted by Paul Van Delst[1] on Wed, 12 Dec 2012 20:59:01 GMT
View Forum Message <> Reply to Message

A little bit more digging found this:

-----%<-----
IDLitVisualization::Is3D

The IDLitVisualization::Is3D function method determines
whether or not this visualization is marked as being
three-dimensional.

Syntax
Result = Obj->[IDLitVisualization::]Is3D()

Return Value
Returns 1 if this visualization is marked as being
three-dimensional, or 0 if it is not three-dimensional.

Arguments
None

Keywords
None
-----%<-----

Based on this I started digging into the various class hierarchies
trying to figure out what is inherited by whom.

Following the list of <Direct> inherited objects...

```
IDL> help, obj_new('idlitsymbol'), /objects
** Object class IDLITSYMBOL, 1 direct superclass, 4 known methods
   Superclasses:
      IDLGRMODEL  <Direct>
      IDLGRCONTAINER
      IDL_CONTAINER
      IDLGRCOMPONENT
      IDLITCOMPONENT

IDL> help, obj_new('idlgrmodel'), /objects
** Object class IDLGRMODEL, 1 direct superclass, 3 known methods
   Superclasses:
      IDLGRCONTAINER  <Direct>
      IDL_CONTAINER
      IDLGRCOMPONENT
      IDLITCOMPONENT

IDL> help, obj_new('idlgrcontainer'), /objects
** Object class IDLGRCONTAINER, 2 direct superclasses, 2 known methods
   Superclasses:
      IDL_CONTAINER  <Direct>
      IDLGRCOMPONENT  <Direct>
      IDLITCOMPONENT
```

IDL_CONTAINER has no superclasses (I think) so:

```
IDL> help, obj_new('idlgrcomponent'), /objects
** Object class IDLGRCOMPONENT, 1 direct superclass, 2 known methods
   Superclasses:
      IDLITCOMPONENT  <Direct>
```

And IDLITCOMPONENT similarly has no superclasses.

So how can IDLITSYMBOL ever get access to the IS3D methjod of IDLITVISUALIZATION? Via a contained object somehow?

Hmm. Let's go the other way:

```
IDL> help, obj_new('idlitvisualization'), /objects
** Object class IDLITVISUALIZATION, 2 direct superclasses, 6 known methods
   Superclasses:
      _IDLITVISUALIZATION  <Direct>
      _IDLITCONTAINER
      _IDLITPROPERTYAGGREGATE
      IDLGRMODEL
      IDLGRCONTAINER
      IDL_CONTAINER
      IDLGRCOMPONENT
      IDLITCOMPONENT
      IDLITSELECTPARENT
      IDLITIMESSAGING
      IDLITPARAMETER  <Direct>
```

Oi vey!

```
IDL> help, obj_new('_idlitvisualization'), /objects
** Object class _IDLITVISUALIZATION, 5 direct superclasses, 7 known methods
   Superclasses:
      _IDLITCONTAINER  <Direct>
      _IDLITPROPERTYAGGREGATE  <Direct>
      IDLGRMODEL  <Direct>
      IDLGRCONTAINER
      IDL_CONTAINER
      IDLGRCOMPONENT
      IDLITCOMPONENT
      IDLITSELECTPARENT  <Direct>
      IDLITIMESSAGING  <Direct>
```

Crikey.

On 12/12/12 12:06, Paul van Delst wrote:

> Hello,
>
> I'm creating a graphics window and storing the id in a hash like so:
>
> gref[tag] = WINDOW(WINDOW_TITLE=tag, $
> DIMENSIONS = [600,800])
>
> I then do a bunch of plotting and legend-ing in that window using the
> /CURRENT keyword.
>
> The "gref" variable is passed out of my procedure so that, if I decide
> to modify the plots (e.g. zoom, add arrows, text, whatever)
> interactively, I can still save it. I've done this in the past and it
> has always worked fine.
>
> Now when I try to do it I get an error:
>
> IDL> gref['afn'].save,'afn.png',height=800
> % SAVE: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'.
> % Execution halted at: $MAIN$
>
> Has anyone seen this error before? Or explain it to me? I can find
> anything in the IDL help about either the object (IDLITSYMBOL) or the
> method (IS3D)
>
> Thanks for any info.
>
> cheers,
>
> paulv
>
>
> p.s. IDL> print, !version
> { x86 linux unix linux 8.2 Apr 10 2012 32 64}
>

