PREFACE

This is not intended to be a tutorial on how to use Blender.
For that, please refer to http://www.blender.org/support/
Another learning resource can be found here http://www.blenderguru.com/
Yet Another good site for learning http://cgcookie.com/blender/

This tutorial was written specifically for users of STK (Satellite Tool Kit) www.agi.com
It is a guide on setting up converted CAD files so that they can be used in STK.
While not necessary, a general working knowledge of 3D modeling and the terms used will be helpful as you work through this tutorial. While you can extract and apply the techniques described in each section, I recommend that you work from beginning to end.

CONTENTS

PREPARATION AND EXPORT OF THE CAD FILE

Most CAD files are cumbersome in OpenGL because of small parts like bolts, bolt holes, screws, nuts and washers that jack up the polygon count. These and other internal parts like circuit boards and gyroscopes should be removed in the host CAD application unless required, before export to STL.

Note: *If the exported STL file is larger than 10MB, or about 90,000 polygons, then it should be broken up into at least a couple of assemblies, depending on the complexity, or it will not load into STK.*
The reason for this is that the XML reading library libXML, used by the COLLADA DOM (which STK uses to ingest Collada), has a limit of 10 million characters within any xml tag.
I'm not saying that you can't load a file any larger than 10MB into STK, only that each component/assemble/node within the Collada file cannot be any larger than 10MB. You can have as many components/assemblies/nodes as are needed.

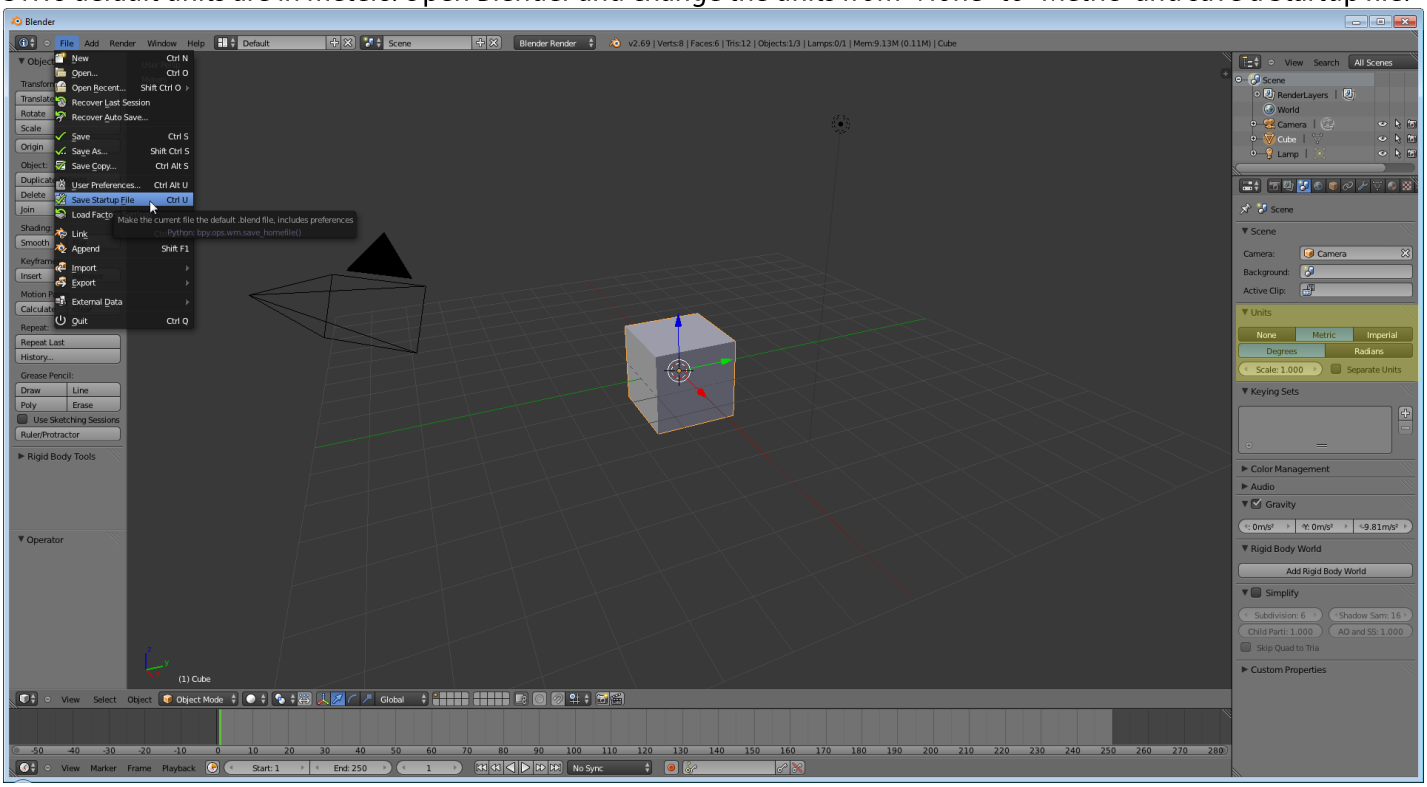
Once done, export the main object as an STL file.
Any part that needs to articulate should be exported as separate .STL files.

I've created a generic satellite for use with this tutorial and it has two parts, the bus and the arrays.

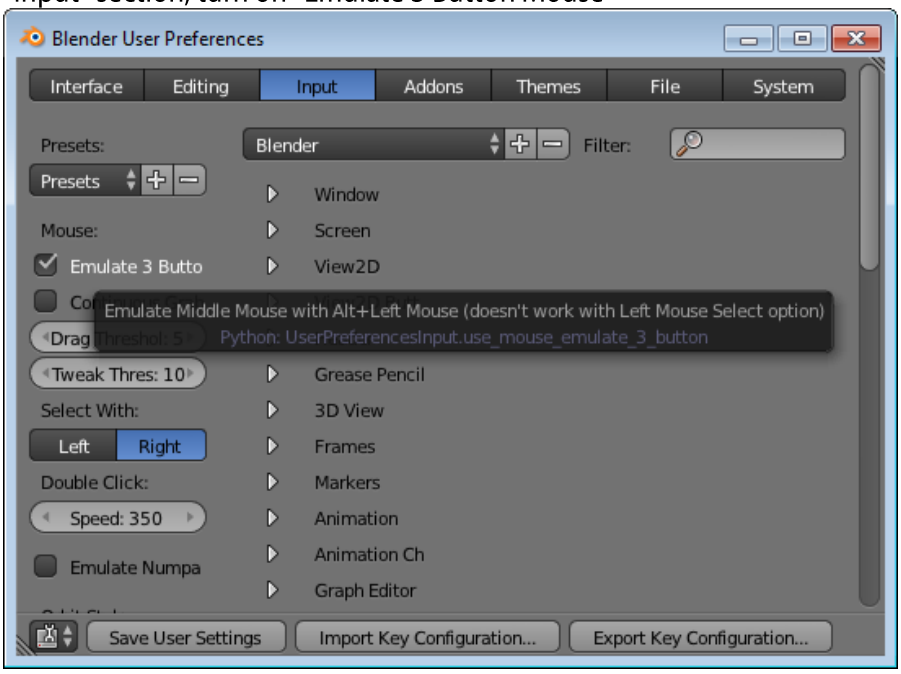
BLENDER SETUP

If you don't have it yet, Blender can be obtained from here: http://www.blender.org/download/

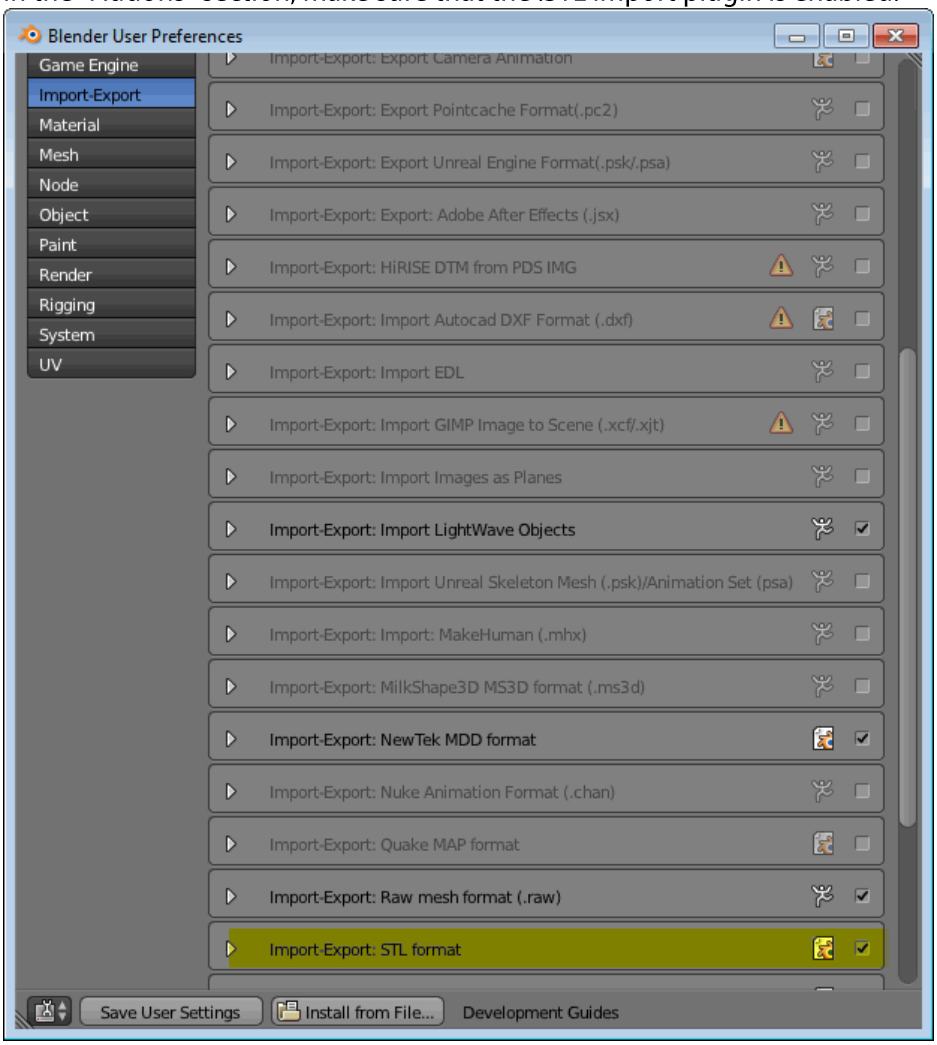
STK's default units are in meters. Open Blender and change the units from "None" to "Metric" and save a Startup file.



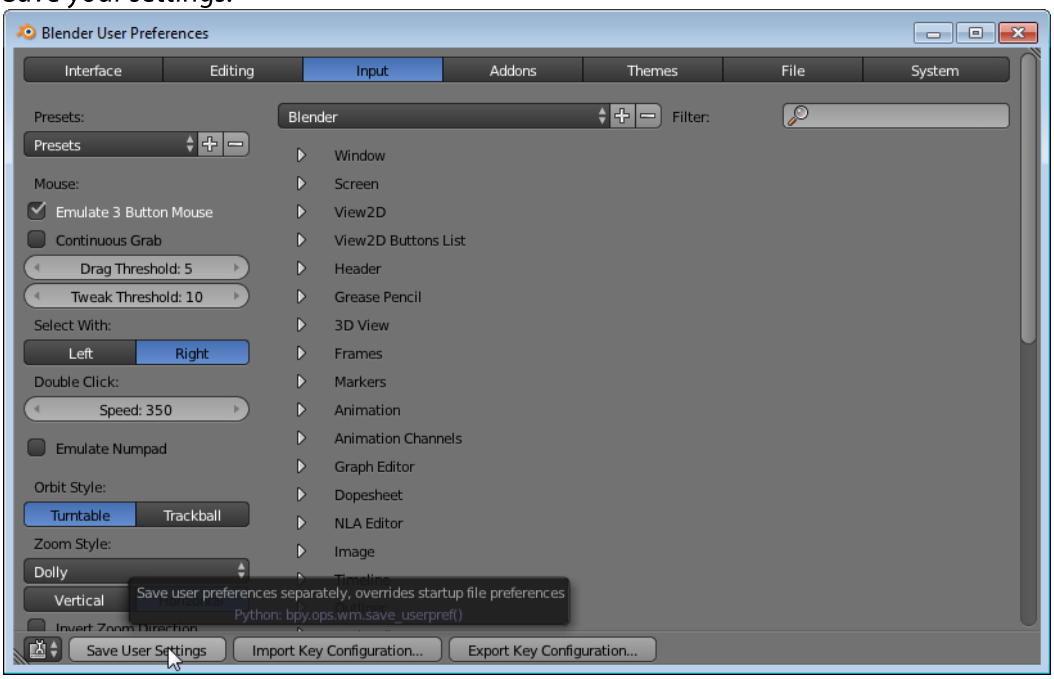
By default, rotating around the 3D scene in Blender is done with the middle mouse button. But, I prefer to rotate around the screen using Alt-LMB because it is that way in modo, Maya, Unity, LightWave 3D and other DCC modeling packages. If you would like to change from the default, go to the “File_User Preferences” pull down menu and in the “Input” section; turn on “Emulate 3 Button Mouse”



In the “Addons” section, make sure that the .STL import plugin is enabled.



Save your settings.

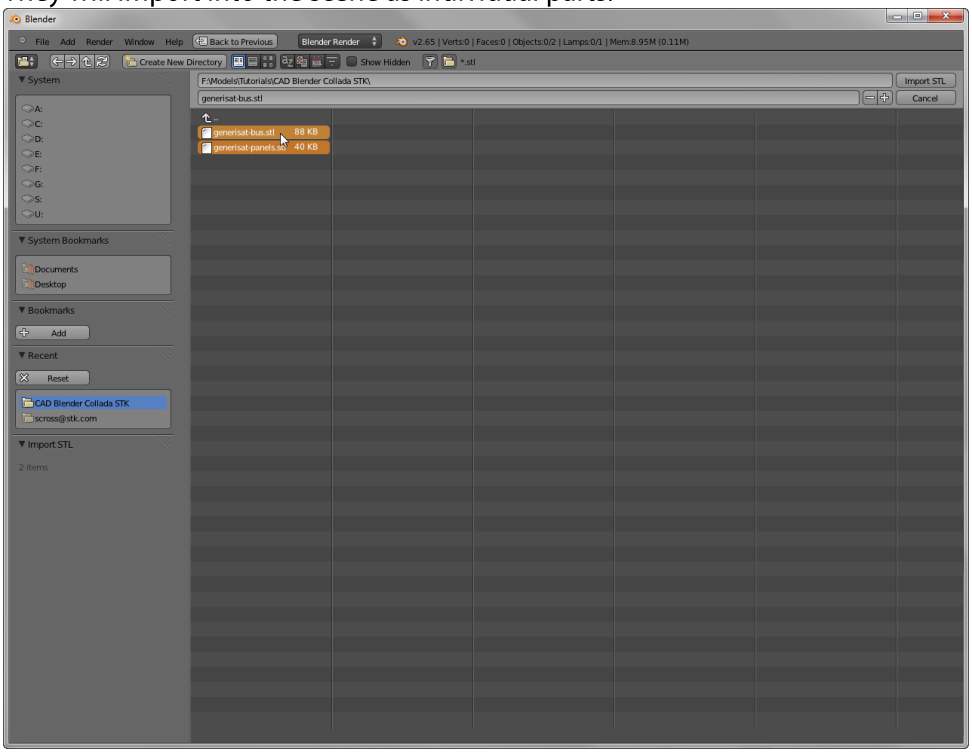


Delete the default cube if it is there.

IMPORTING THE FILES

We will run through the example model first.
Import the STL files included with this tutorial into Blender by going to the “File_Import_Stl”.
Hint: *You can multi select by holding down the Shift key.*

They will import into the scene as individual parts.

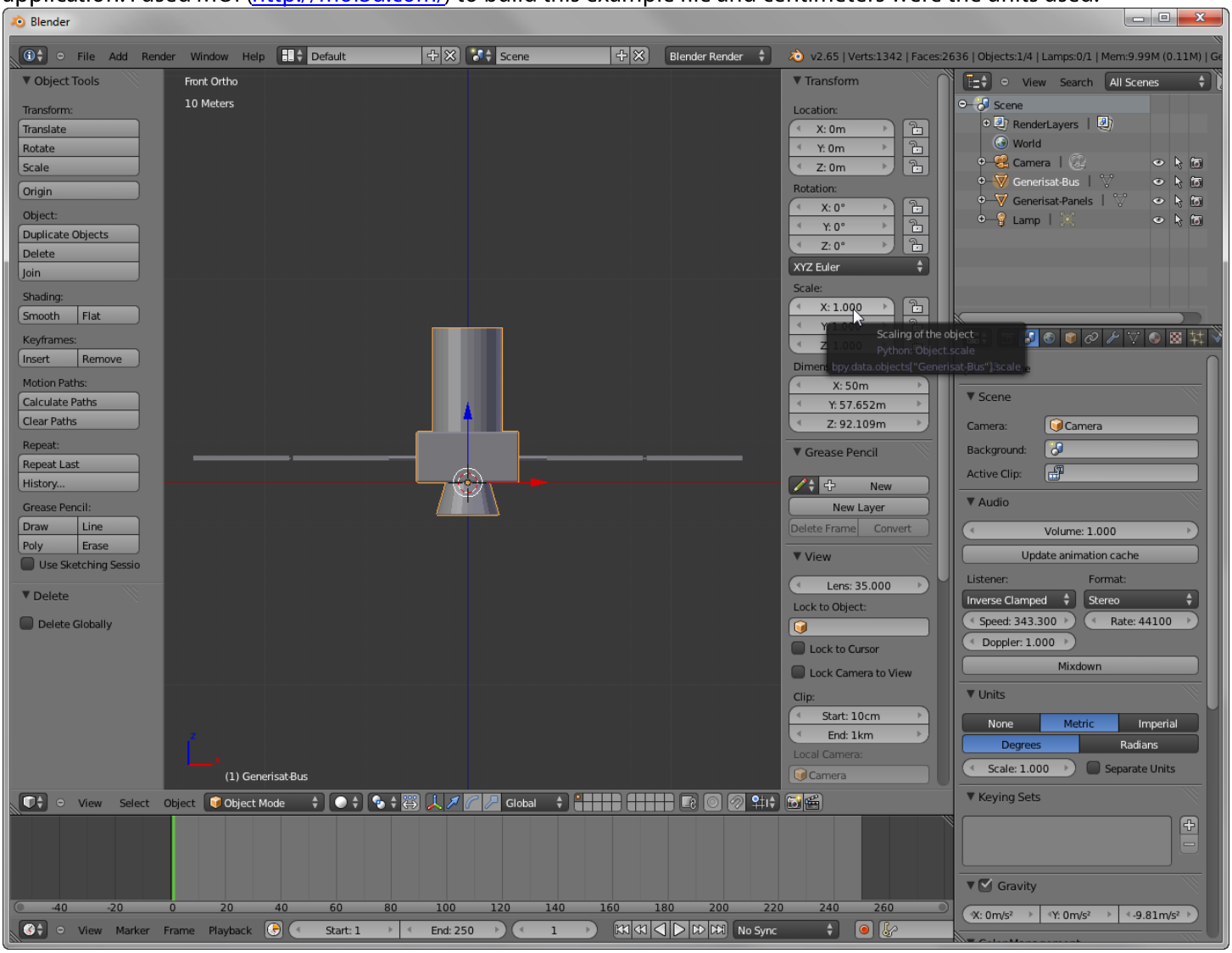


SCALING THE OBJECTS

Units from the host CAD application will be converted to Blender units, which most likely are not the same.
The model will need to be re-scaled.
Select a part with the "right mouse" in the 3D window, or by using the "Outliner".

