Subject: Help needed in using object graphics
Posted by huangsu on Tue, 14 Jun 2016 05:53:37 GMT
View Forum Message <> Reply to Message

I encountered a problem using IDL object graphics. The source code is listed below. The program drawing a plot using IDLgrPlot using an object window IDLgrWindow. The plot displayed then disappeared immediately. I have to put a WAIT statement to confirm the plot is painted. Once the WAIT is finished, the plot disappeared. Can some one help me identify the problem?

```
PRO PLOTTEST
  x = FINDGEN(20)-10
  y = SIN(x)*10

  base = WIDGET_BASE(/ROW, TITLE= 'TEST', XSIZE=800, YSIZE=600)
  draw1 = WIDGET_WINDOW(base, XSIZE=800, YSIZE=600, GRAPHICS_LEVEL=2)
  WIDGET_CONTROL, base, /REALIZE
  WIDGET_CONTROL, draw1, GET_VALUE=owindow1
  view1 = OBJ_NEW('IDLgrView', VIEWPLANE_RECT=[-15,-15,30,30])
  model1 = OBJ_NEW('IDLgrModel')
  plot1 = OBJ_NEW('IDLgrPlot', x, y, COLOR=[255, 0, 0])
  model1->ADD, plot1
  view1->ADD, model1
  owindow1->DRAW, view1
  WAIT, 3
END
```

Platform: IDL 8.4.1, Linux Debian

Thanks for any reply.

Regards,

Subject: Re: Help needed in using object graphics
Posted by markb77 on Tue, 14 Jun 2016 18:24:14 GMT
View Forum Message <> Reply to Message

My guess would be that the objects you are creating exist while your procedure is running, and then when the procedure ends and they are destroyed by automatic garbage collection. If you run the same commands at the command prompt, the plot does not disappear..

Mark

Subject: Re: Help needed in using object graphics
Posted by Dick Jackson on Wed, 15 Jun 2016 21:47:43 GMT
View Forum Message <> Reply to Message

On Tuesday, 14 June 2016 11:24:16 UTC-7, superchromix  wrote:
> My guess would be that the objects you are creating exist while your procedure is running, and then when the procedure ends and they are destroyed by automatic garbage collection.  If you run the same commands at the command prompt, the plot does not disappear..
>
> Mark

Good thought, Mark… I bet you're right.

The problem happens for me on Windows 10 as well. Trying to find a solution (see notes in code)

```
PRO PLOTTEST
  x = FINDGEN(20)-10
  y = SIN(x)*10

  base = WIDGET_BASE(/ROW, TITLE= 'TEST', XSIZE=800, YSIZE=600)
  draw1 = WIDGET_WINDOW(base, XSIZE=800, YSIZE=600, GRAPHICS_LEVEL=2)
;;; Note: GRAPHICS_LEVEL is not documented for WIDGET_WINDOW
;  draw1 = WIDGET_DRAW(base, XSIZE=800, YSIZE=600, GRAPHICS_LEVEL=2)
  WIDGET_CONTROL, base, /REALIZE
  WIDGET_CONTROL, draw1, GET_VALUE=owindow1
  view1 = OBJ_NEW('IDLgrView', VIEWPLANE_RECT=[-15,-15,30,30])
  model1 = OBJ_NEW('IDLgrModel')
  plot1 = OBJ_NEW('IDLgrPlot', x, y, COLOR=[255, 0, 0])
  model1->ADD, plot1
  view1->ADD, model1
  owindow1->DRAW, view1
  WAIT, 1

  ; The problem is that the IDLgrPlot object disappears when this routine
  ; returns. Possible cause: garbage collection on objects that are no longer
  ; referenced when this routine returns.
  ; Try to make a persistent reference to the objects. This didn't work
  ; when WIDGET_WINDOW was used:
  WIDGET_CONTROL, base, SET_UVALUE={plot1:plot1, $
                                    model1:model1, $
                                    view1:view1, $
                                    owindow1:owindow1, $
                                    draw1:draw1, $
                                    base:base}
  HELP, base
  ;DEFSYSV, '!PLOTTEST_PLOT1', plot1

END
```

But look at this, all the objects are indeed referenced. I have it printing the Widget ID of the base:

```
IDL> plottest
BASE          LONG      =          25

IDL> widget_control,25, get_uvalue=uvalue
IDL> uvalue
{
   "PLOT1": <ObjHeapVar18227(IDLGRPLOT)>,
   "MODEL1": <ObjHeapVar18226(IDLGRMODEL)>,
   "VIEW1": <ObjHeapVar18224(IDLGRVIEW)>,
   "OWINDOW1": <ObjHeapVar16618(GRAPHICSWIN)>,
   "DRAW1": 26,
   "BASE": 25
}

IDL>  print,obj_valid([uvalue.plot1,uvalue.model1,uvalue.view1,uva lue.owindow1])
  1   1   1   1
```