Subject: Re: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
Posted by David Fanning on Wed, 12 Dec 2012 21:19:48 GMT
View Forum Message <> Reply to Message

Paul van Delst writes:

> Crikey.

Indeed. I'm sending a case of Tylenol your way for
Christmas. You are going to need it if you keep investigating
iTool code! ;-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
Posted by Paul Van Delst[1] on Wed, 12 Dec 2012 21:37:52 GMT
View Forum Message <> Reply to Message

On 12/12/12 16:19, David Fanning wrote:
> Paul van Delst writes:
>
>> Crikey.
>
> Indeed. I'm sending a case of Tylenol your way for
> Christmas. You are going to need it if you keep investigating
> iTool code! ;-)

Dunno if tylenol will do it. Maybe a short vacay to Washington state...?

:oD

Subject: Re: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
Posted by Paul Van Delst[1] on Wed, 12 Dec 2012 22:25:49 GMT
View Forum Message <> Reply to Message

Alrighty, I've figured out the cause.

After the main plots (looping over some data) I plot simple grid lines
on teh first loop iteration (because the FG ticklen/gridstyle is
hopelessly busted in v8.2):

-----%<-----

...looping over "i"...

afn_plotref[2,i] = PLOT(f_afn.x, f_afn.y, $

```
   TITLE='Band '+STRTRIM(band,2)+' CrIS Apodisation Function @ +Xmax', $
   XTITLE='Optical delay (cm)', $
   XRANGE=[xmax-delta,xmax+delta], $
   YRANGE=[-0.01,0.1], $
   COLOR=color, $
   SYMBOL=symbol, $
   LAYOUT=[1,3,3], $
   OVERPLOT=overplot[2], $
   /CURRENT)
IF ( i EQ 0 ) THEN BEGIN
   pm = PLOT([xmax,xmax],afn_plotref[2,i].Yrange, $
     LINESTYLE=1, $
     OVERPLOT=afn_plotref[2,i], $
     /CURRENT)
   pm.Order, /SEND_TO_BACK
   pz = PLOT(afn_plotref[2,i].Xrange,[0,0], $
     LINESTYLE=1, $
     OVERPLOT=afn_plotref[2,i], $
     /CURRENT)
   pz.Order, /SEND_TO_BACK
ENDIF

-----%<-----
```

If I comment out *both* the
```
   pm.Order, /SEND_TO_BACK
```
and
```
   pz.Order, /SEND_TO_BACK
```
statements, everything works.

Otherwise, I get the
```
   Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
```
error when I invoke the SAVE method on the graphic.

This little quest (and the idiosyncracies of FG in general) reminds me of a question I had in my final Physics exam back in high school. I don't remember the actual question, just our post-exam reactions to it and our subsequent morphing of the problem to ridiculous proportions. It sorta went like:

> Facing due north, you start a ball rolling down a plane inclined at 10deg and of length 10m. A westerly wind is blowing at 2.5ms^-1. Using g=10.0ms^-2, what is the orbital velocity of Jupiter?
>
> Show all workings.

How are these things at all related?