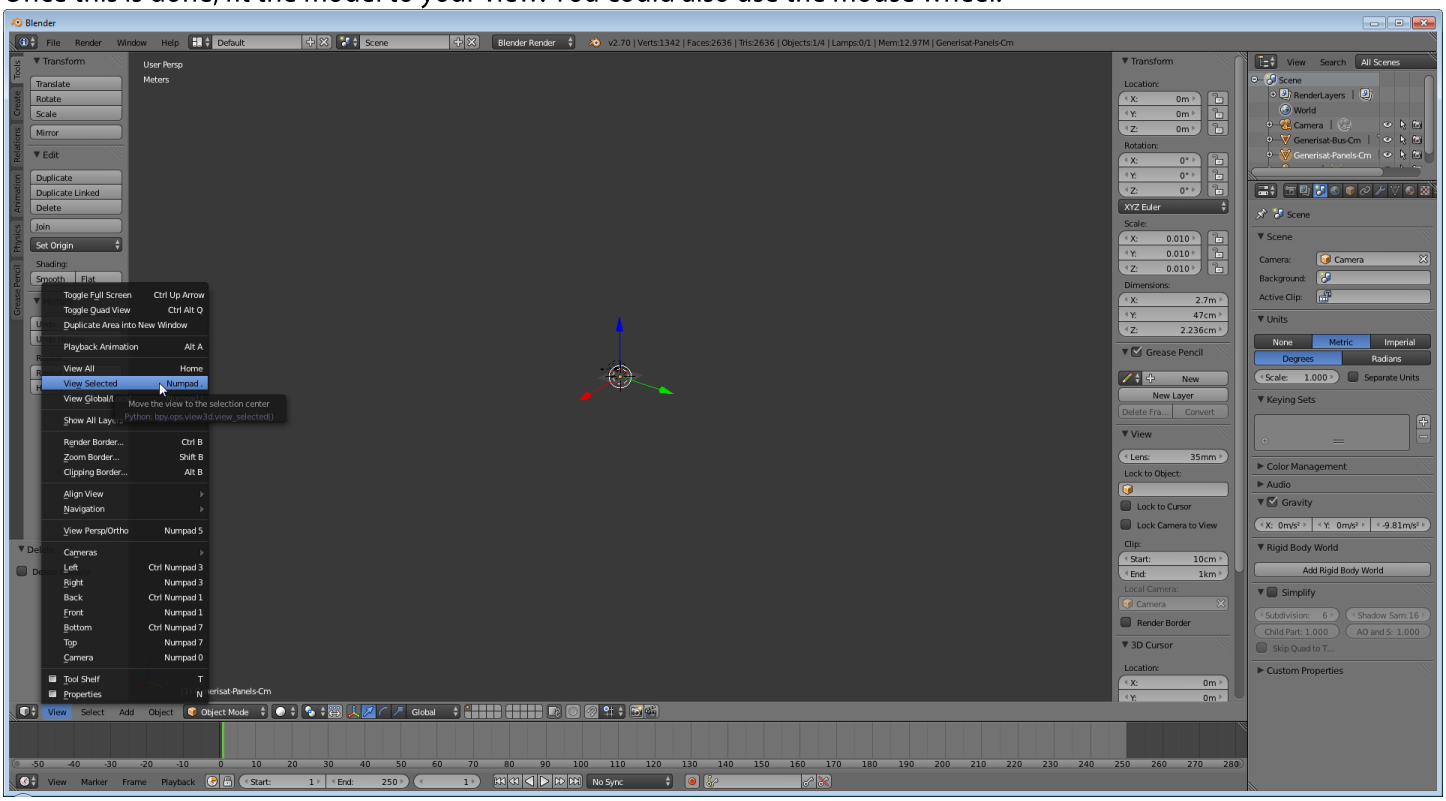
Press "n" to bring up the "Numeric Panel"
Hint: *Place your mouse over the OpenGl window.*

In the "Transform" section, scale the model down, depending on which units where used in in the host CAD application. I used MOI (http://moi3d.com/) to build this example file and centimeters were the units used.

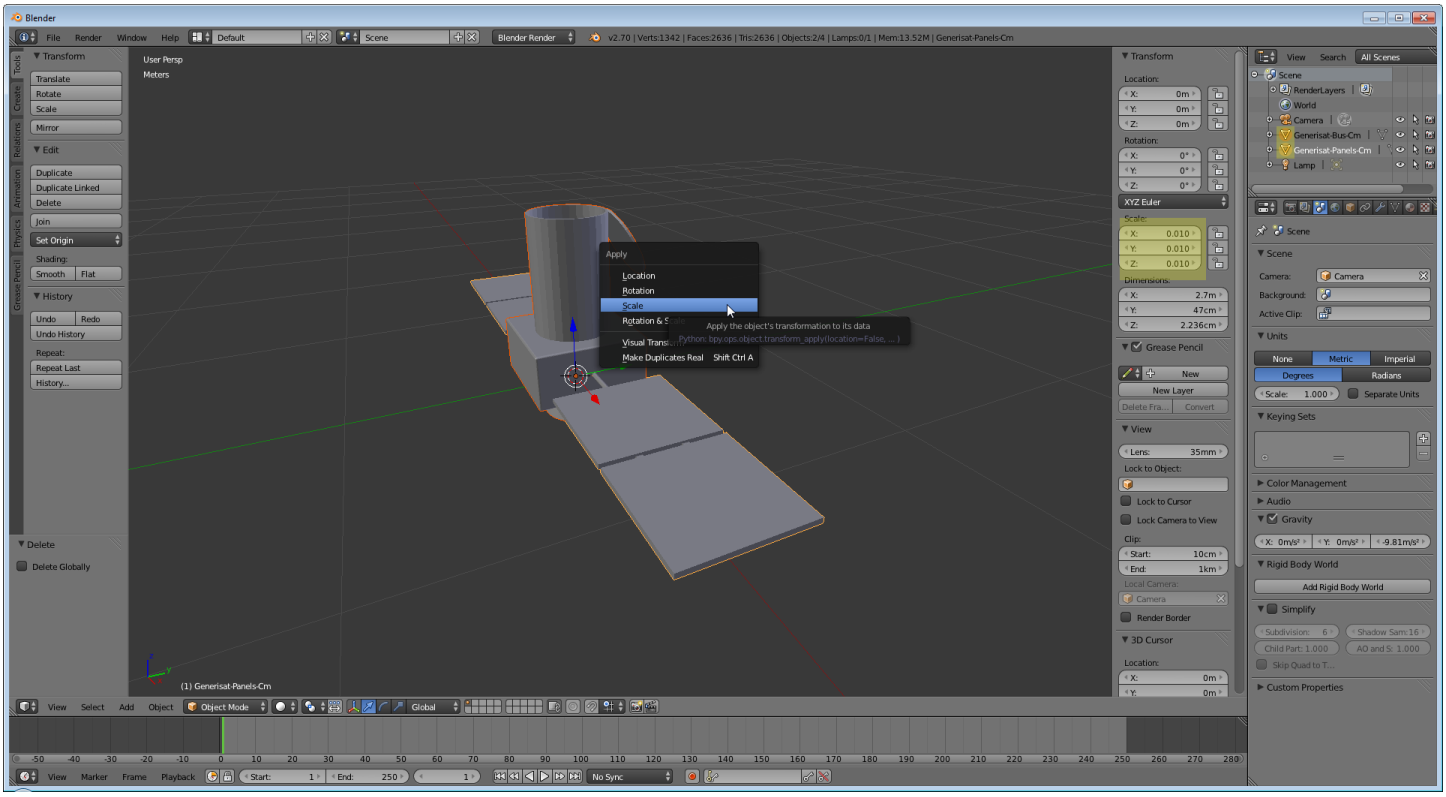


Hint: If the CAD file was built in
Millimeters; scale on all three axes by 0.001
Centimeters; scale on all three axes by 0.01
Decimeters; scale on all three axes by 0.1
Inches; scale on all three axes by 0.0254
Feet; scale on all three axes by 0.3048

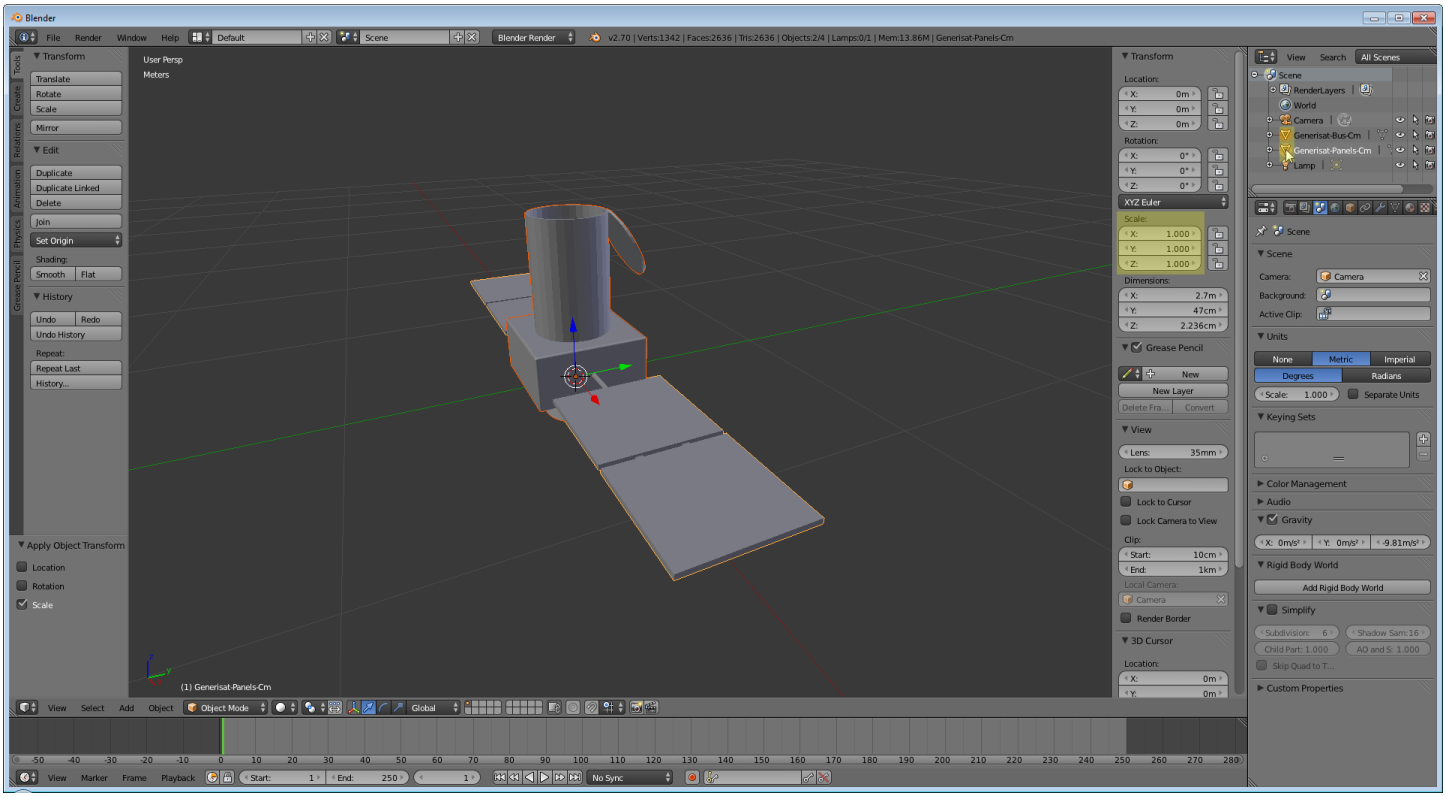
Once this is done, fit the model to your view. You could also use the mouse wheel.



Shift select all of the imported parts and press "Ctrl-a" and select “Scale” to "Reset Scale Transformation"
Before



After

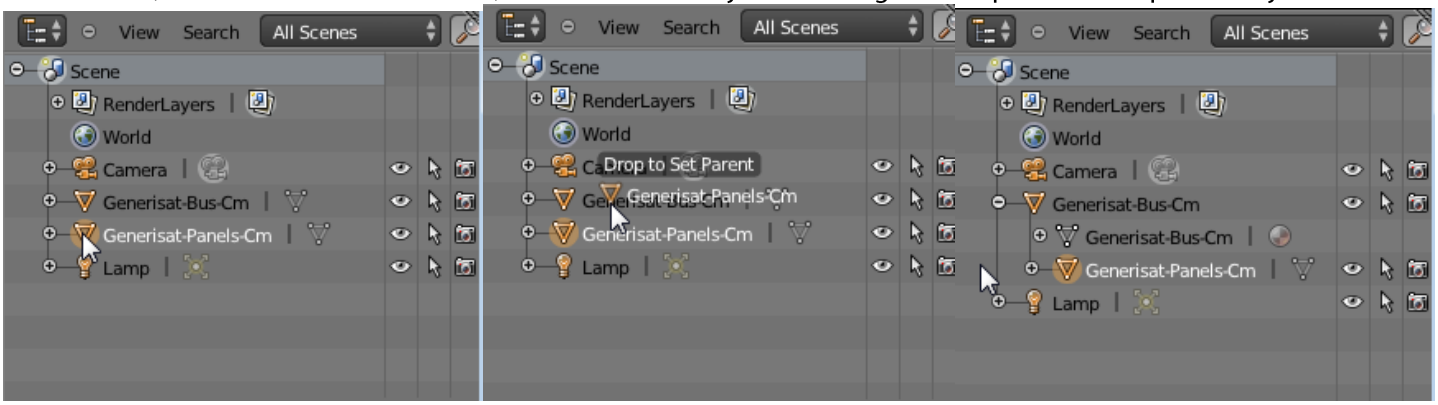


This makes your scaling changes permanent. Do this for each part you imported.
Save a Blender file as generisat.blend

OBJECT PARENTING

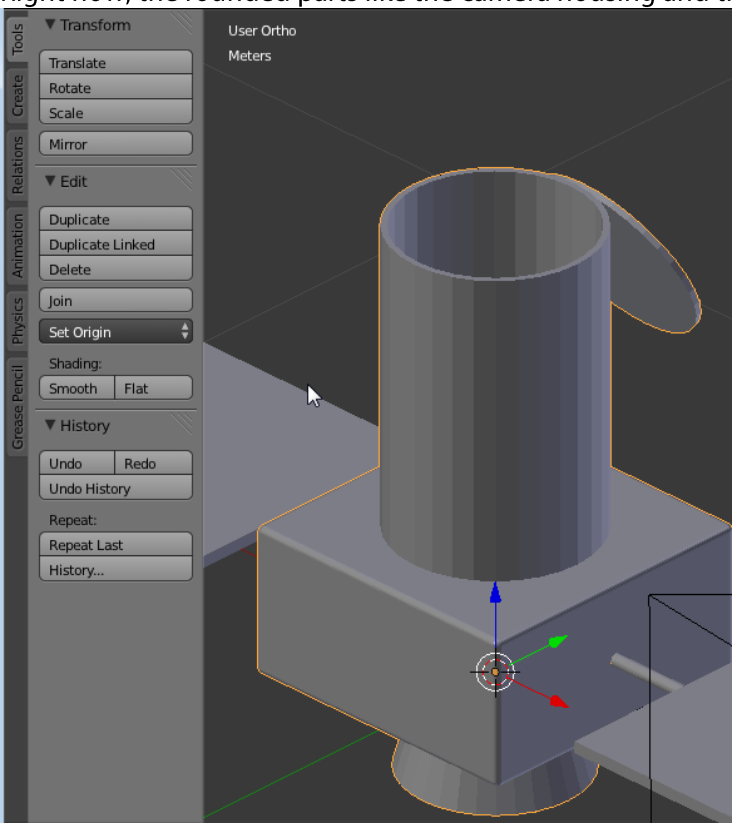
Object parenting is important if you want your models parts to articulate correctly in STK. If you don't parent, when you rotate the model via articulation, the rest of the model will remain stationary.

In "Outliner", with the left mouse button, select the child object and drag and drop it onto the parent object.



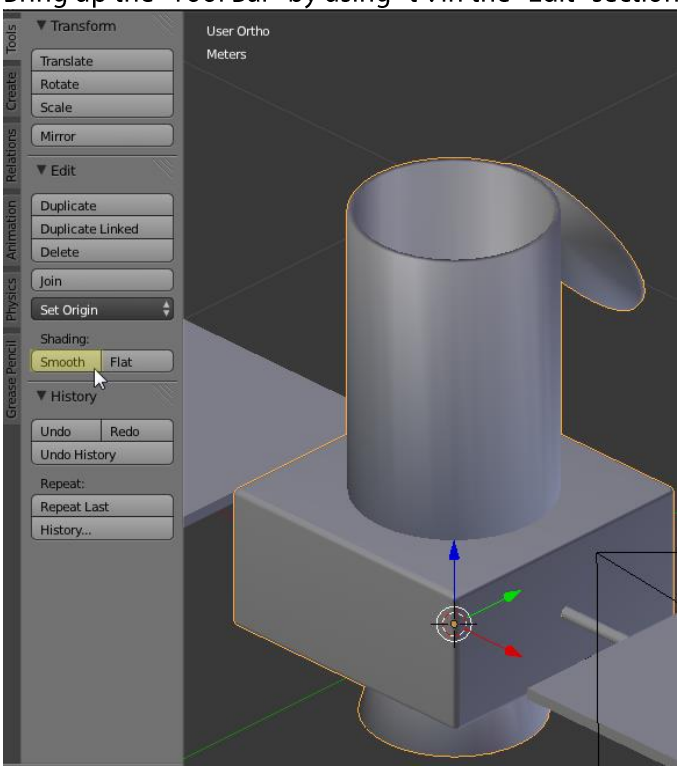
SMOOTH SHADING

Right now, the rounded parts like the camera housing and the engine bell housing are "Flat Shaded" or faceted.



