
Subject: Re: Isurface, inverting the axis without changing default lighting direction
Posted by [penteado](#) on Wed, 05 May 2010 01:49:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

On May 4, 10:06 pm, cameron bowles <cameronbowle...@gmail.com> wrote:

```
> Thankyou so much for your tips Ken, they really helped. In the end I
> got it to look kind of OK with this code;
>
> void = ITGETCURRENT(TOOL=itool_obj)
> lights_id = itool_obj -> FindIdentifiers('*LIGHTS', /
> VISUALIZATIONS) ;Get lights ID
> lights_obj = itool_obj ->
> GetByIdentifier(lights_id) ;Get lights object
> reference
> lights_obj -> SetProperty, INTENSITY = 0.7, DISTANCE =
> -20 ;Set intensity and distance (z axis)
> itool_obj -> RefreshCurrentWindow
>
> However I would really like to be able to set the location and
> direction of the directional light, this should be possible with;
>
> lights_obj -> SetProperty, DIRECTION = [x,y,z], DISTANCE = [x,y,z]
>
> But I found that no matter what setting I had for x/y/z it wouldnt
> affect the lighting. Has anyone done this manually? I roughly want to
> have the light coming with a vector of [1,1,-1] from a position
> direction of [-1,-1,1]. If anyone knows the tricks to get that working
> I would appreciate it.
```

I think your problem is the object you are picking to edit. By default, isurface makes two lights, one ambient (isotropic, I guess), and one directional. For instance,

```
IDL> ids=itool_obj->findidentifiers('*LIGHT*',/visualization)
IDL> for i=0,n_elements(ids)-1 do print,ids[i]
/TOOLS/SURFACE TOOL/WINDOW/VIEW_1/VISUALIZATION LAYER/LIGHTS
/TOOLS/SURFACE TOOL/WINDOW/VIEW_1/VISUALIZATION LAYER/LIGHTS/LIGHT
/TOOLS/SURFACE TOOL/WINDOW/VIEW_1/VISUALIZATION LAYER/LIGHTS/LIGHT_1
```

You are selecting the lights object, but the properties you want to edit are for the directional object, which in this case is LIGHT_1:

```
IDL> light=itool_obj->getbyidentifier('/TOOLS/SURFACE TOOL/WINDOW/
VIEW_1/VISUALIZATION LAYER/LIGHTS/LIGHT')
IDL> light->getproperty,name=name & print,name
Ambient Light
IDL> light=itool_obj->getbyidentifier('/TOOLS/SURFACE TOOL/WINDOW/
VIEW_1/VISUALIZATION LAYER/LIGHTS/LIGHT_1')
```

```
IDL> light->getproperty,name=name & print,name  
Directional Light
```

With the object for the directional light, I can move and point it around with

```
light->setproperty,location=[1,1,-1],direction=[-1,-1,1]
```

Note that it may take a

```
itool_obj->commitactions
```

for the image to be updated after you change the objects.