:o)

cheers,

paulv

On 12/12/12 12:06, Paul van Delst wrote:
> Hello,
>
> I'm creating a graphics window and storing the id in a hash like so:
>
> gref[tag] = WINDOW(WINDOW_TITLE=tag, $
> DIMENSIONS = [600,800])
>
> I then do a bunch of plotting and legend-ing in that window using the
> /CURRENT keyword.
>
> The "gref" variable is passed out of my procedure so that, if I decide
> to modify the plots (e.g. zoom, add arrows, text, whatever)
> interactively, I can still save it. I've done this in the past and it
> has always worked fine.
>
> Now when I try to do it I get an error:
>
> IDL> gref['afn'].save,'afn.png',height=800
> % SAVE: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'.
> % Execution halted at: $MAIN$
>
> Has anyone seen this error before? Or explain it to me? I can find
> anything in the IDL help about either the object (IDLITSYMBOL) or the
> method (IS3D)
>
> Thanks for any info.
>
> cheers,
>
> paulv
>
>
> p.s. IDL> print, !version
> { x86 linux unix linux 8.2 Apr 10 2012 32 64}
>

Subject: Re: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
Posted by chris_torrence@NOSPAM on Thu, 13 Dec 2012 03:14:38 GMT
View Forum Message <> Reply to Message

Hi Paul,

This sounds like a bug to me. I'll try to reproduce it, but if you could come up with a simple reproduce case, that would be very helpful.

By the way, you said that the "ticklen/gridstyle is hopelessly busted in v8.2". What's wrong with it? If there is a bug, I'd like to fix that before IDL 8.2.2.

Thanks!

-Chris
ExelisVIS

Subject: Re: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
Posted by Paul Van Delst[1] on Thu, 13 Dec 2012 14:23:33 GMT
View Forum Message <> Reply to Message

Hi Chris,

Re: IDLITSYMBOL::IS3D error - I will try and come up with a small reproducible case.

Re: the tickmark issue:

In FG when I use [XY]ticklen = 1.0 and [XY]gridstyle=1 *all* the tickmarks are gridded, both major and minor. Try:

```
IDL> p=plot(/test,layout=[1,3,1],$ &
IDL>        yticklen=1.0,ygridstyle=1,$ &
IDL>        xticklen=1.0,xgridstyle=1)
```

My result:
http://ftp.emc.ncep.noaa.gov/jcsda/CRTM/.plots/all_ticks_gri dded.png

In DG only the major tickmarks were affected -- which is how the documentation describes the behaviour for both FG and DG.

As you can imagine, having all the tickmarks gridded makes for a rather busy plot.

Upon further reading of the docs I saw the [XY]subticklen keyword. I gave that a try, setting it to 0.05. But the minor tickmarks are *still*

subject to the [XY]gridstyle keyword (I tend to use [XY]gridstyle=1,
dotted, for this).

Try:
```
IDL> p=plot(/test,layout=[1,3,1],$ &
IDL>       yticklen=1.0,ygridstyle=1,ysubticklen=0.05,$ &
IDL>       xticklen=1.0,xgridstyle=1,xsubticklen=0.05)
```

My result:
ftp://ftp.emc.ncep.noaa.gov/jcsda/CRTM/.plots/using_subtickl en.png

So, as you can see, what I have are uneven length minor tickmarks. And
they are still dotted due the gridstyle keyword.

What I want is the equivalent of the following DG:
```
IDL> !p,multi=[0,1,3]
IDL> plot, lindgen(100),$ &
IDL>       yticklen=1.0,ygridstyle=1,$ &
IDL>       xticklen=1.0,xgridstyle=1
```

(Actually, even there I would like the x- and y minor tickmarks to be
the same *absolute*, not relative, length. But, that's a quibble.)

The FG documentation clearly states that [XY]ticklen = 1.0 is for the
major tickmarks only - so that behaviour is a definite bug.

The [XY]gridstyle docs do not explicitly state that only the major ticks
are affected, but that was the behaviour with DG so I expect that to be
the behaviour for FG.