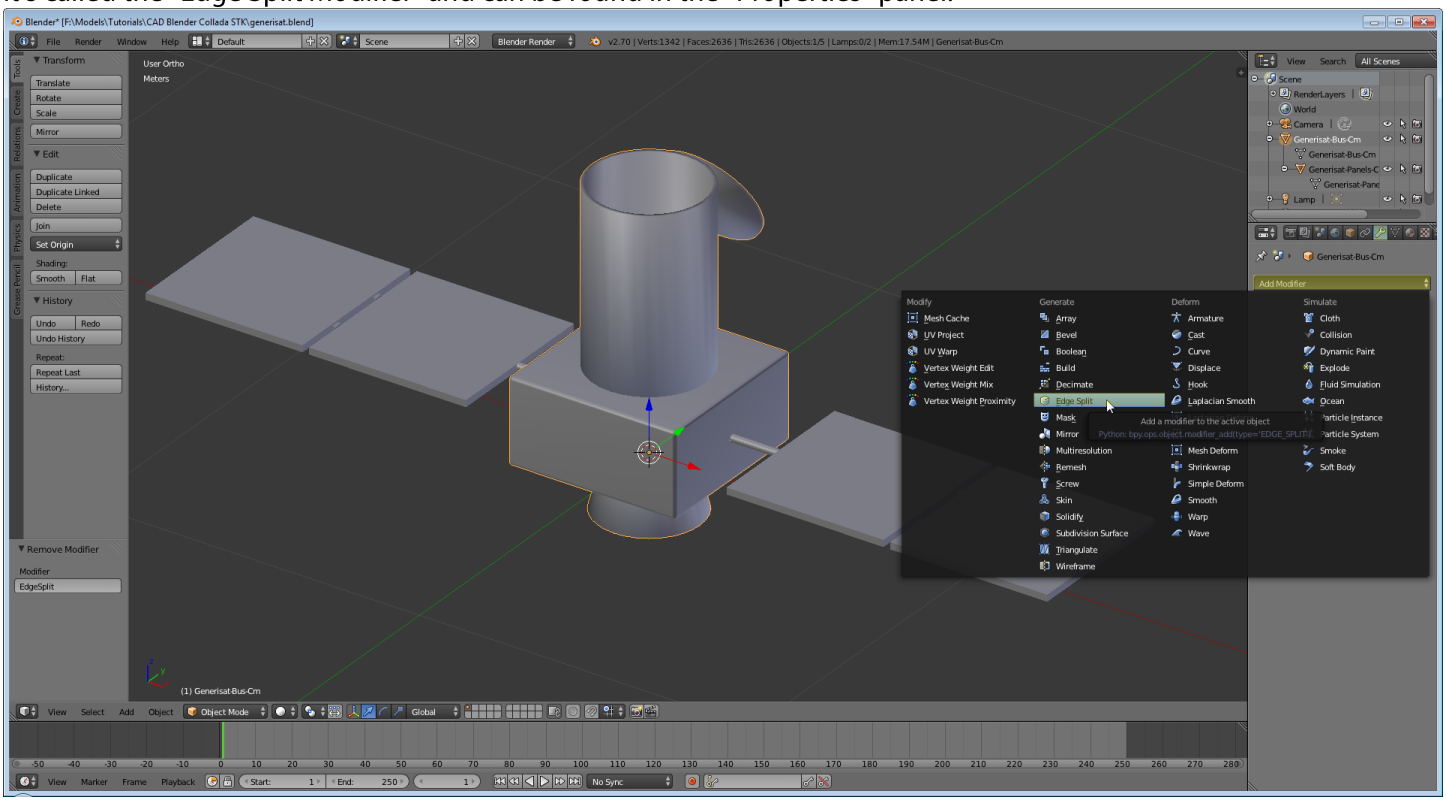
We can smooth out the transition between the polygons.
Select the "Generisat-Bus-Cm" layer.

Bring up the “Tool Bar” by using “t”. In the “Edit” section of the tool bar, select “Smooth” shading.



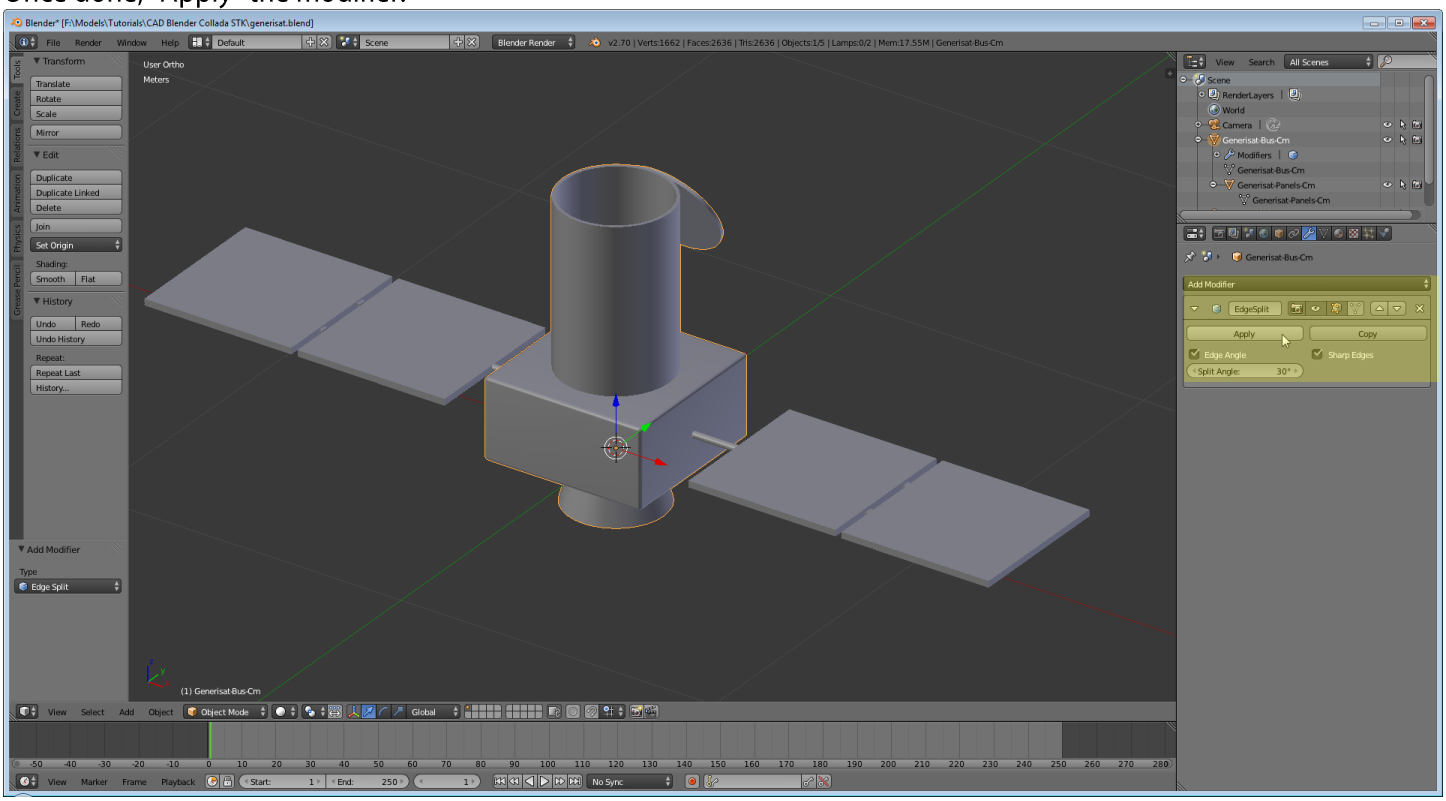
This still isn’t quite right, because the smoothing is being applied globally and ignores areas where flat and rounded regions meet. These regions need to be split apart from each other in order to be correctly shaded.

There is a “Modifier” that will help us separate the shading between smooth and flat areas of our geometry.
It’s called the “Edge Split Modifier” and can be found in the “Properties” panel.



Once you add the modifier, you can set the angle to what suits your needs.
For ours, the default of 30 degrees will suffice.

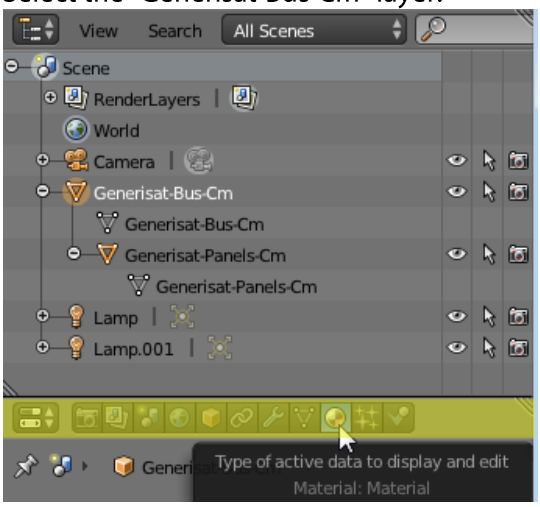
Once done, “Apply” the modifier.



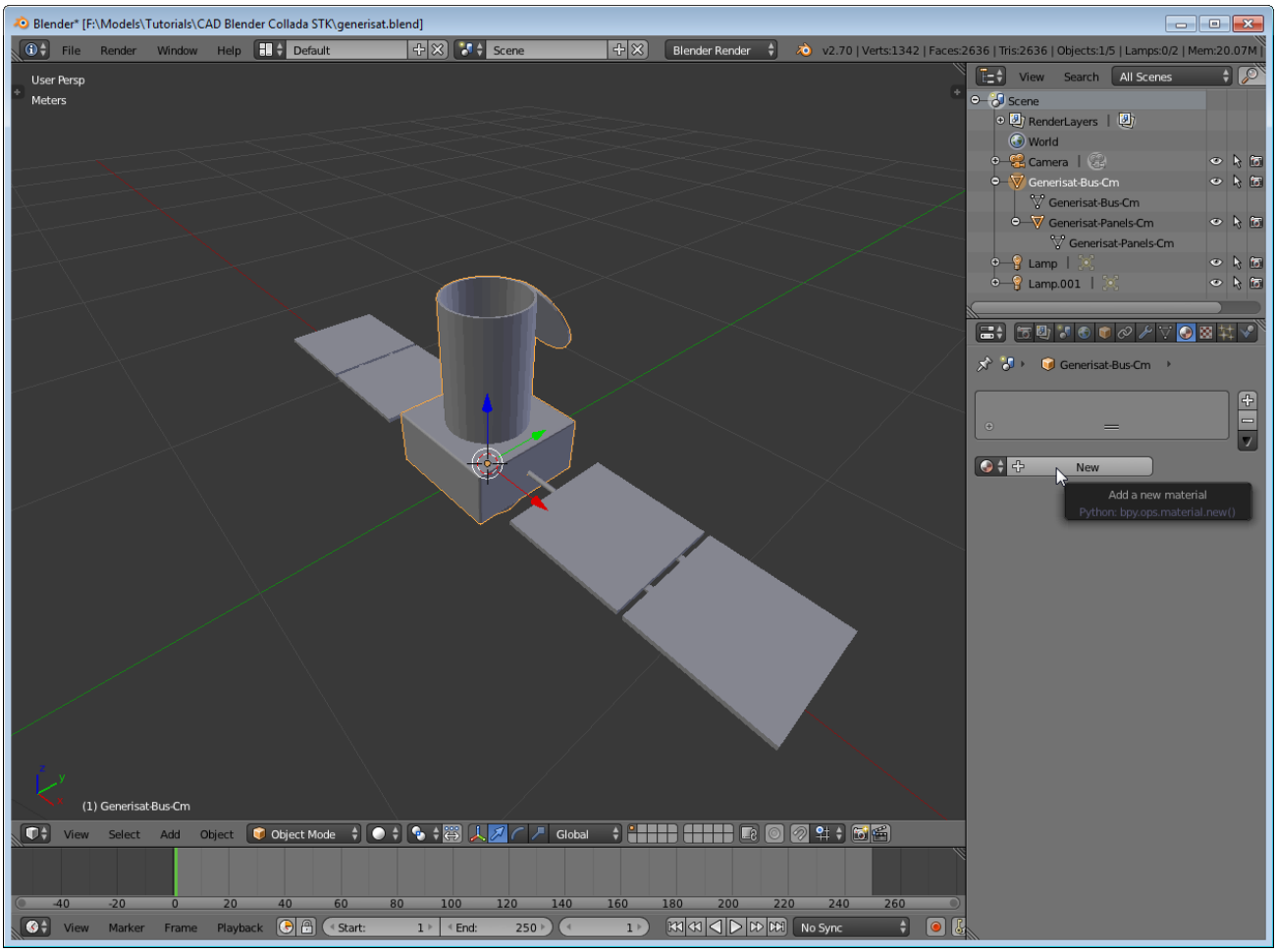
Note: *This smooth shading section was added after the completion of version 5 of this document and won’t be seen in any of the following screen shots. The “Edge Split Modifier” can be applied at any step of this tutorial.*

ASSIGNING MATERIALS

We are going to use the “Properties” panel to add a new material to the bus.
Select the “Generisat-Bus-Cm” layer.



In the “Properties” section just below the “Outliner”, select “Material” and then “New”, then re-name the material.
Give it the name “Bus”



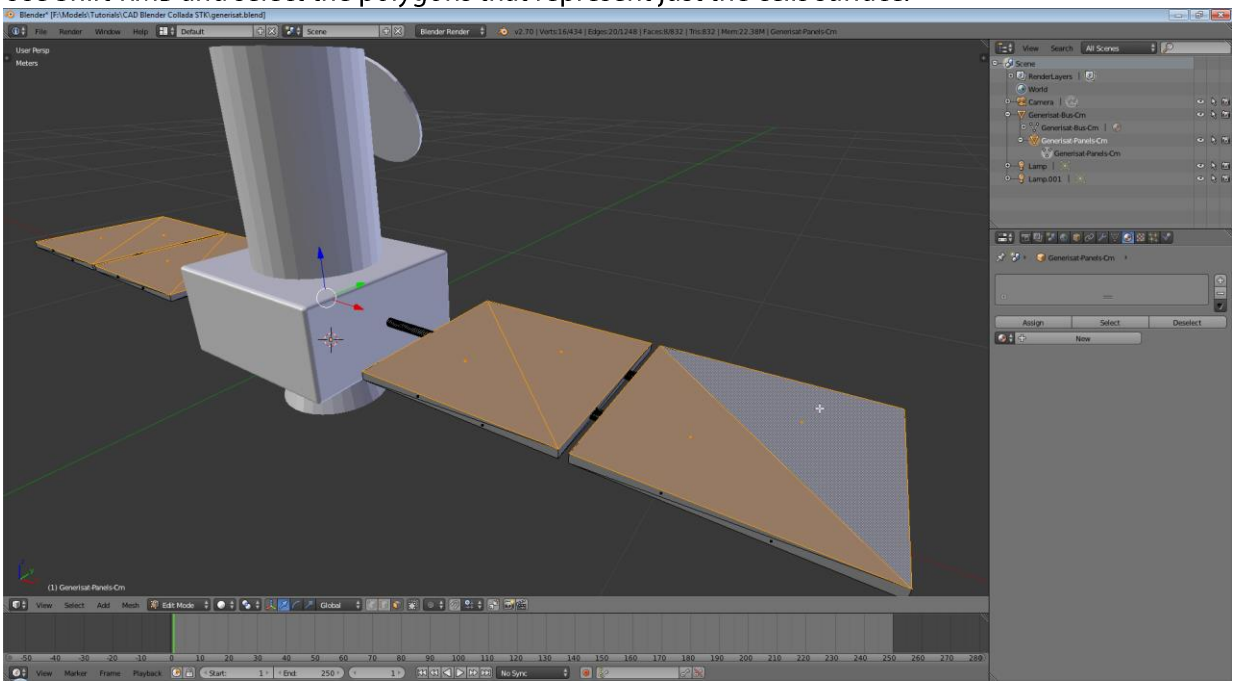
Set the Diffuse to white and leave the Intensity, Specular and Hardness at their defaults for a slightly shinny surface.
Note: *There is currently a bug in Modeler 10 having to do with Collada models that have UV's, but no texture assigned. No matter what diffuse color is assigned, the displayed color is blue.*
This bug will be fixed in a future release of STK 10

SOLAR PANEL GROUPS

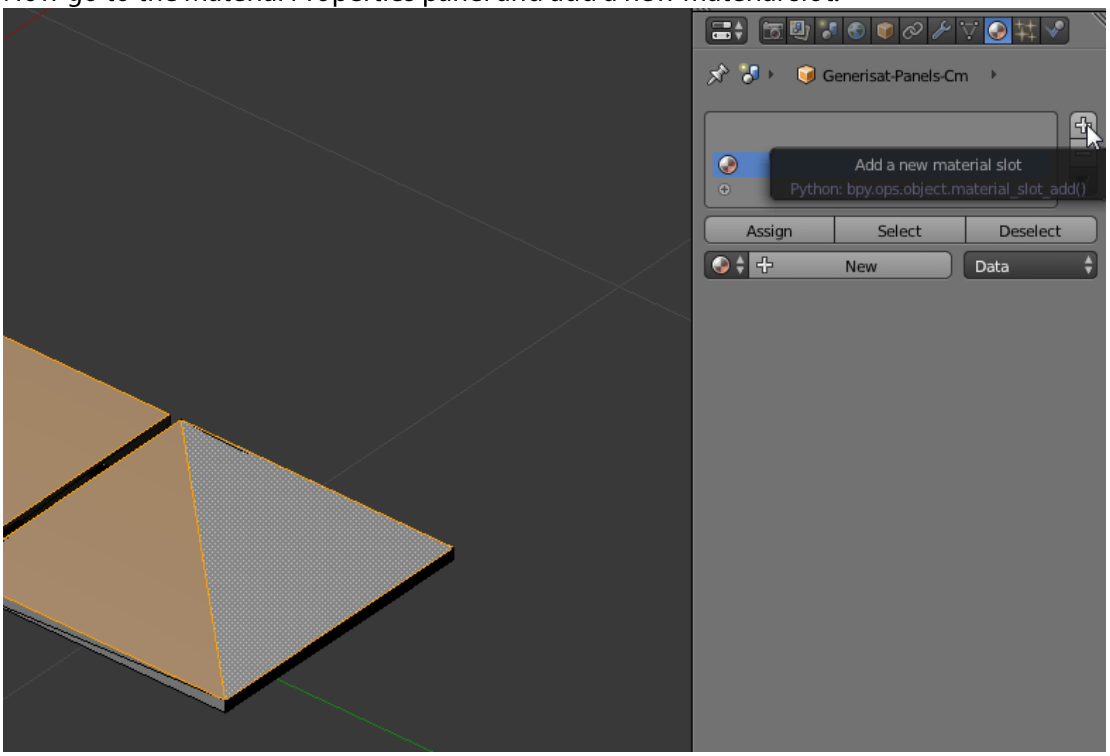
Note: *In order for the Solar Panel Tool to work in STK, no cell can be covered by a texture.*
This section covers the initial setup for Solar Panel Groups. See the section on "Ancillary File Creation" for the remainder.

In "Outliner", select the layer that the panels are on.

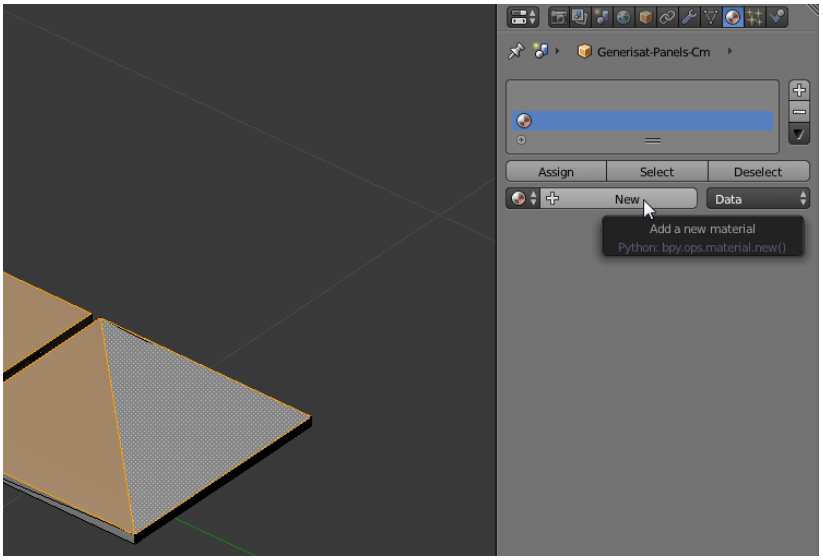
Enter into "Edit" mode (Tab key).
Change your selection type to "Polygon"
Use Shift-RMB and select the polygons that represent just the cells surface.



