Subject: Re: show a use-defined object
Posted by David Fanning on Wed, 15 Jul 2009 12:32:10 GMT
View Forum Message <> Reply to Message

danyang writes:

> s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1,$
> POS=[x(i),y(i),z(i)], select_target=0) ;to define a ball
>
> oModel->Add, s
> XOBJVIEW, oModel

You don't tell us where you are trying to position
the object, and my guess (from working with object
graphics in general) is that you are positioning
the object outside the viewplane rectangle. In
any case, to just answer your question, this works:

s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1)
XOBJVIEW, s

Cheers,

David

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

Subject: Re: show a use-defined object
Posted by rtowler on Wed, 15 Jul 2009 16:18:35 GMT
View Forum Message <> Reply to Message

On Jul 15, 5:32 am, David Fanning <n...@dfanning.com> wrote:
> danyang writes:
>> s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1,$
>> POS=[x(i),y(i),z(i)], select_target=0) ;to define a ball
>
>> oModel->Add, s
>> XOBJVIEW, oModel
>
> You don't tell us where you are trying to position
> the object, and my guess (from working with object
> graphics in general) is that you are positioning
> the object outside the viewplane rectangle. In

> any case, to just answer your question, this works:
>
> s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1)
> XOBJVIEW, s
>

XOBJVIEW will automatically determine the extents of all objects
contained in the model hierarchy and it sets up the view such that it
is centered on these extents. So usually position is not an issue.

```
oModel = obj_new('IDLgrModel')
s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1, $
   POS=[4,13,10], select_target=0)
oModel->add, s
t = obj_new('orb',color=[250,250,250],radius=0.25,shading=1, $
   POS=[14,23,3], select_target=0)
oModel->add, t

XOBJVIEW, oModel
```

I suspect there is something else going on. Post more of your code.

-Rick

Subject: Re: show a use-defined object
Posted by danyang on Thu, 16 Jul 2009 09:16:35 GMT
View Forum Message <> Reply to Message

On Jul 15, 2:32 pm, David Fanning <n...@dfanning.com> wrote:
> danyang writes:
>> s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1,$
>> POS=[x(i),y(i),z(i)], select_target=0) ;to define a ball
>
>> oModel->Add, s
>> XOBJVIEW, oModel
>
> You don't tell us where you are trying to position
> the object, and my guess (from working with object
> graphics in general) is that you are positioning
> the object outside the viewplane rectangle. In
> any case, to just answer your question, this works:
>
> s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1)
> XOBJVIEW, s
>
> Cheers,

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

Thanks for your kindness.

In fact, I have more than 1000 objects. I can get their positions from an ASCII file. Therefore, I used a 'for' loop to put them into a show window.
My objects were spheres. That's why there was an 'orb' in my code. In this step, I don't have any more questions.

But now my objects should be changed to some kind of air ballons, which has hard shell with a radius of R1 and air-kern with the radius of R2(R1>R2). Some of them should be cut(e.g. instead of the whole 3D region just to show part of it) , in order to show their hollow kern.

Is it possible to make my aim come true with 'XOBJVIEW'?

Thanks,
Danyang

Subject: Re: show a use-defined object
Posted by danyang on Thu, 16 Jul 2009 09:24:10 GMT
View Forum Message <> Reply to Message

On Jul 15, 6:18 pm, rtowler <rtow...@gmail.com> wrote:
> On Jul 15, 5:32 am, David Fanning <n...@dfanning.com> wrote:
>
>
>
>> danyang writes:
>>> s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1,$
>>> POS=[x(i),y(i),z(i)], select_target=0) ;to define a ball
>
>>> oModel->Add, s
>>> XOBJVIEW, oModel
>
>> You don't tell us where you are trying to position
>> the object, and my guess (from working with object
>> graphics in general) is that you are positioning

>> the object outside the viewplane rectangle. In
>> any case, to just answer your question, this works:
>
>> s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1)
>> XOBJVIEW, s
>
> XOBJVIEW will automatically determine the extents of all objects
> contained in the model hierarchy and it sets up the view such that it
> is centered on these extents. So usually position is not an issue.
>
> oModel = obj_new('IDLgrModel')
> s = obj_new('orb',color=[250,250,250],radius=0.25,shading=1, $
> POS=[4,13,10], select_target=0)
> oModel->add, s
> t = obj_new('orb',color=[250,250,250],radius=0.25,shading=1, $
> POS=[14,23,3], select_target=0)
> oModel->add, t
>
> XOBJVIEW, oModel
>
> I suspect there is something else going on. Post more of your code.
>
> -Rick

Thank you, Rick.
Yes, your suspection is right. As I discribed in the last post, my
question is how to show over 1000 air ballons in one 3D region.
Furthermore, some of ballons should be cut by changing the show
region, in order to show their hollow kern.
Is XOBJVIEW a better solution?

Thanks,
Danyang

Subject: Re: show a use-defined object
Posted by David Fanning on Thu, 16 Jul 2009 12:10:06 GMT
View Forum Message <> Reply to Message

danyang writes:

> n fact, I have more than 1000 objects. I can get their positions from
> an ASCII file. Therefore, I used a 'for' loop to put them into a show
> window.
> My objects were spheres. That's why there was an 'orb' in my code. In
> this step, I don't have any more questions.
>
> But now my objects should be changed to some kind of air ballons,

> which has hard shell with a radius of R1 and air-kern with the radius
> of R2(R1>R2). Some of them should be cut(e.g. instead of the whole 3D
> region just to show part of it) , in order to show their hollow
> kern.
>
> Is it possible to make my aim come true with 'XOBJVIEW'?

XObjView doesn't have anything to do with your aim. It
is just one way you can display your graphics hierarchy.
There are many others. You need to concentrate on the
hierarchy.

Cheers,

David

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

Subject: Re: show a use-defined object
Posted by rtowler on Thu, 16 Jul 2009 16:24:45 GMT
View Forum Message <> Reply to Message

On Jul 16, 5:10 am, David Fanning wrote:
> danyang writes:
>> But now my objects should be changed to some kind of air ballons,
>> which has hard shell with a radius of R1 and air-kern with the radius
>> of R2(R1>R2). Some of them should be cut(e.g. instead of the whole 3D
>> region just to show part of it) , in order to show their hollow
>> kern.
>
>> Is it possible to make my aim come true with 'XOBJVIEW'?
>
> XObjView doesn't have anything to do with your aim. It
> is just one way you can display your graphics hierarchy.
> There are many others. You need to concentrate on the
> hierarchy.

I suppose it depends on what aim you are talking about ;)

But David's point is a good one. You've created some orbs and (I am
assuming) displayed them. Now you want to display a custom object and
you have to "build" it. The best way to approach this would be to

create your own object by subclassing IDLgrModel. You'll have to come up with the meshing routine yourself but you'll probably want to look at orb__define.pro as a place to start. In fact, orb__define.pro is also an example of how to subclass IDLgrModel so you may want to just copy it and start modifying it for your needs. You'll want to look at the MESH_CLIP function too. I don't really understand what you're after, but maybe the init method of your new balloon class would take a inner and outer radius, position (x,y,z) and an optional clipping plane and it would create your object. Then you would use this new class just like you used orb initially.

Once you get to this point, then you can worry about moving from XOBJVIEW to a more flexible display technique.

-Rick