Surprise! The objects are still valid! (not *destroyed* by GC at least) Looking more:

```
IDL> foreach obj, [uvalue.plot1,uvalue.model1,uvalue.view1,uvalue.owindow1] do print,obj
<ObjHeapVar18227(IDLGRPLOT)>
<ObjHeapVar18226(IDLGRMODEL)>
<ObjHeapVar18224(IDLGRVIEW)>
GRAPHICSWIN <16618>
  BACKGROUND_COLOR         = 255 255 255
  DIMENSIONS               = 800.000      600.000
  EVENT_HANDLER            = <NullObject>
  KEYBOARD_HANDLER         = ''
  MOUSE_DOWN_HANDLER       = ''
  MOUSE_MOTION_HANDLER     = ''
  MOUSE_UP_HANDLER         = ''
  MOUSE_WHEEL_HANDLER      = ''
  NAME                     = 'IDL'
  RESOLUTION               = 0.035277778     0.035277778
  SELECTION_CHANGE_HANDLER  = ''
  TITLE                    = <NullObject>
  WINDOW_TITLE             = ''
```

So, are they just hidden?

```
IDL> foreach obj, [uvalue.plot1,uvalue.model1,uvalue.view1,uvalue.owindow1] do begin &
obj.getproperty,hide=hide & print,hide & endforeach
          0
          0
          0
```

```
% GRAPHICSWIN: Unknown property: HIDE
% Execution halted at: $MAIN$
```

Nope… what else could be going on? Let's look at the easily visible properties:

```
IDL> foreach obj, [uvalue.plot1,uvalue.model1,uvalue.view1,uvalue.owindow1] do begin &
obj.getproperty,all=all & help,/str,all & endforeach
** Structure <aaf25b0>, 31 tags, length=224, data length=217, refs=1:
   ALPHA_CHANNEL   FLOAT           1.00000
   ANTIALIAS       LONG                 0
   CLIP_PLANES     LONG                -1
   COLOR           BYTE      Array[3]
   COMPONENT_VERSION
                   INT           851
   DEPTH_TEST_DISABLE
                   LONG                 0
   DEPTH_TEST_FUNCTION
                   LONG                 0
   DEPTH_WRITE_DISABLE
                   LONG                 0
   DOUBLE          LONG                 0
   HIDE            LONG                 0
   HISTOGRAM       LONG                 0
   LINESTYLE       LONG                 0
   MAX_VALUE       DOUBLE             NaN
   MIN_VALUE       DOUBLE             NaN
   NAME            STRING    ''
   NSUM            LONG                 1
   PALETTE         OBJREF    <NullObject>
   PARENT          OBJREF    <ObjHeapVar18226(IDLGRMODEL)>
   POLAR           LONG                 0
   SHADER          OBJREF    <NullObject>
   SYMBOL          OBJREF    <NullObject>
   THICK           FLOAT           1.00000
   USE_ZVALUE      LONG                 0
   VERT_COLORS     LONG                -1
   XCOORD_CONV     DOUBLE    Array[2]
   XRANGE          DOUBLE    Array[2]
   YCOORD_CONV     DOUBLE    Array[2]
   YRANGE          DOUBLE    Array[2]
   ZCOORD_CONV     DOUBLE    Array[2]
   ZRANGE          DOUBLE    Array[2]
   ZVALUE          DOUBLE          0.00000000
** Structure <b83af20>, 13 tags, length=192, data length=186, refs=1:
   ACTIVE_POSITION LONG                 0
   CLIP_PLANES     LONG                -1
   COMPONENT_VERSION
```

```
            INT         851
  DEPTH_TEST_DISABLE
            LONG          0
  DEPTH_TEST_FUNCTION
            LONG          0
  DEPTH_WRITE_DISABLE
            LONG          0
  HIDE          LONG          0
  LIGHTING      LONG          2
  NAME          STRING    ''
  PARENT        OBJREF    <ObjHeapVar18224(IDLGRVIEW)>
  RENDER_METHOD   LONG          0
  SELECT_TARGET   LONG          0
  TRANSFORM       DOUBLE    Array[4, 4]
** Structure <b6284d0>, 17 tags, length=144, data length=137, refs=1:
  CENTER        LONG          -1
  COLOR         BYTE      Array[3]
  COMPONENT_VERSION
            INT         851
  DEPTH_CUE       FLOAT     Array[2]
  DIMENSIONS      FLOAT     Array[2]
  DOUBLE        LONG          0
  EYE           DOUBLE        4.0000000
  HIDE          LONG          0
  LOCATION        FLOAT     Array[2]
  NAME          STRING    ''
  OBLIQUE         FLOAT     Array[2]
  PARENT        OBJREF    <NullObject>
  PROJECTION      LONG          1
  TRANSPARENT     LONG          0
  UNITS         LONG          0
  VIEWPLANE_RECT  DOUBLE    Array[4]
  ZCLIP         DOUBLE    Array[2]
% GRAPHICSWIN: Unknown property: ALL
% Execution halted at: $MAIN$
```

Well, I can't see anything in there. I notice if I double-click in the window, I get the property sheet for the view, and I can even change the background colour! But I can't seem to make the model or plot appear.

This seems like a bug to me, and I'd love to hear some feedback from Exelis/Harris. I discovered that the problem does not occur if we use WIDGET_DRAW instead of WIDGET_WINDOW, but that may not be an acceptable workaround.

Cheers,
-Dick

Dick Jackson Software Consulting Inc.