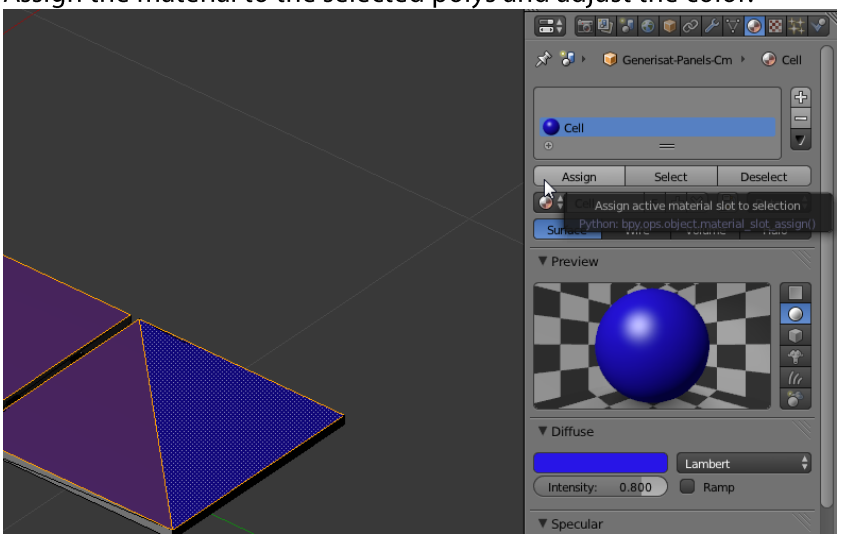
Now go to the Material Properties panel and add a new material slot.



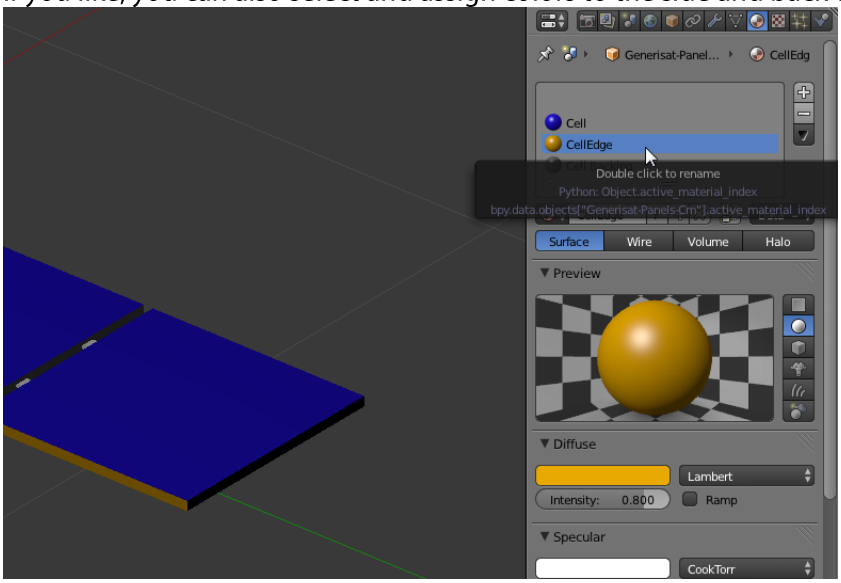
Then add a new material to that slot



Assign the material to the selected polys and adjust the color.

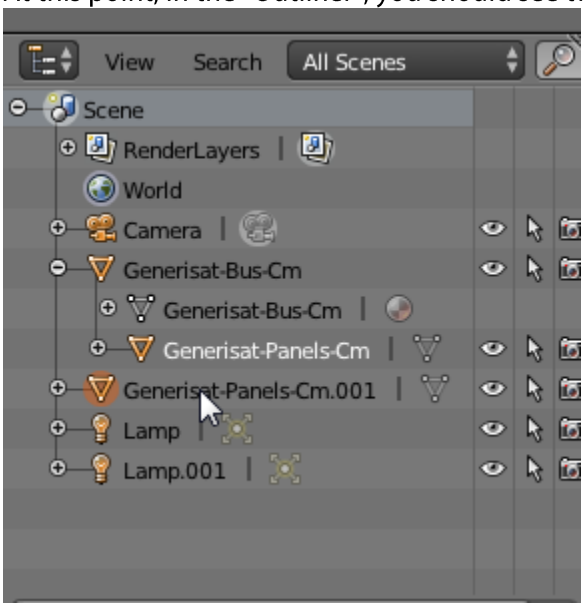


If you like, you can also select and assign colors to the side and back of the panels.



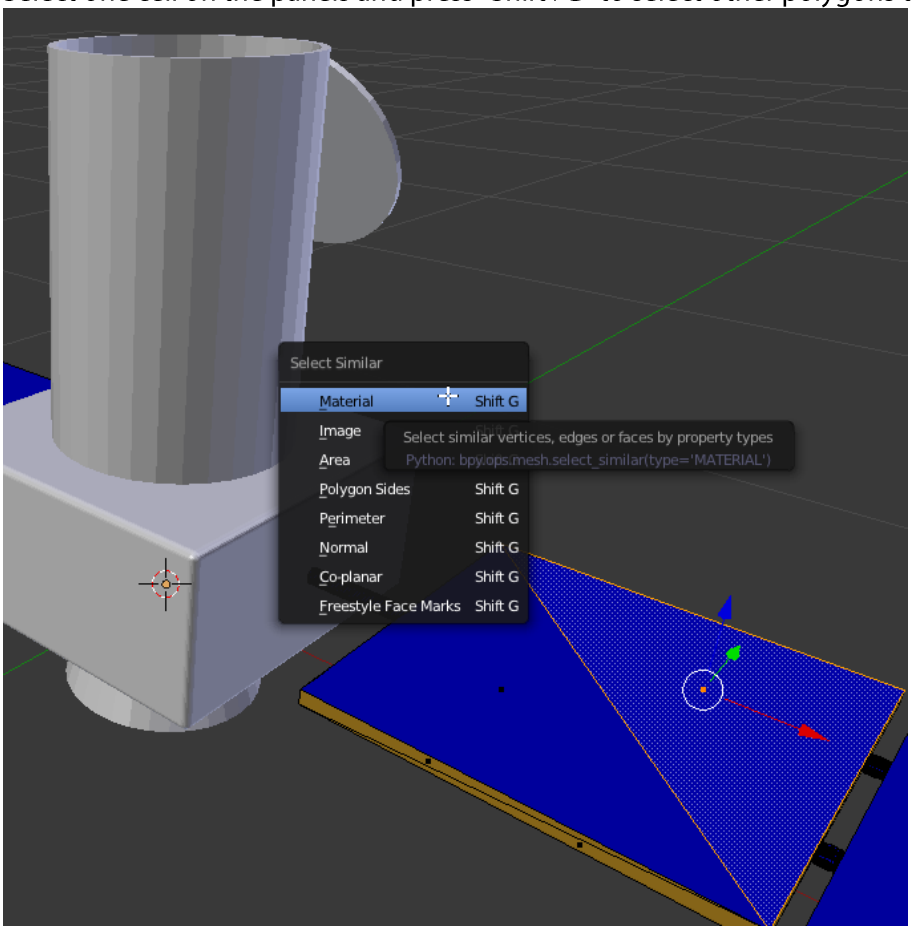
Now that we have assigned the colors that we want, we need to take an additional step in order to ensure that only the cells will be evaluated by STK's Solar Panel Tool and not the other geometry in this layer.
We are going to separate out just the cells, put them into their own layer and parent them back to the solar panels.
To do this first make sure that you still have the "Generisat-Panels-Cm" layer selected and that you are back in "Object Mode" (Tab key).

Make sure that your mouse in over the OpenGl window and copy (Ctrl-c) and paste (Ctrl-v)
At this point, in the "Outliner", you should see two copies of the solar panel.

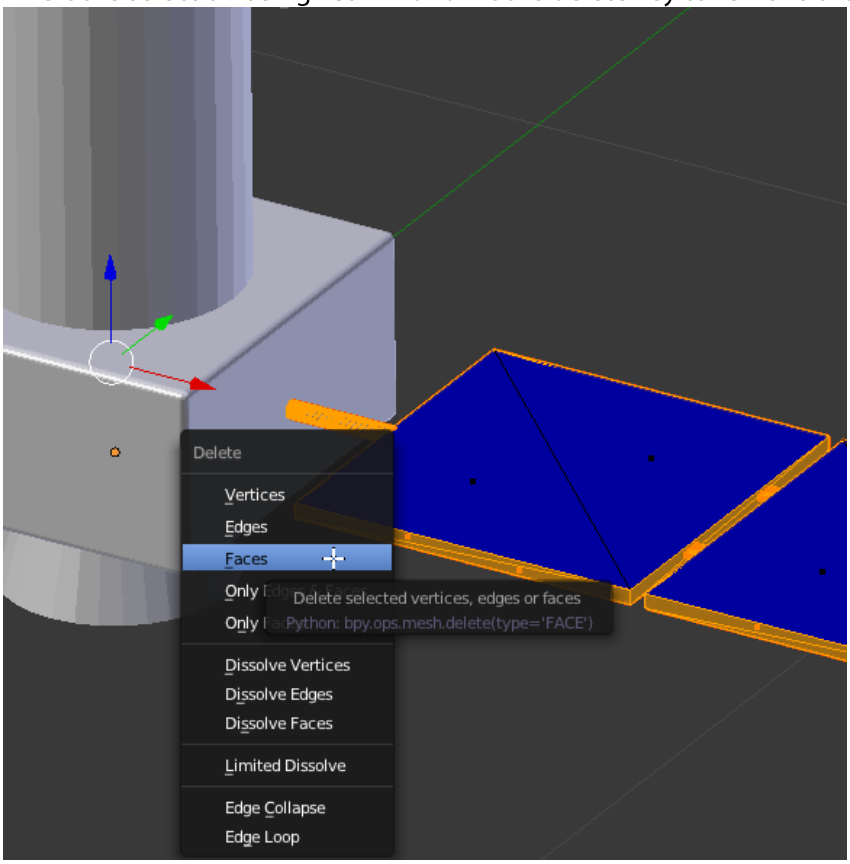


Rename the copied geometry "SolarCells" or whatever you like.

Select the “SolarCells” mesh, use the TAB key to go into “Edit Mode” and change your selection type to “Face”. Select one cell on the panels and press “Shift+G” to select other polygons that have the same material.

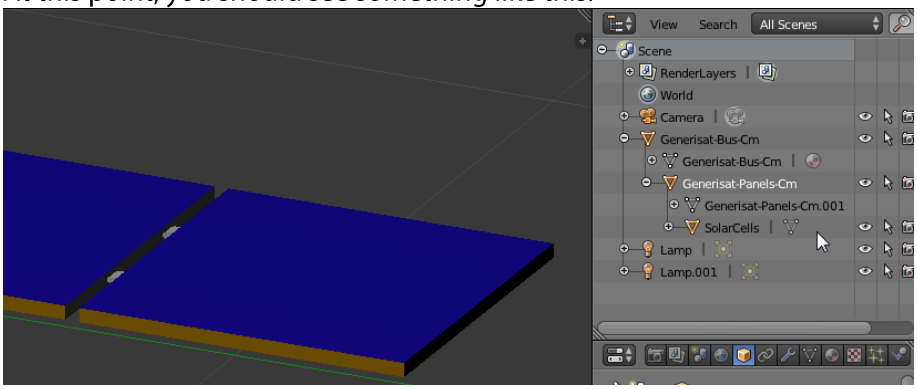


Invert the selection using “Ctrl+i” and hit the delete key to remove the unwanted geometry.

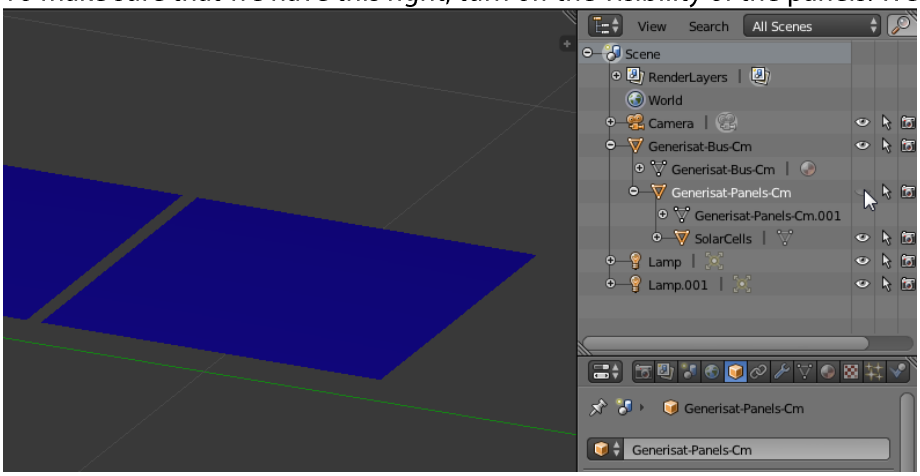


So now, only the cells reside on this layer that we just created, but the original layer still has the cells, so go back to the layer “Generisat-Panels-Cm”, go into “Edit Mode” select the cells surface polygons and remove them.
Now, parent the new layer “SolarCells” to the layer containing the panels “Generisat-Panels-Cm”

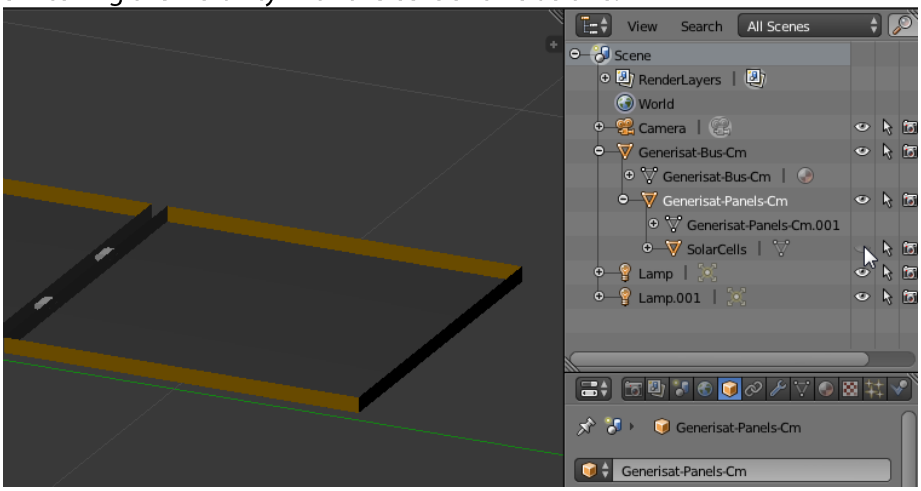
At this point, you should see something like this.



To make sure that we have this right, turn off the visibility of the panels. We should see this.



Switching the visibility with the cells shows us this.



Save the file as generisat_a.blend

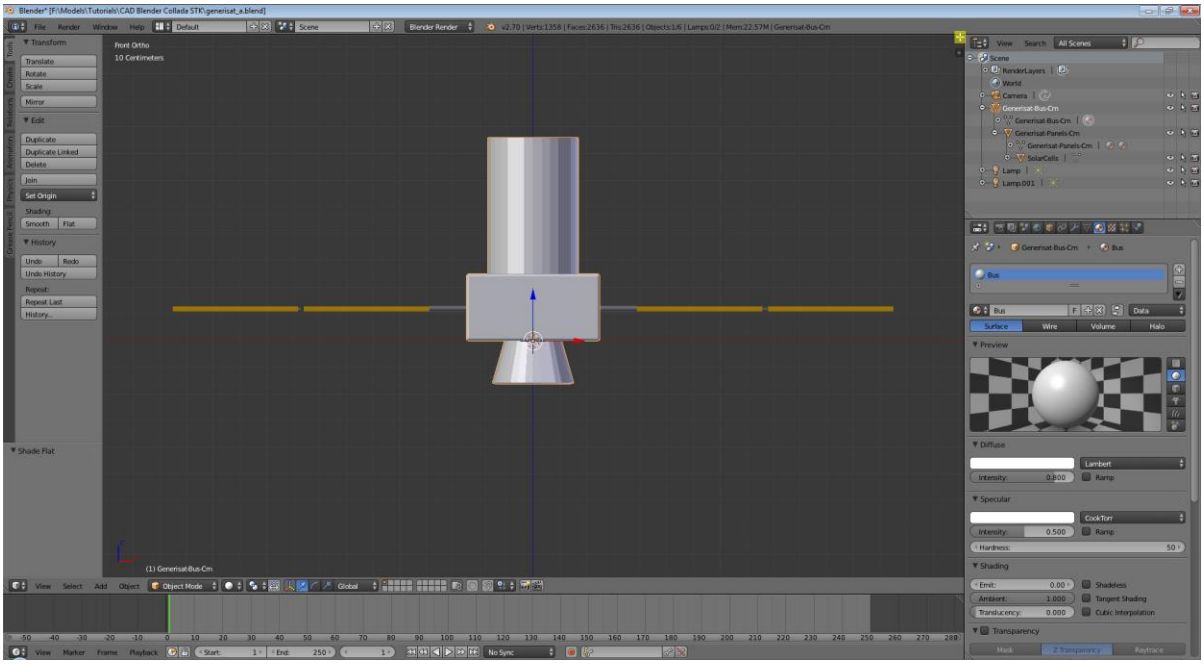
More on assigning materials can be found here
http://wiki.blender.org/index.php/Doc:2.6/Manual/Materials/Assigning_a_material

ASSIGNING A UV MAP

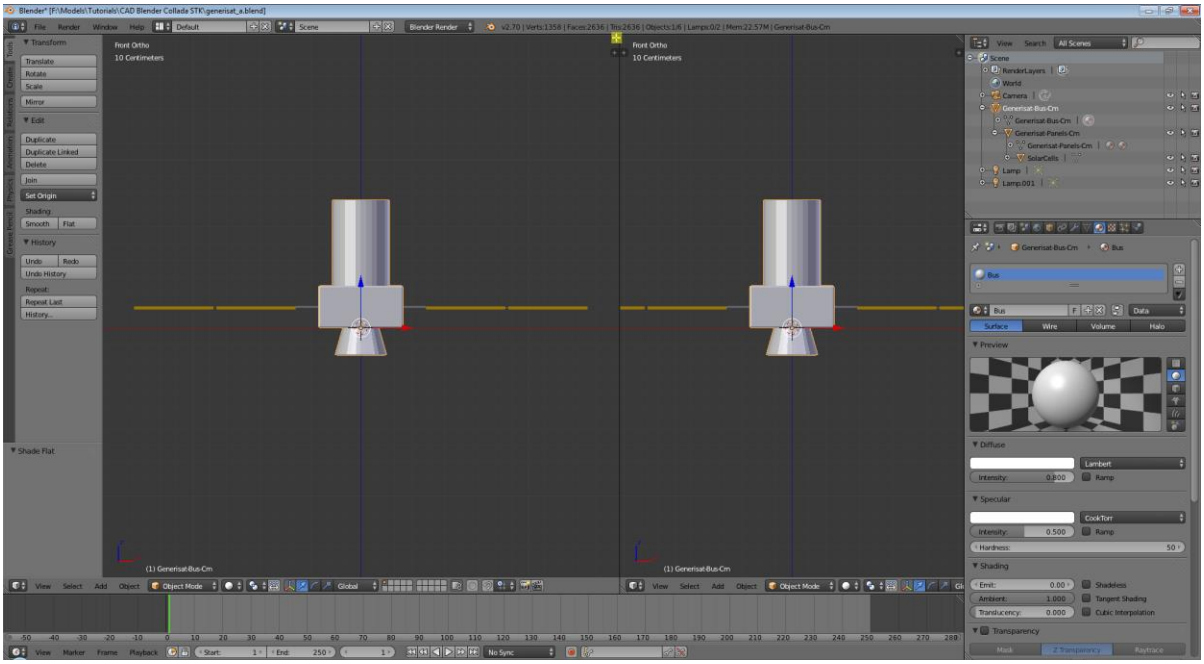
Note: *If you do not plan on adding textures to your model, you can skip this step.*
Giving the model UV's are important if you want to texture your model.
But, they are not necessary and will load into STK without them.