You might want to consider adding some examples with these keywords set
to the "Plot Examples" section of the docs. Maybe I am doing something
wrong in FG-space, but after reading the docs I have nothing but my
DG-experience to go by.

cheers,

paulv

On 12/12/12 22:14, Chris Torrence wrote:
> Hi Paul,
>
> This sounds like a bug to me. I'll try to reproduce it, but if you
> could come up with a simple reproduce case, that would be very
> helpful.
>
> By the way, you said that the "ticklen/gridstyle is hopelessly busted

> in v8.2". What's wrong with it? If there is a bug, I'd like to fix
> that before IDL 8.2.2.
>
> Thanks!
>
> -Chris ExelisVIS

Subject: Re: Attempt to call undefined method: 'IDLITSYMBOL::IS3D'
Posted by Mark Piper on Fri, 14 Dec 2012 16:54:35 GMT
View Forum Message <> Reply to Message

On Thursday, December 13, 2012 7:23:33 AM UTC-7, Paul van Delst wrote:
>
> Re: the tickmark issue:
>
>
>
> In FG when I use [XY]ticklen = 1.0 and [XY]gridstyle=1 *all* the
>
> tickmarks are gridded, both major and minor. Try:
>
>
>
> IDL> p=plot(/test,layout=[1,3,1],$ &
>
> IDL> yticklen=1.0,ygridstyle=1,$ &
>
> IDL> xticklen=1.0,xgridstyle=1)
>
>
>
> My result:
>
> http://ftp.emc.ncep.noaa.gov/jcsda/CRTM/.plots/all_ticks_gri dded.png
>
>
>
> In DG only the major tickmarks were affected -- which is how the
>
> documentation describes the behaviour for both FG and DG.
>
>
>
> As you can imagine, having all the tickmarks gridded makes for a rather
>
> busy plot.
>

>
>
> Upon further reading of the docs I saw the [XY]subticklen keyword. I
>
> gave that a try, setting it to 0.05. But the minor tickmarks are *still*
>
> subject to the [XY]gridstyle keyword (I tend to use [XY]gridstyle=1,
>
> dotted, for this).
>
>
>
> Try:
>
> IDL> p=plot(/test,layout=[1,3,1],$ &
>
> IDL> yticklen=1.0,ygridstyle=1,ysubticklen=0.05,$ &
>
> IDL> xticklen=1.0,xgridstyle=1,xsubticklen=0.05)
>
>
>
> My result:
>
> ftp://ftp.emc.ncep.noaa.gov/jcsda/CRTM/.plots/using_subtickl en.png
>
>
>
> So, as you can see, what I have are uneven length minor tickmarks. And
>
> they are still dotted due the gridstyle keyword.
>
>
>
> What I want is the equivalent of the following DG:
>
> IDL> !p,multi=[0,1,3]
>
> IDL> plot, lindgen(100),$ &
>
> IDL> yticklen=1.0,ygridstyle=1,$ &
>
> IDL> xticklen=1.0,xgridstyle=1
>
>
>
> (Actually, even there I would like the x- and y minor tickmarks to be
>

> the same *absolute*, not relative, length. But, that's a quibble.)
>
>
>
> The FG documentation clearly states that [XY]ticklen = 1.0 is for the
>
> *major* tickmarks only - so that behaviour is a definite bug.
>
>
>
> The [XY]gridstyle docs do not explicitly state that only the major ticks
>
> are affected, but that was the behaviour with DG so I expect that to be
>
> the behaviour for FG.
>
>
>
> You might want to consider adding some examples with these keywords set
>
> to the "Plot Examples" section of the docs. Maybe I am doing something
>
> wrong in FG-space, but after reading the docs I have nothing but my
>
> DG-experience to go by.
>
>
>
> cheers,
>
>
>
> paulv
>

Hi Paul,

I'll log this tick behavior as a bug. For a temporary (but not really convenient) fix, try using CURRENT to overlay a second plot with the same data ranges:

```
  q1 = plot(/test,$
    yticklen=1.0, ygridstyle=1, ysubticklen=0.0, $
    xticklen=1.0, xgridstyle=1, xsubticklen=0.0)
  q2 = plot(/test, /nodata, /current)
```

Also, whenever you encounter a bug, please send it to Tech Support (support at exelisvis dot com); it's the best way to get this info to us.