Subject: Re: coordinates in threedimensional object graphics ?
Posted by Rick Towler on Mon, 24 Oct 2005 18:07:39 GMT
View Forum Message <> Reply to Message

Klemens Barfus wrote:
> Hi Rick,
>
> its exactly the point you mentioned, that the object rotates out of the
> view.
> The volumes I want to visualize should fit all in the same way in the
> view though they have different ranges in x, y and z direction.
> When I rotate them around z and y axis I have to fit the viewplane_rect
> to get for example a certain frame around the object.

O.K. This is easy :)

Rotations are applied about the origin to the world coordinate system.
Consider an object with extents 1 by 1 by 1, centered at [5,0,0]. If
you rotate the world coordinate system 90 degs about the Z axis, your
object will now be at [0,5,0]. If you had a view 2x2 units centered at
[5,0,0] the object would be out of view after the rotation.

Translate that same object to the world origin. Now when you apply the
rotation, the object doesn't move. It starts at [0,0,0] and ends at
[0,0,0]. If you had your view centered about the origin, you would see
the object both before and after the rotation.

The general solution involves finding the center and extents of your
objects then translating the objects such that they are centered about
the origin. RSI provides the get_obj_range function (found in
C:\Program Files\RSI\IDL61\lib\utilities) which does just this. For
bonus points you can use the extents to determine the width and height
of your view.

If you have multiple objects you need to rotate, the approach is to
translate them to the origin, rotate, then translate them back.

If you are building a viewing system you may want to consider using my
camera object. If you're just building a simple viewer it will be
overkill, but if you plan on extending it using the camera will give you
a lot more flexibility.

You can download it here:

 http://www.acoustics.washington.edu/~towler/RHTgrCamera.html

-Rick

>
> Rick Towler wrote:
>
>>
>>> I play around with object graphics and try to build a routine which
>>> is able to visualize 3d objects of different sizes in th same way.
>>
>>
>>
>> XOBJVIEW does this. I believe there is an object-ified version
>> available from the RSI user contrib library which probably adds a good
>> bit of flexibility.
>>
>>> When I define the viewplane_rect and the zclip of my view, everything
>>> works fine before I rotate the view. Then the coordinate system is
>>> not the coordinate system of the defined axes anymore.
>>
>>
>>
>> I'm not sure exactly what your issues are. When you rotate the object
>> does it rotate out of the view?
>>
>>> Is there an easy way to calculate the coordinates for viewplane_rect
>>> and zclip from the coordinates of the rotated axes ?
>>
>>
>>
>> While there probably is a way to do this, you'd be working backwards.
>> It is best to think of the view as fixed and (unless you are
>> "zooming") the viewplane_rect as fixed. You operate on the objects
>> contained within the view, manipulating their transforms to yield the
>> desired results. So your question is regarding transforming your
>> models in some way. Maybe you can add some details.
>>
>>
>> -Rick