Note: *These are steps on how to add a basic UV maps.*
More on UV mapping can be found http://wiki.blender.org/index.php/Doc:2.6/Manual/Textures/Mapping/UV/

Split the window frame.
Before

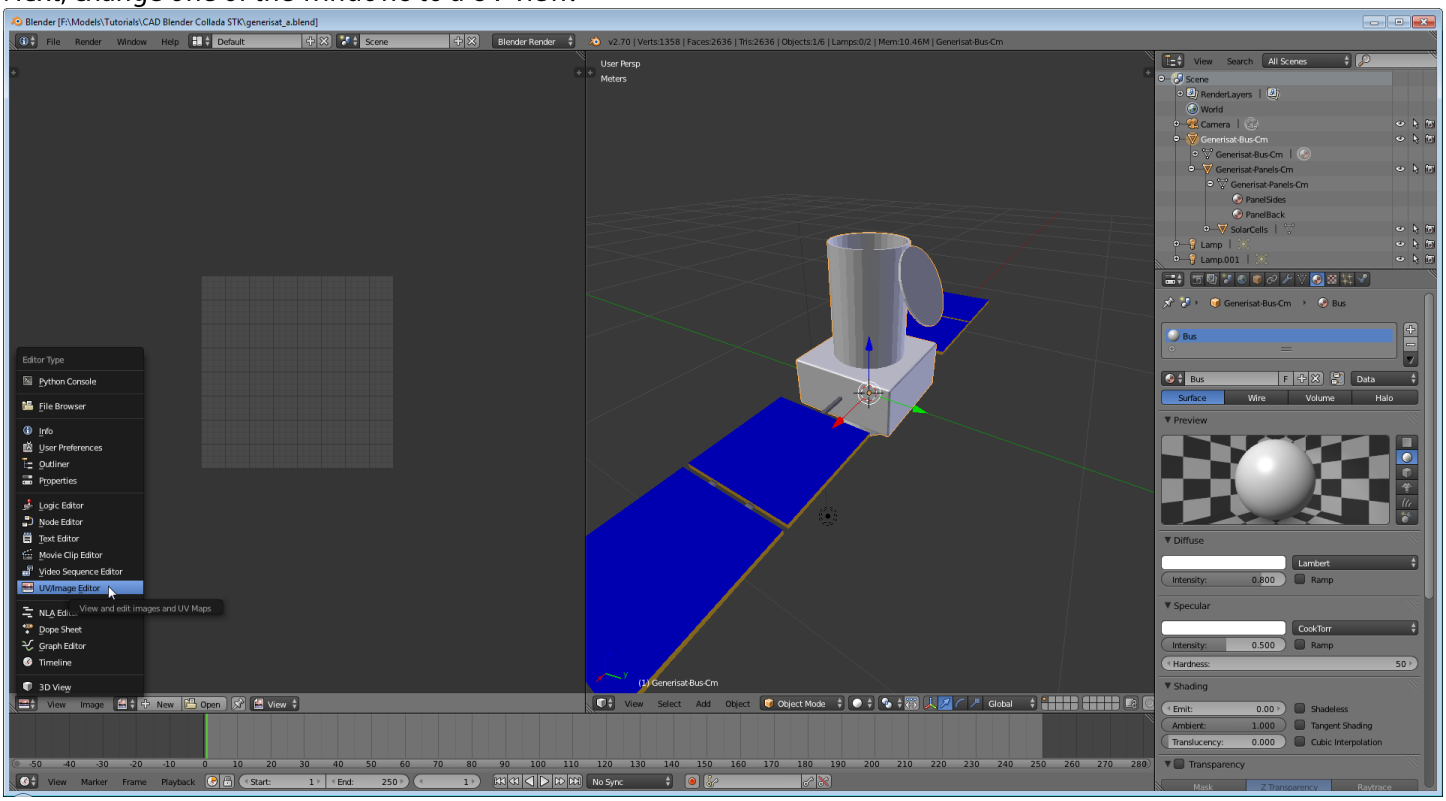


After

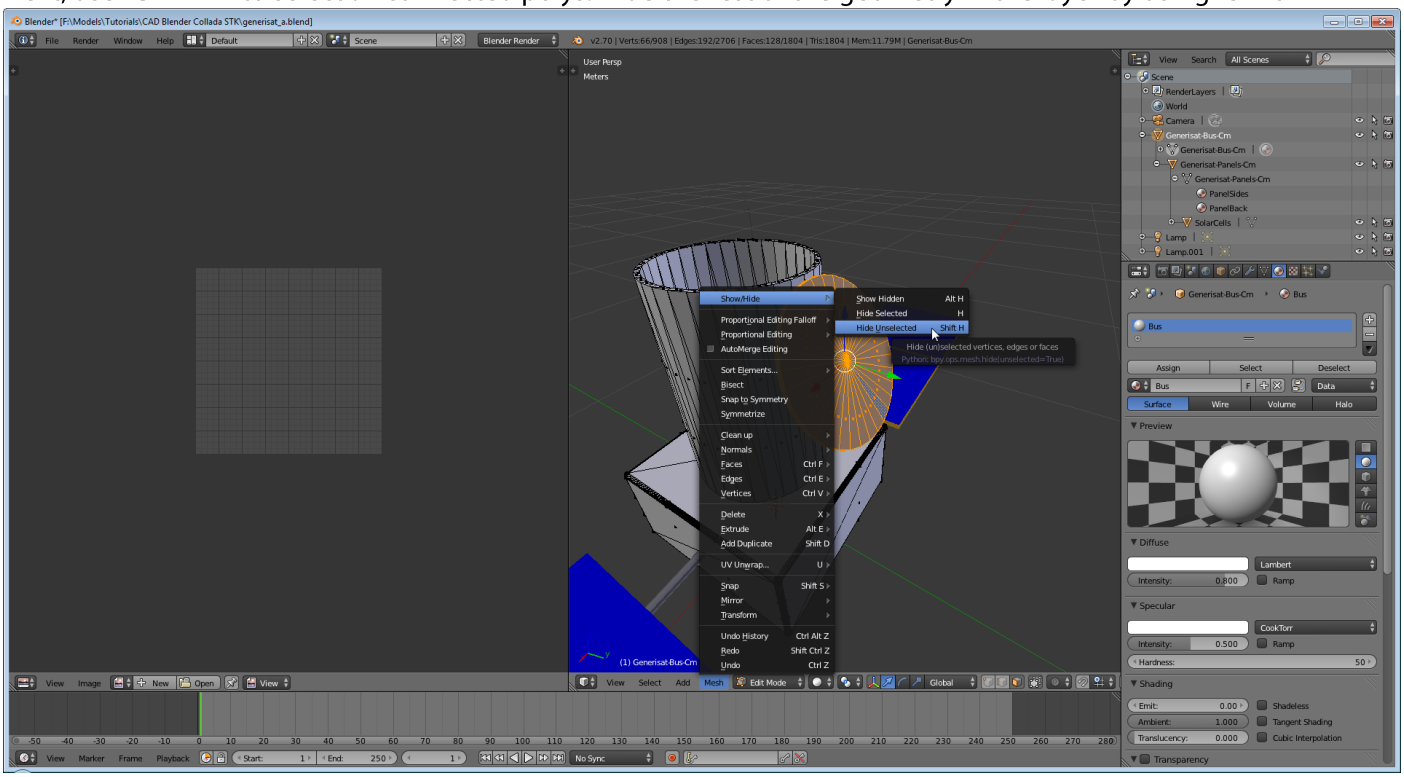


See this link for more info on splitting and merging windows.
http://wiki.blender.org/index.php/Doc:2.6/Manual/Interface/Window_system/Arranging_frames/

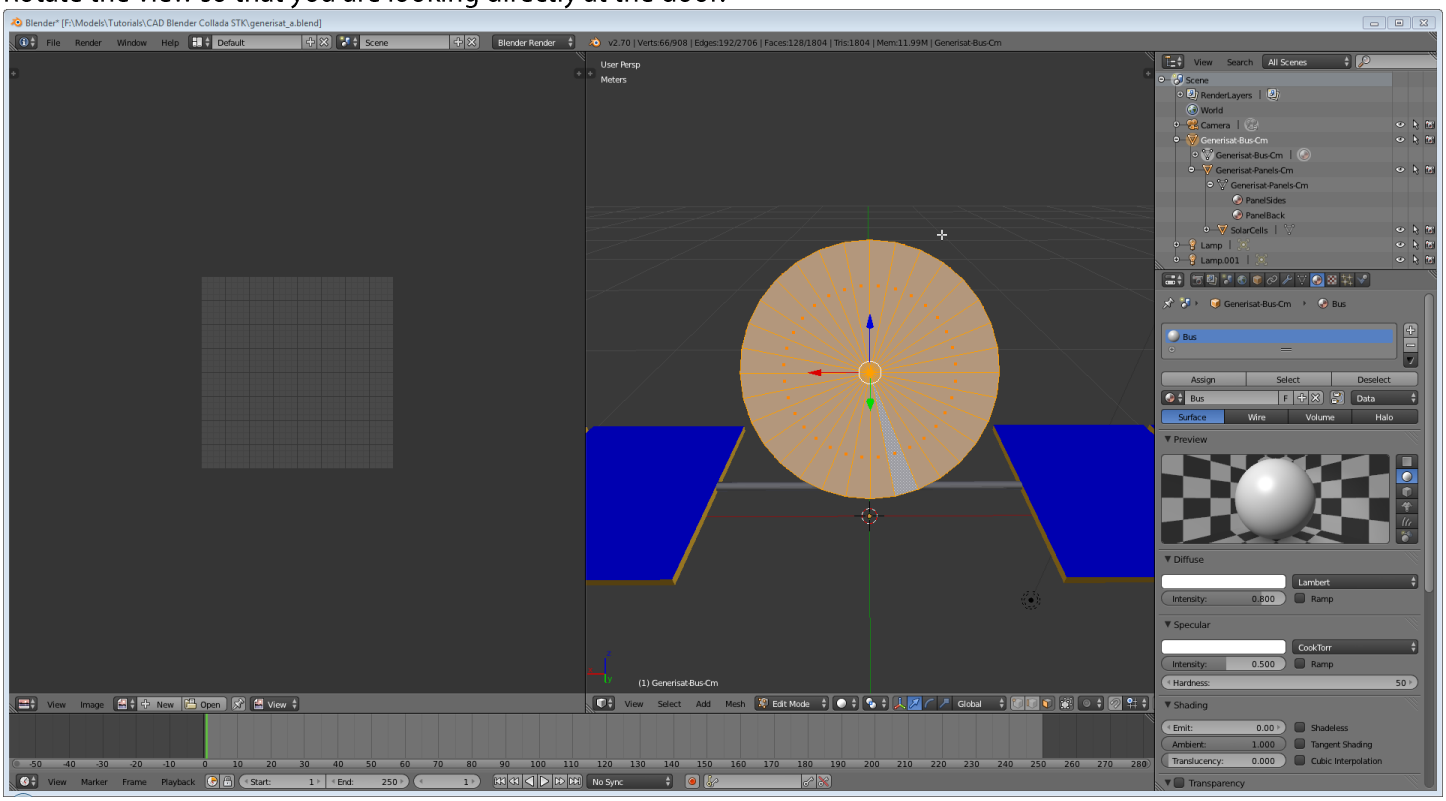
Press "t" to close the Tool Bar of each window frame.
Hint: *Hover mouse over OpenGL window.*
Next, change one of the windows to a UV view.



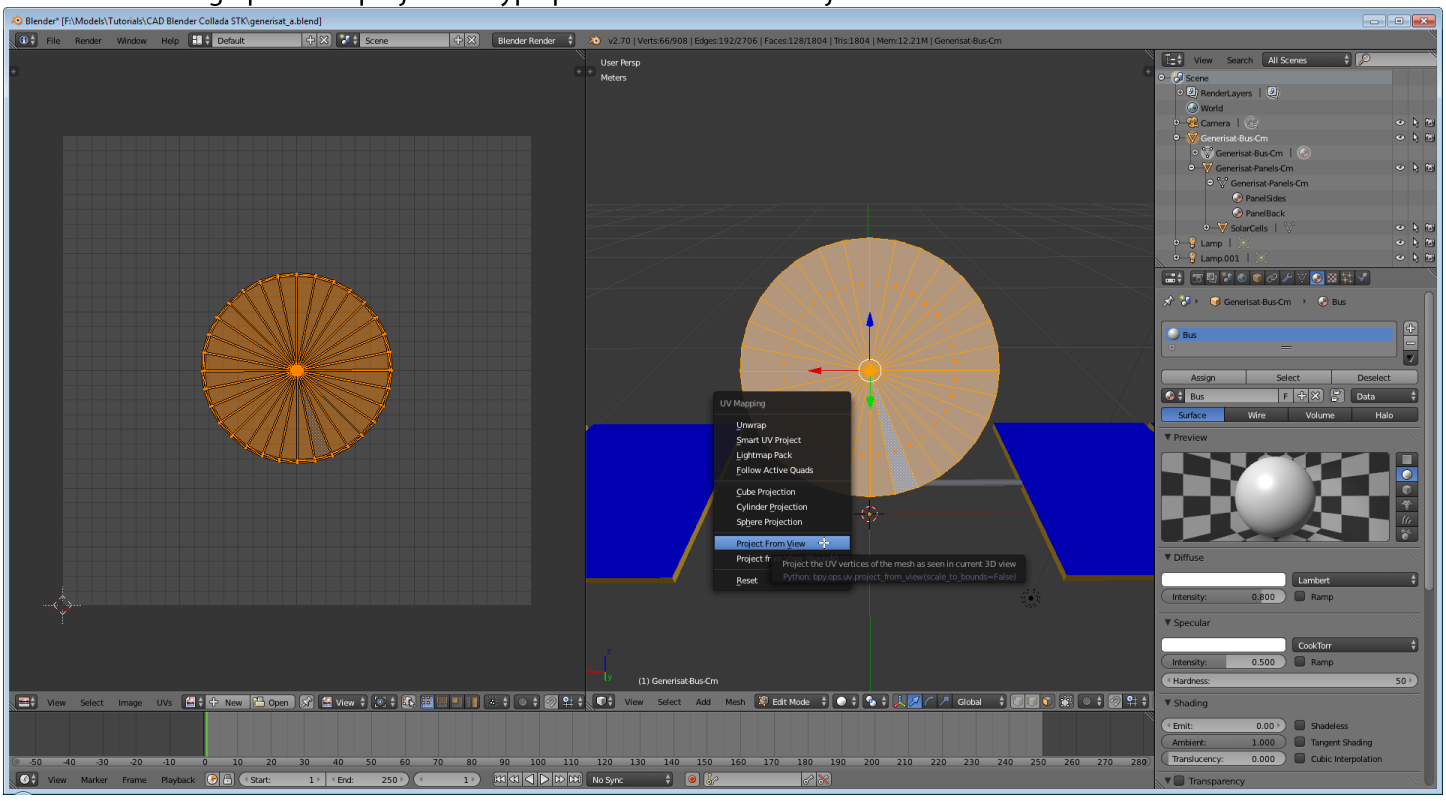
We are going to select and add a UV map to each part.
Make sure that you have the “Generisat-Bus-Cm” layer selected and hit the "Tab" key to go into "Edit Mode"
All geometry will be selected by default, so with the cursor in the window with the geometry, press "a" to select none.
We are going to select just one part at a time, hide the rest and create UVs for each part.
We’ll start with the camera door. Select one polygon of the door with the RMB.
Next, use “CTRL+L” to select all connected polys. Hide the rest of the geometry in the layer by using “Shift+H”



Rotate the view so that you are looking directly at the door.

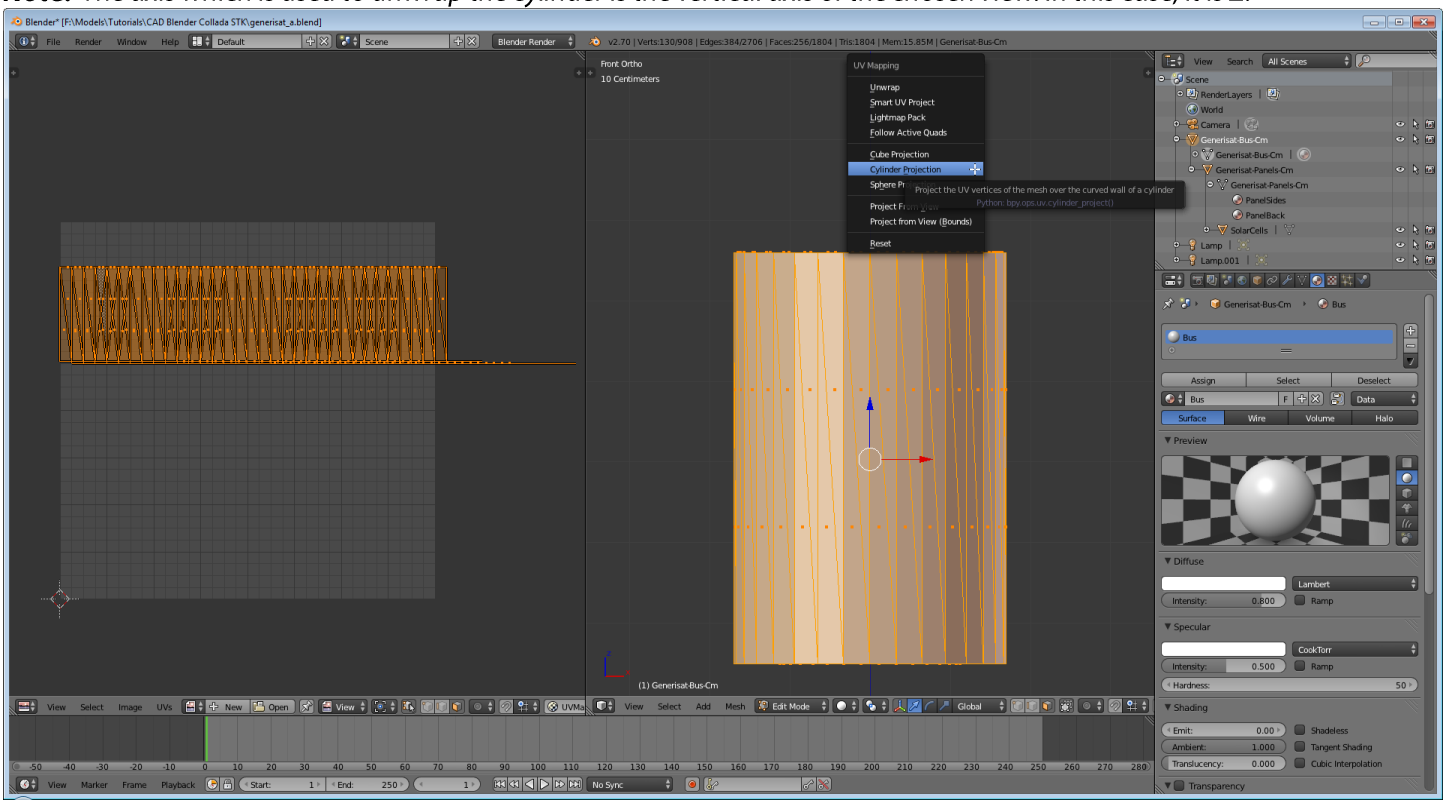


Select “u” to bring up the UV projection type panel and choose “Project From View”

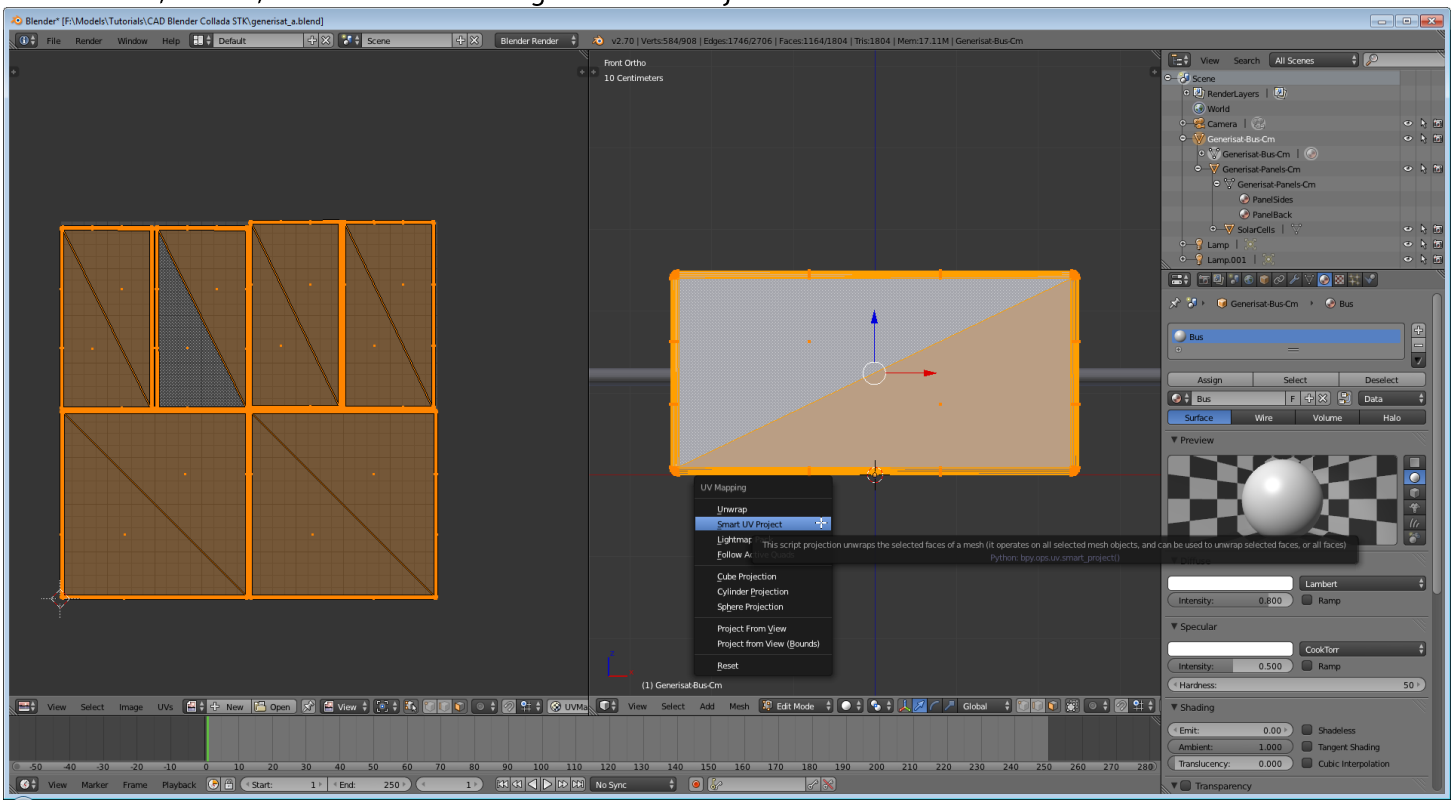


.

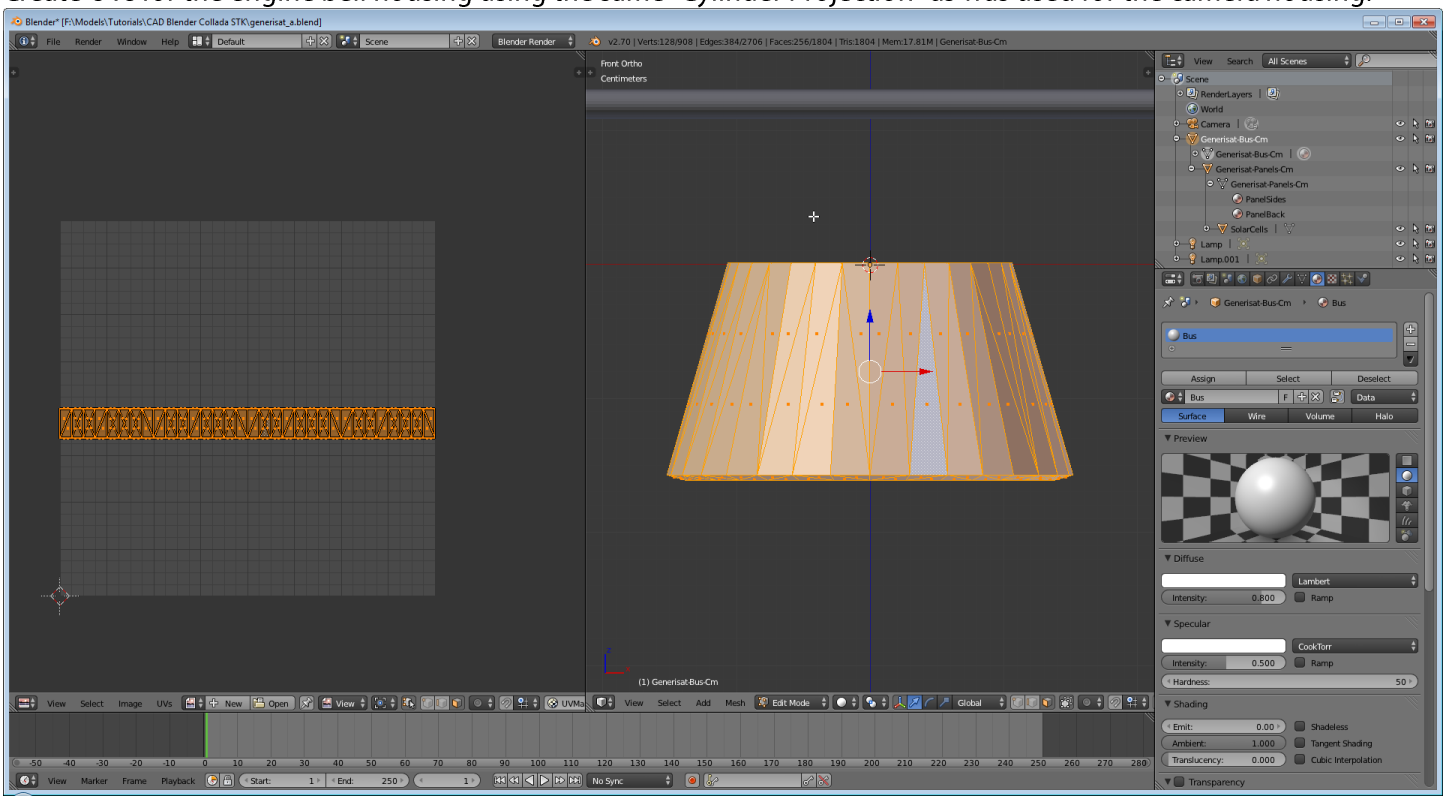
Now move back over to the right window and unhide the rest of the mesh using “Alt+H”.
Select the camera housing, hide the rest of the geometry and use “1”on the numbers key pad to go to a front view.
Select “u” to bring up the UV projection type panel and choose “Cylinder Projection”
Note: *The axis which is used to unwrap the cylinder is the vertical axis of the chosen view. In this case, it is Z.*



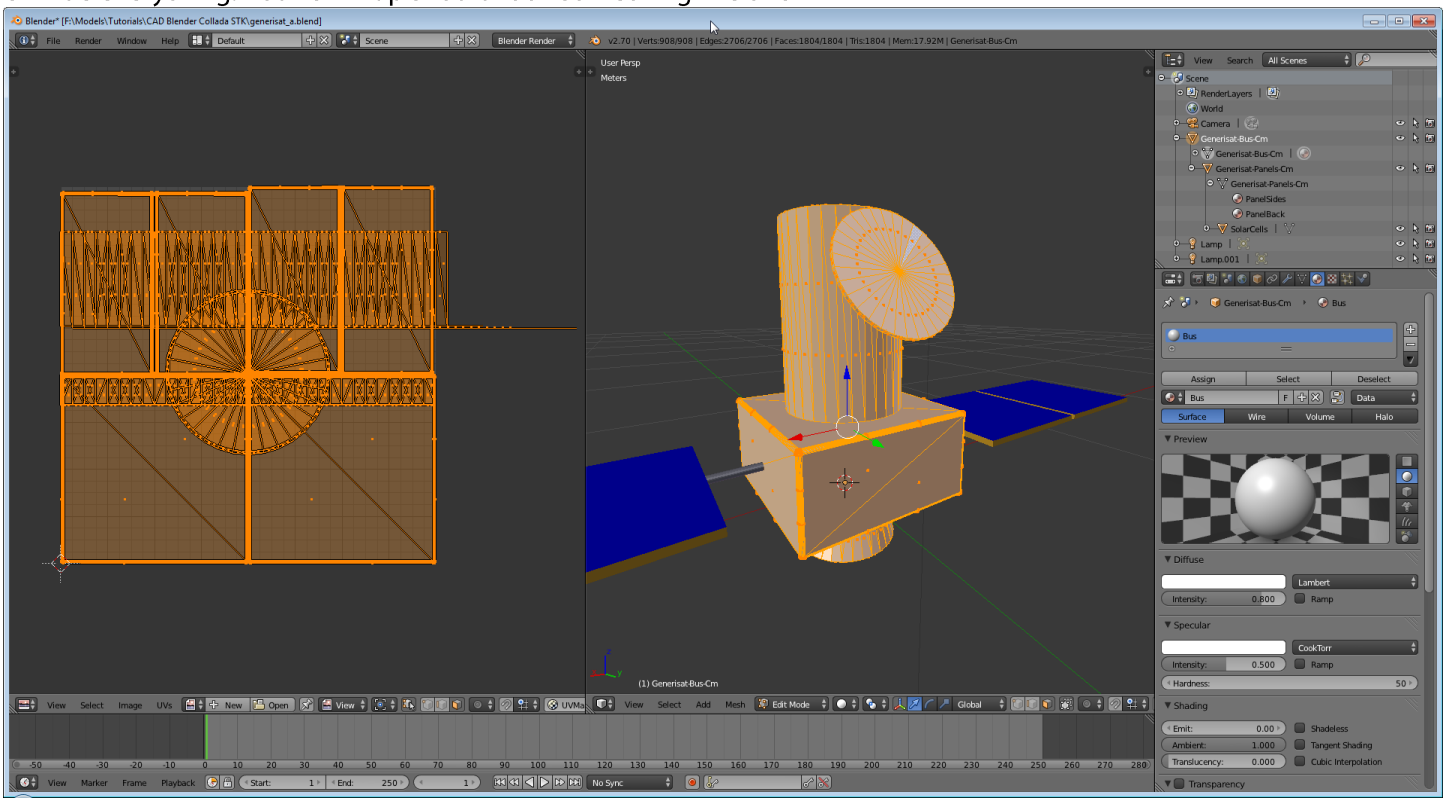
Unhide the rest of the geometry again and select the box that represents the heart of the bus.
Hide the rest, use “u”, and this time selecting “Smart UV Project” to create the UVs.



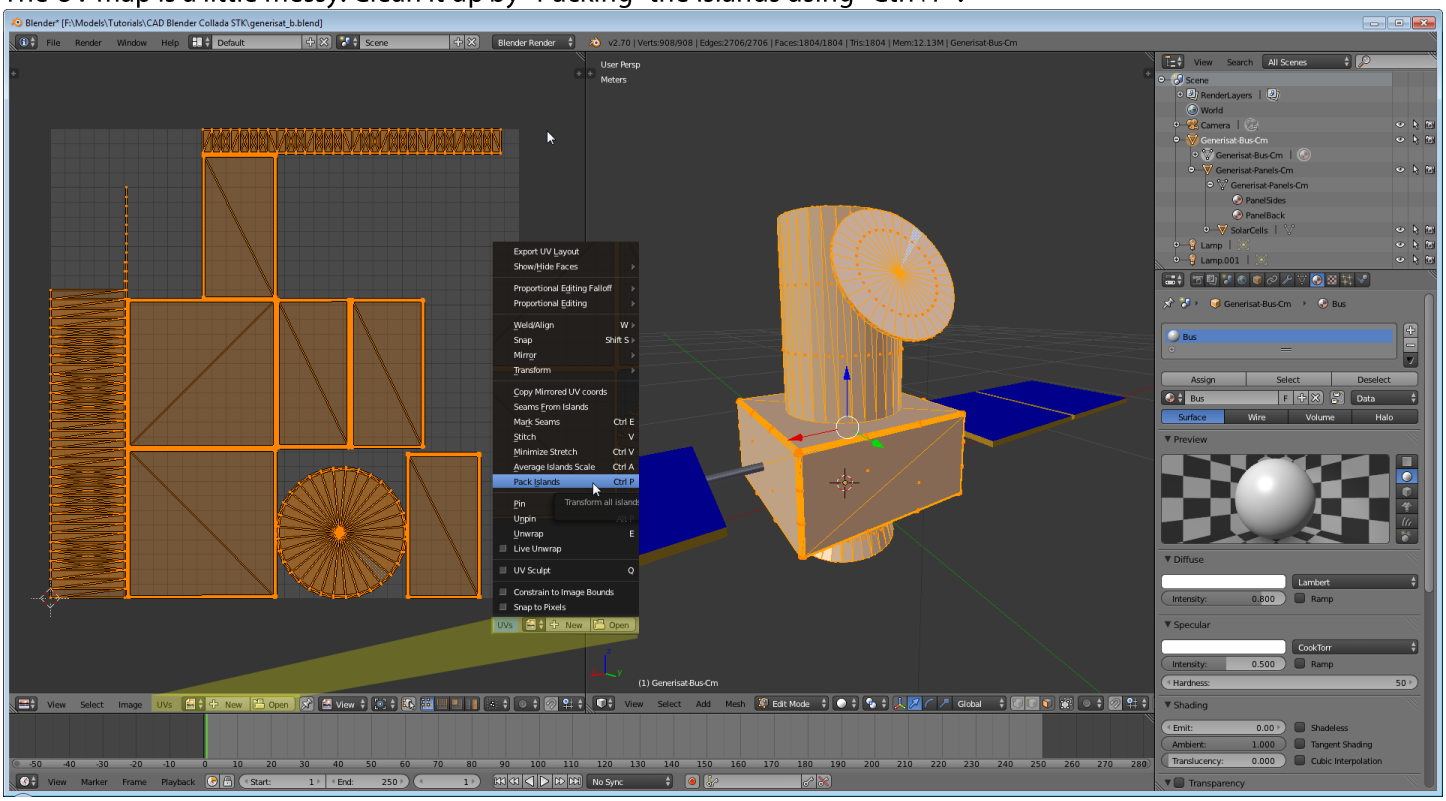
Create UVs for the engine bell housing using the same “Cylinder Projection” as was used for the camera housing.



Unhide everything. Your UV map should look something like this

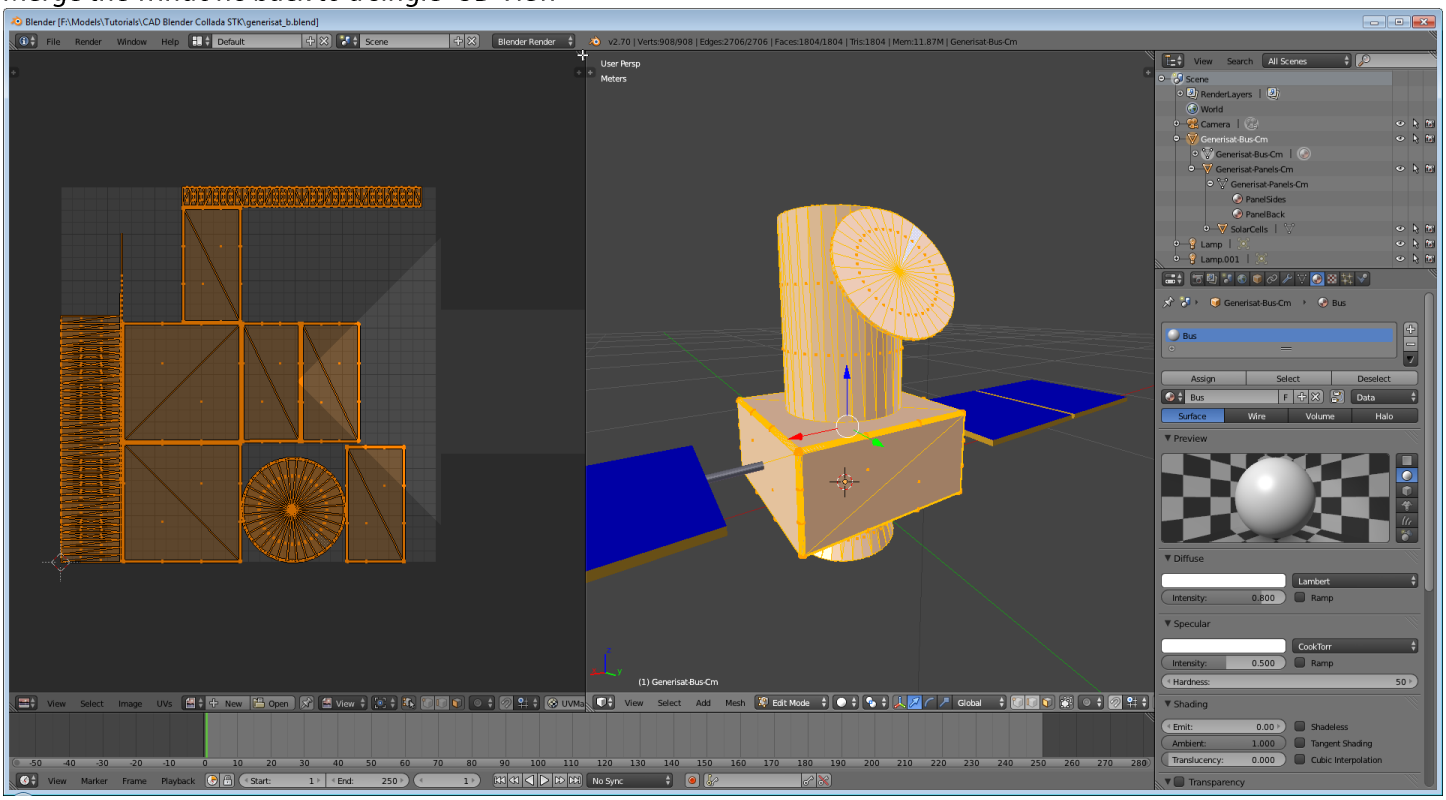


The UV map is a little messy. Clean it up by “Packing” the islands using “Ctrl+P”.



We are going to place a generic gold foil texture on the bus using this UV map.
If you have more complex UV mapping requirements see
http://wiki.blender.org/index.php/Doc:2.6/Manual/Textures/Mapping/UV

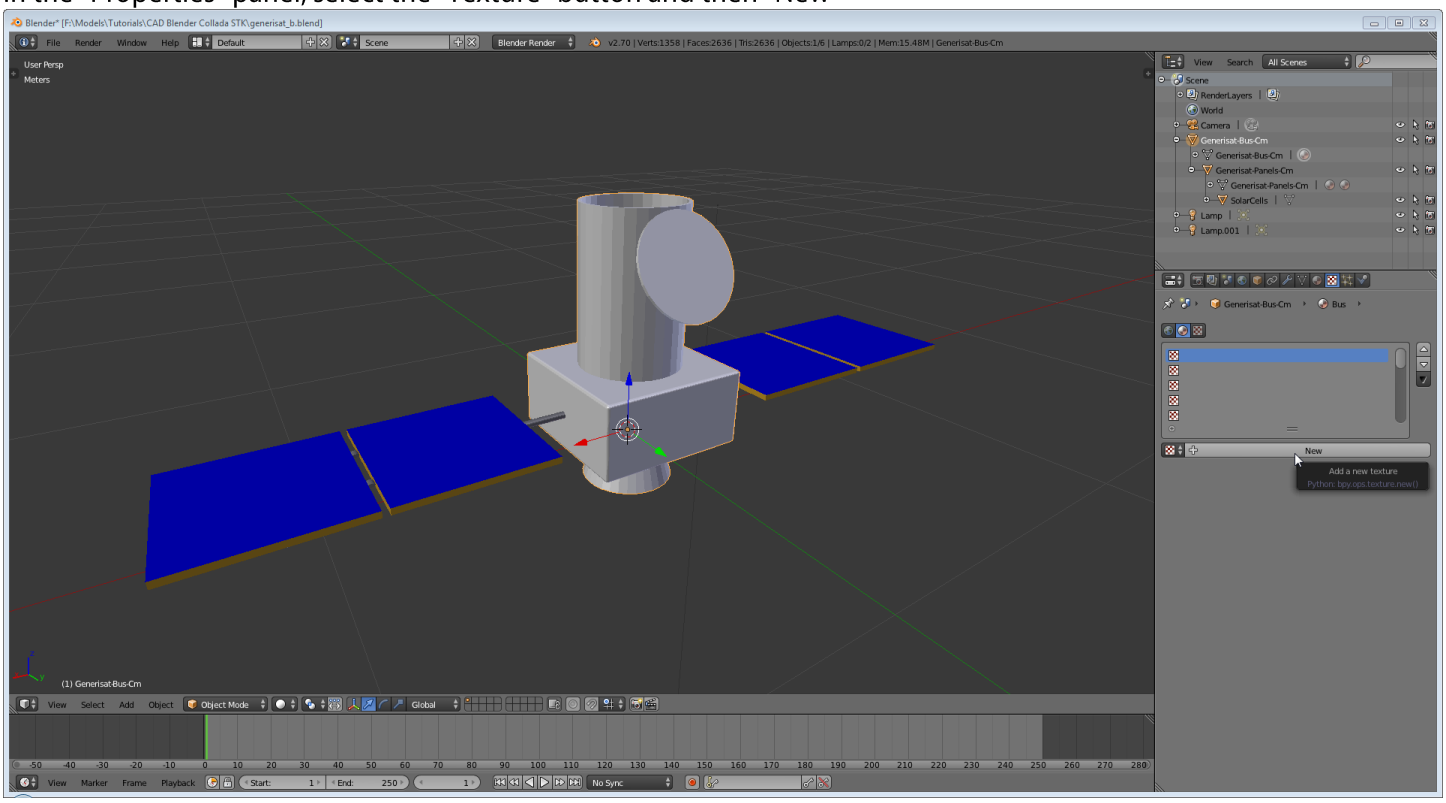
Merge the windows back to a single “3D View”



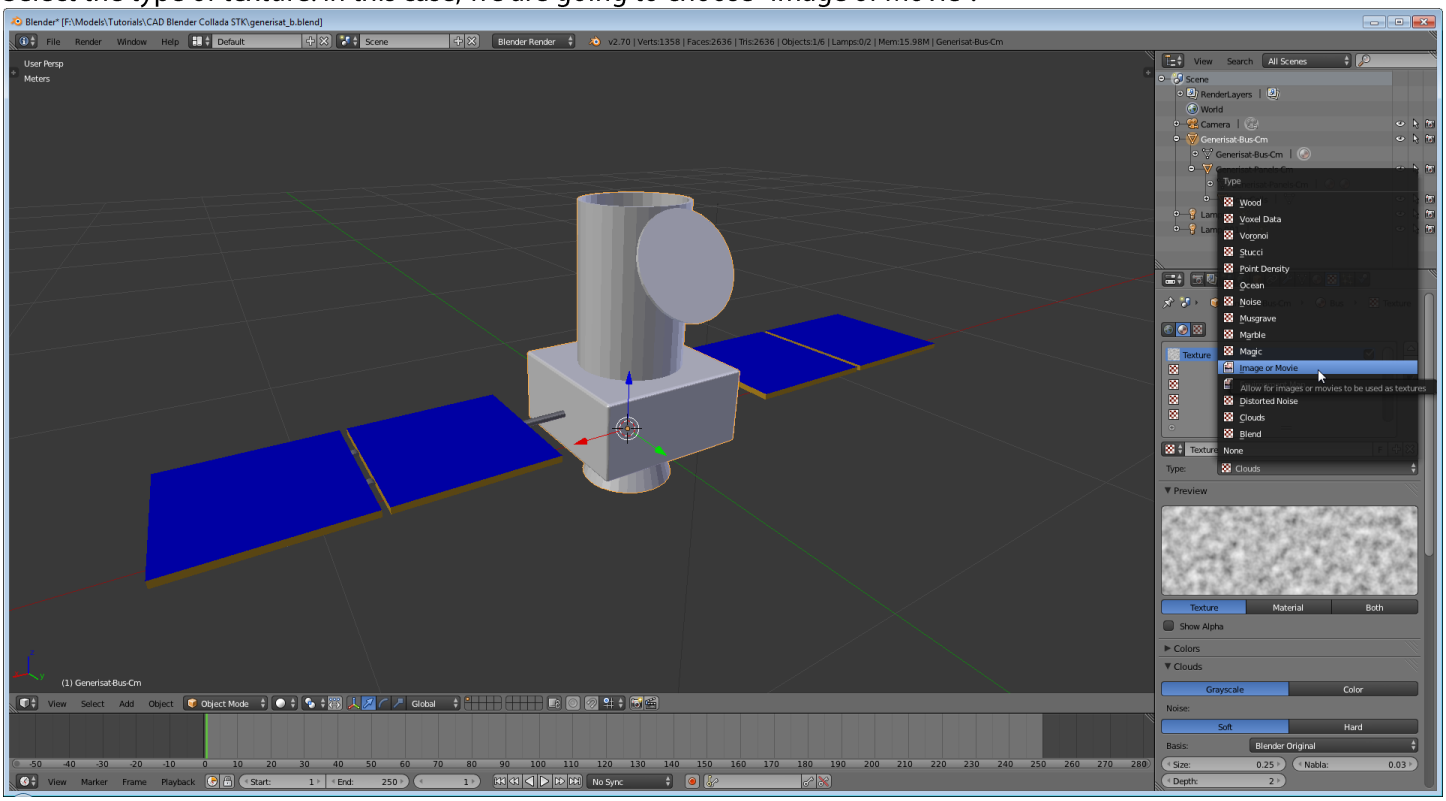
Save another .blend file as generisat_b.blend

ASSIGNING TEXTURES

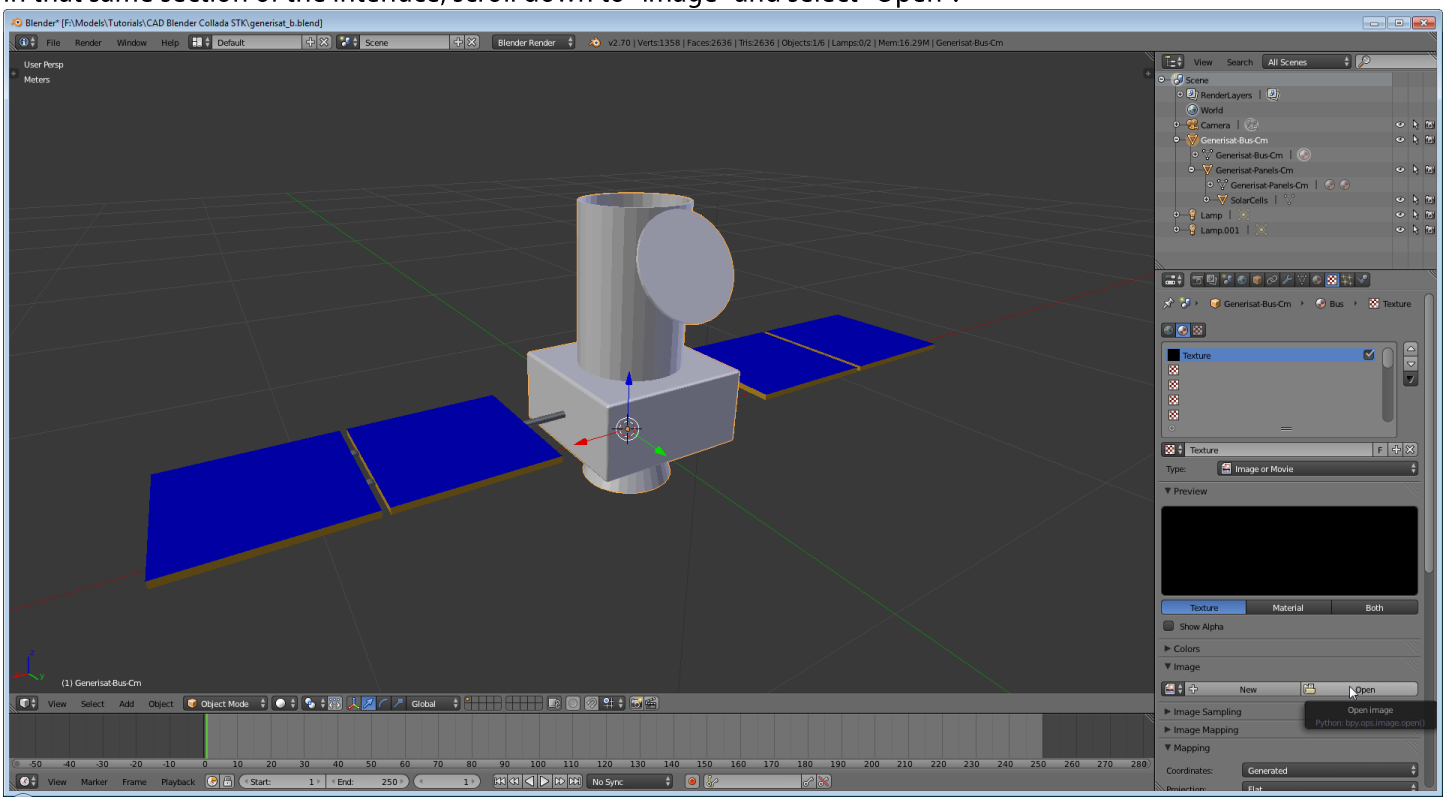
In the “Properties” panel, select the” Texture” button and then “New”



Select the type of texture. In this case, we are going to choose “Image or Movie”.

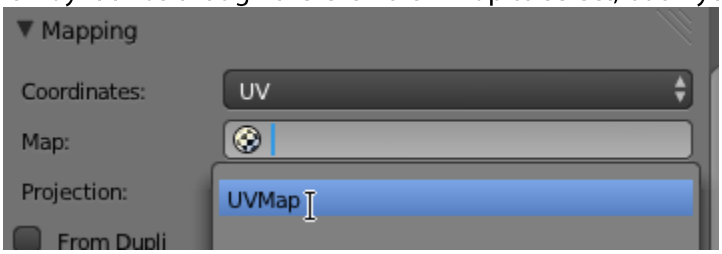


In that same section of the interface, scroll down to "Image" and select "Open".



Choose an image from your hard drive. You are welcome to use any image that came along with the STK install.
The location in a typical PC install is C:\Program Files (x86)\AGI\STK 10\STKData\VO\Textures\
I chose goldfoil4.tga which is included with this tutorial.

We need to assign this new image to our UV map.
Scroll a little further down to a section labeled "Mapping" and change the “Coordinates” to "UV" and select “UVMap”
It may look as though there is no UV map to select, but if you left click on the small checkered sphere you will see it.

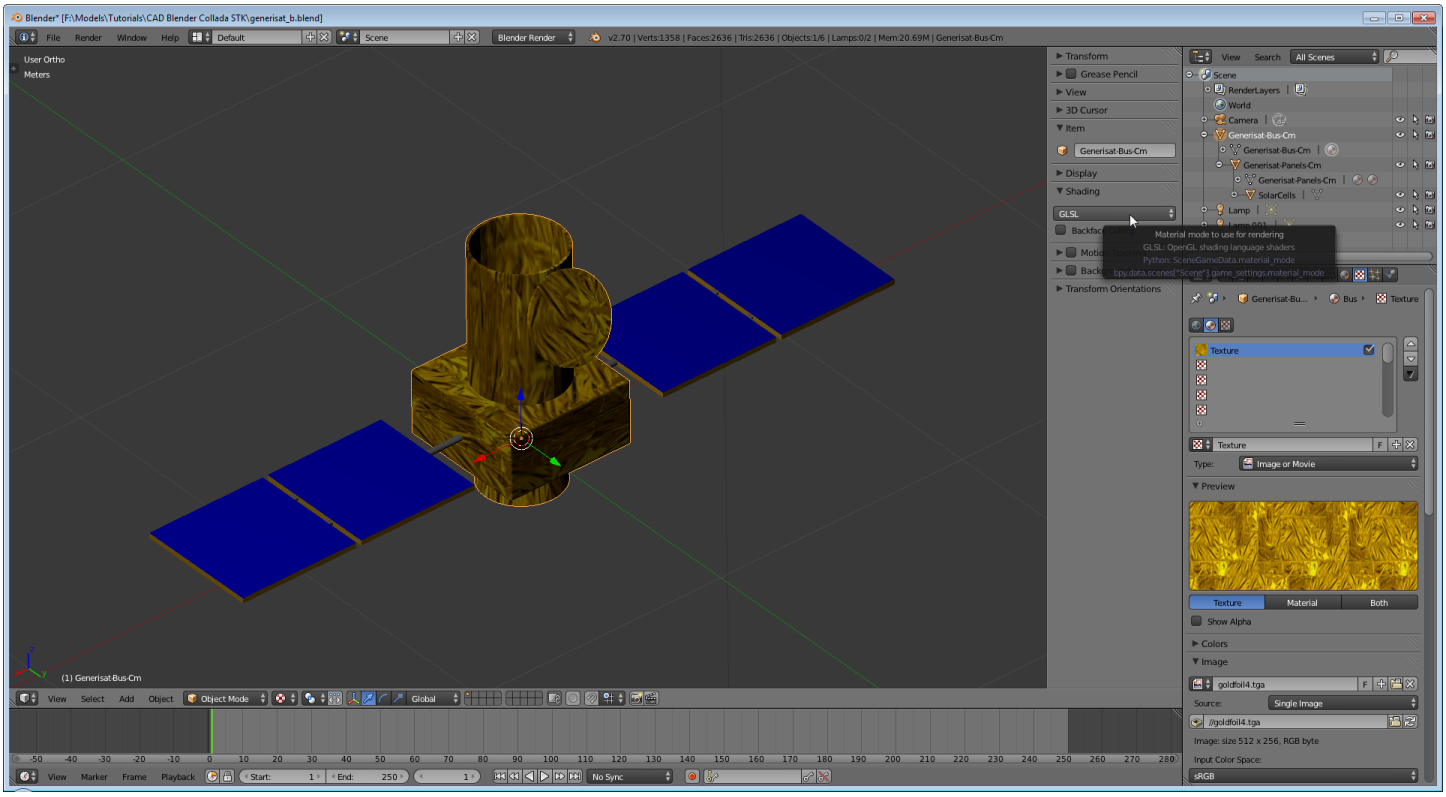


If you don’t see the texture on your model, there are two sections that you’ll need to change.
The first is the “Viewport Shading” and is along the bar directly under the viewport.

Make sure that it is set to “Texture”

The other is in the “Numeric Panel”
Again, this is the panel that is hidden on the right side of the main OpenGl window and can be accessed using the “n” key, while the mouse is over the OpenGl window.

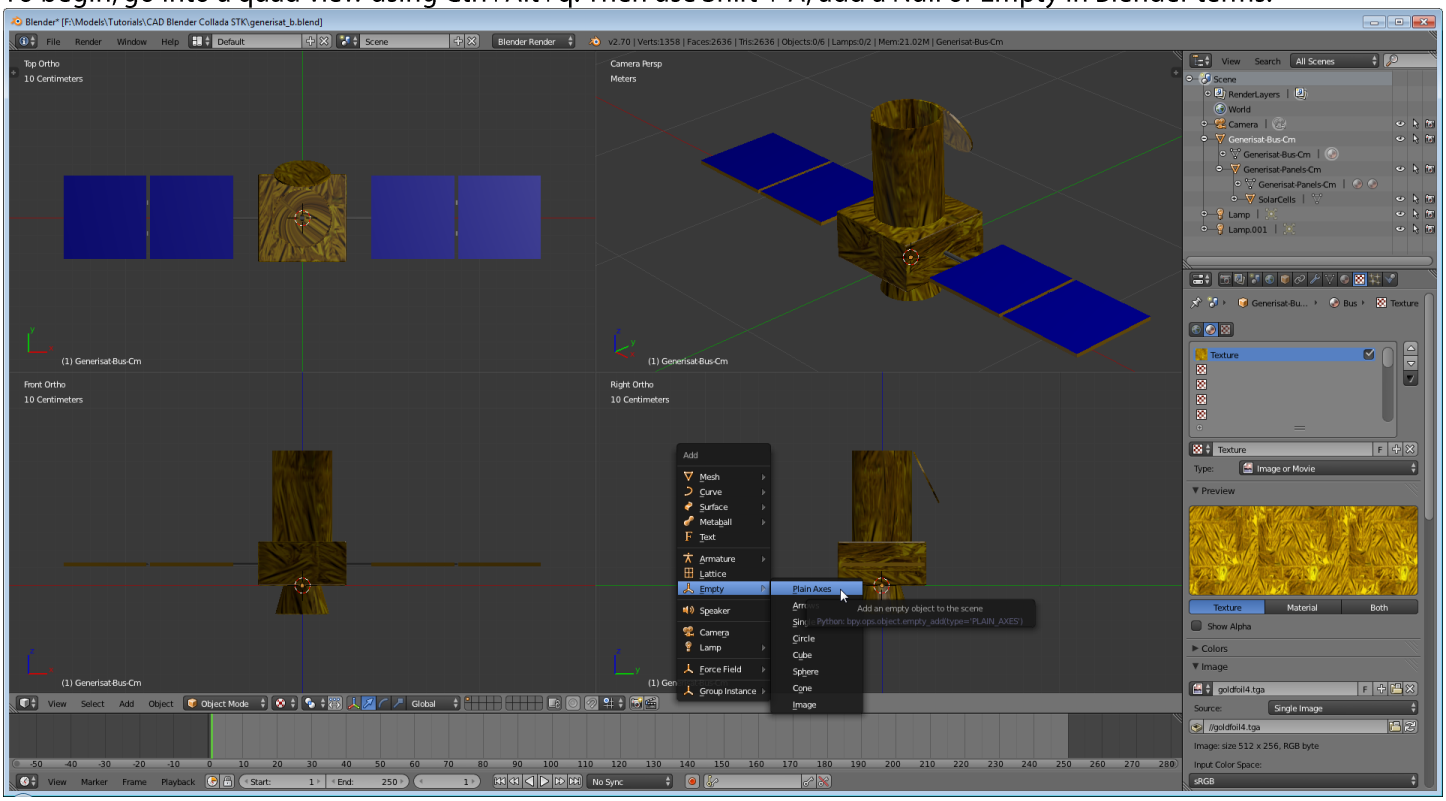
In the section titled “Display” and under “Shading”, select the type “GLSL”



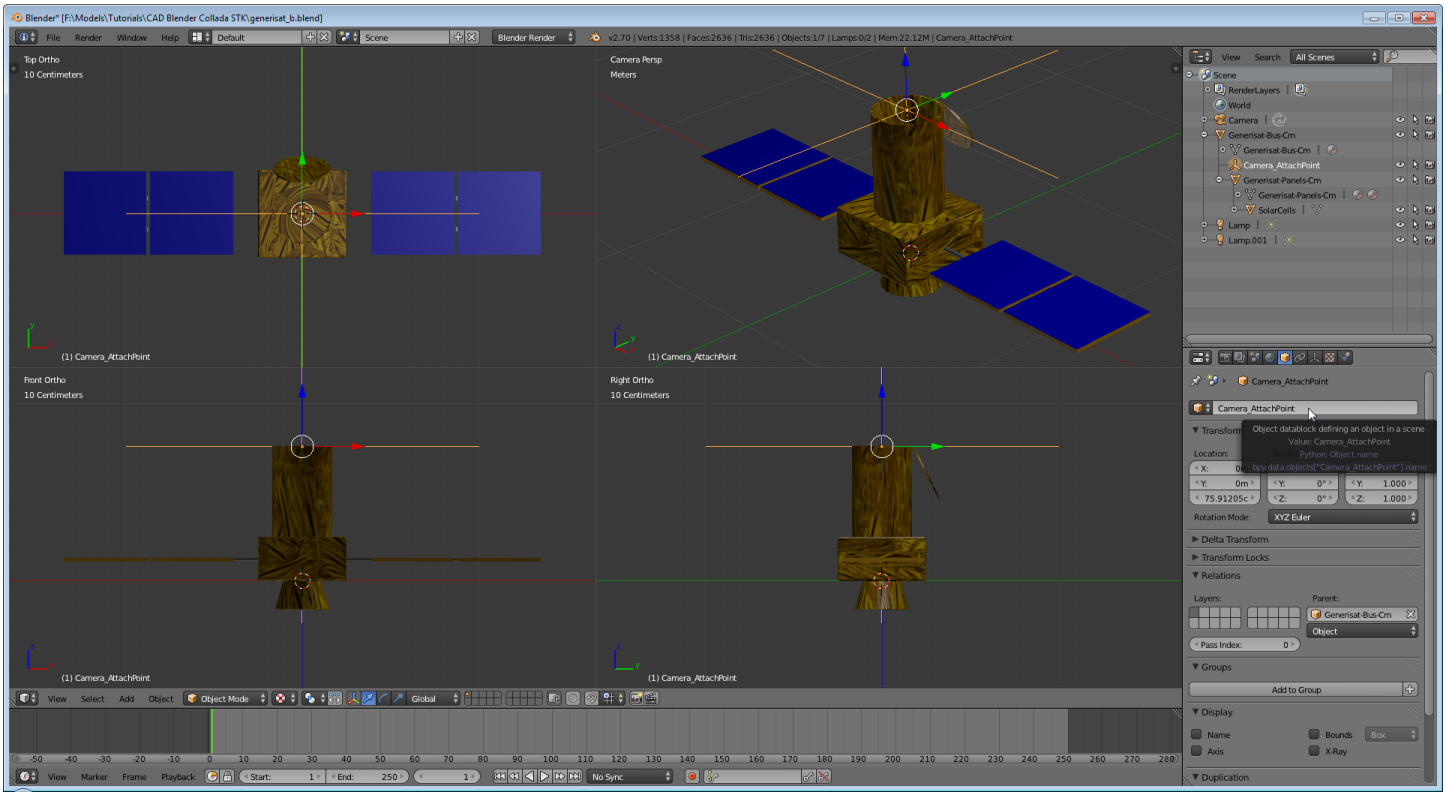
Please see Blender's documentation if you have more complex texturing requirements.
http://wiki.blender.org/index.php/Doc:2.6/Manual/Textures/Assigning_a_Texture/

SENSOR AND EFFECT ATTACH POINTS

Attach points are used as origins for either a STK sensor or smoke effect. Here we are going to add a sensor.
To begin, go into a quad view using Ctrl+Alt+q. Then use Shift + A, add a Null or Empty in Blender terms.

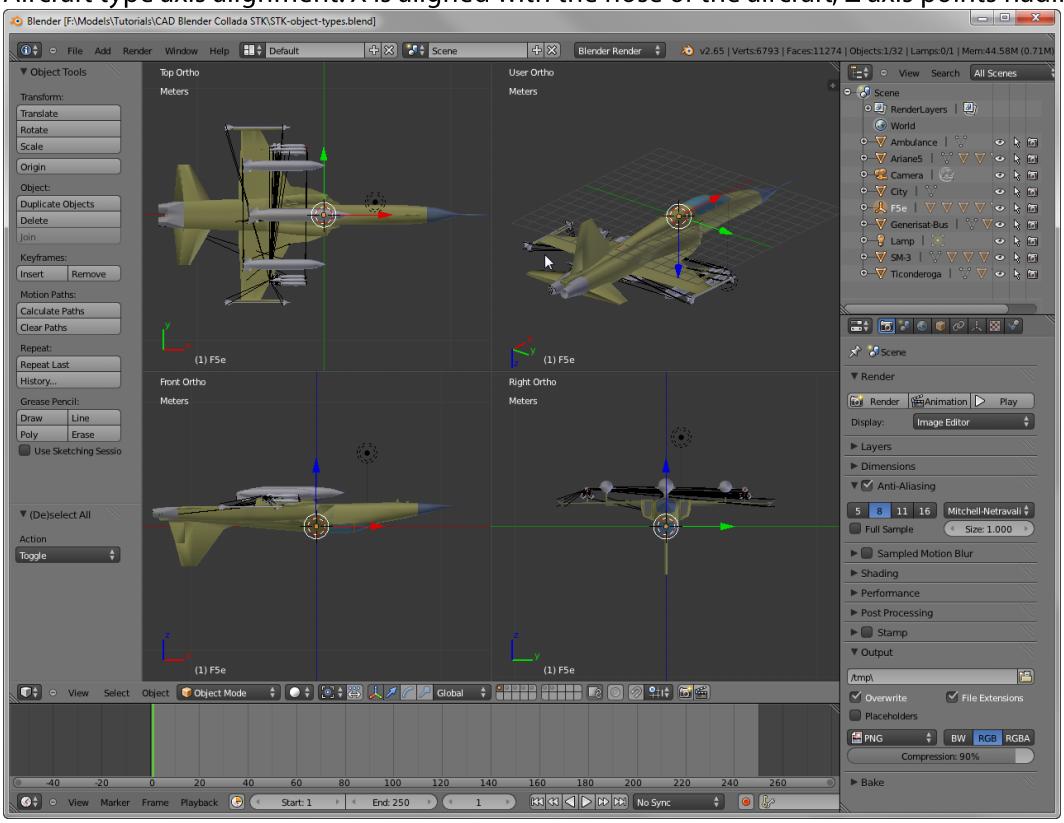


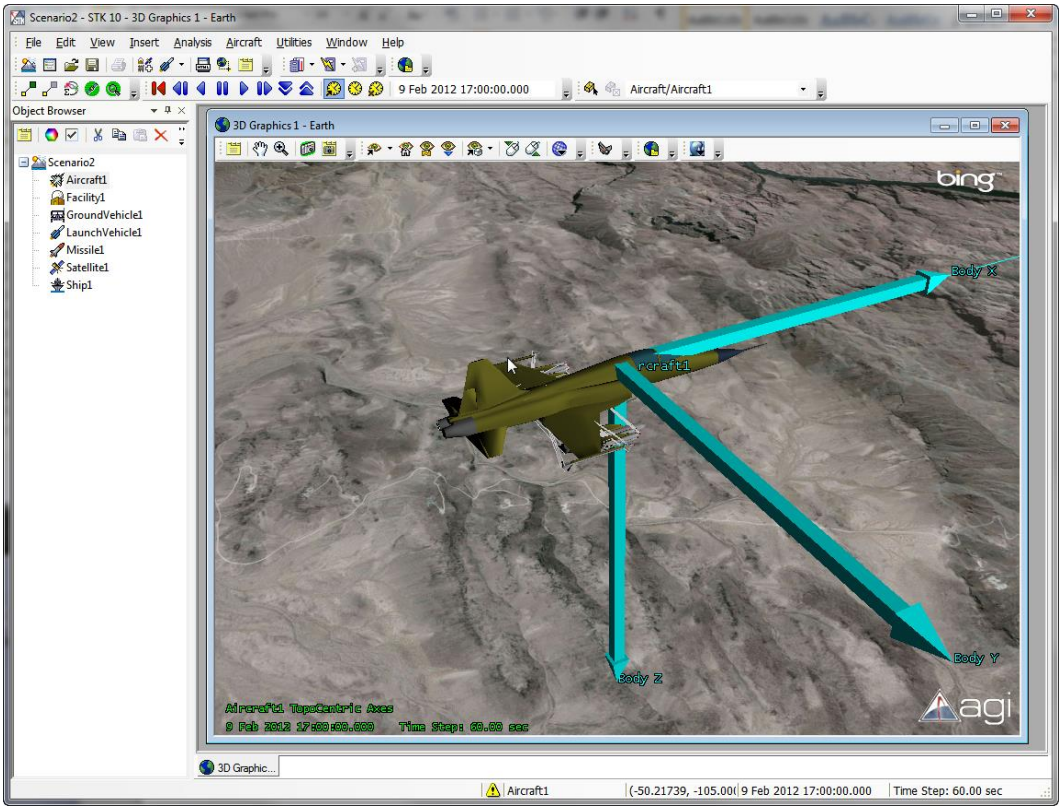
Nulls are used for all sorts of things in the 3D world.
But we need it as a locator for the sensor or effects attach point that we would like to add to our object.
Next, move, parent and re-name the Null. Use the transform widget to move the Null to where the sensor's origin is.
Note: *You do not have to be in "Edit" mode to move the Null.*



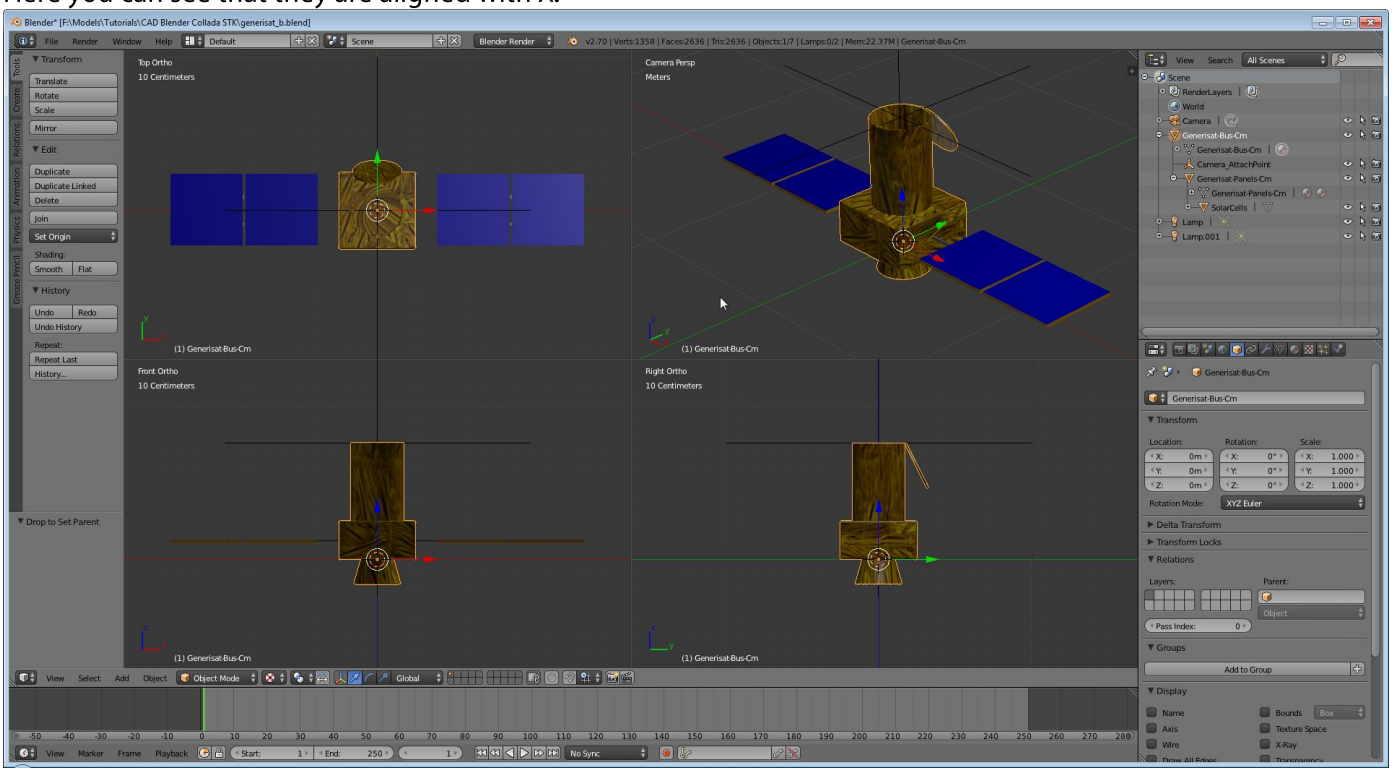
AXIS ALIGNMENT

Blender exports a Collada file with a Z-up axis. The default up axis in STK is also Z.
But depending on the type of object imported into STK, the axis alignment for each type will be different.
In setting up articulations, it is also important to have correct axis alignment. If not roll will be pitch and yaw will be roll. Not what we want.
Aircraft type axis alignment. X is aligned with the nose of the aircraft, Z axis points nadir.

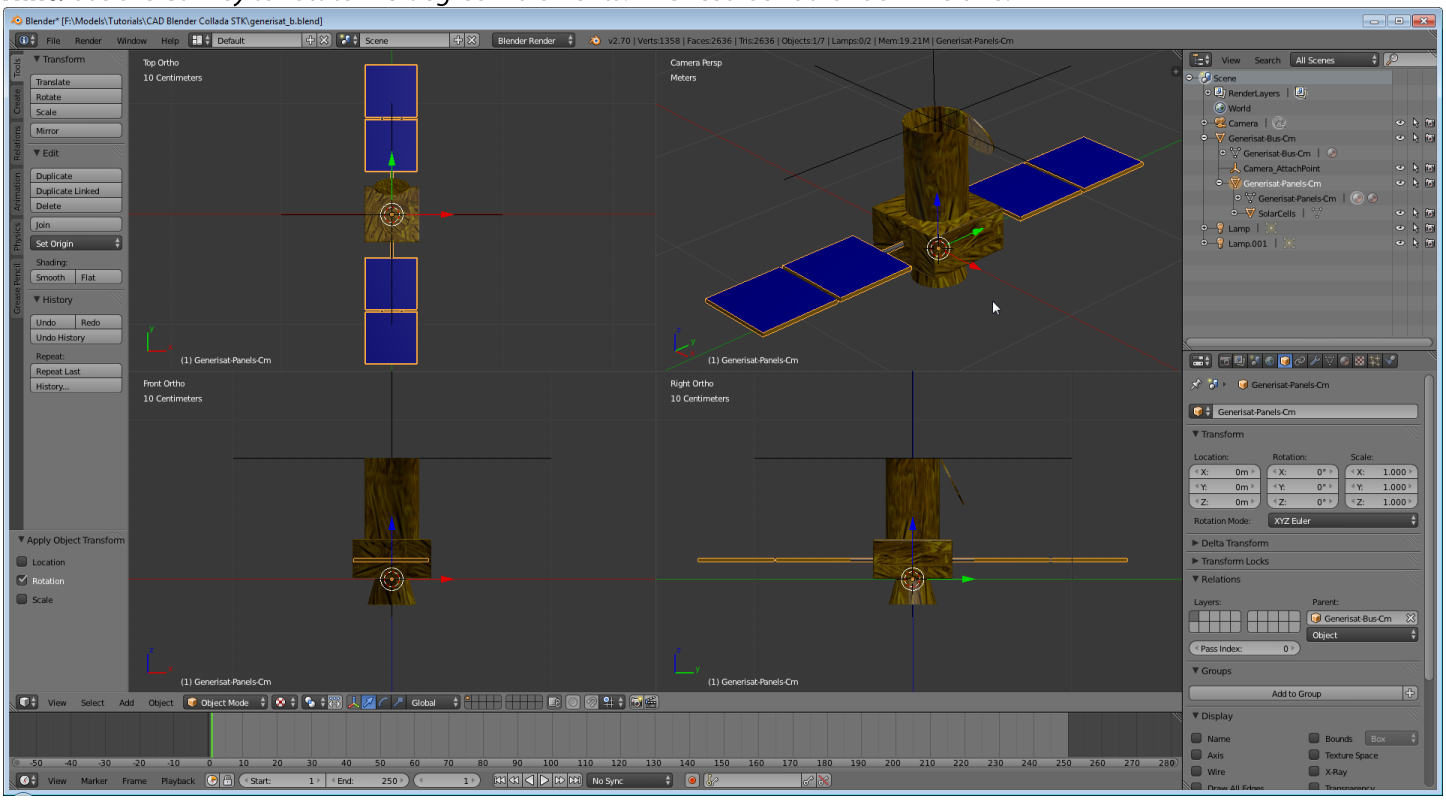




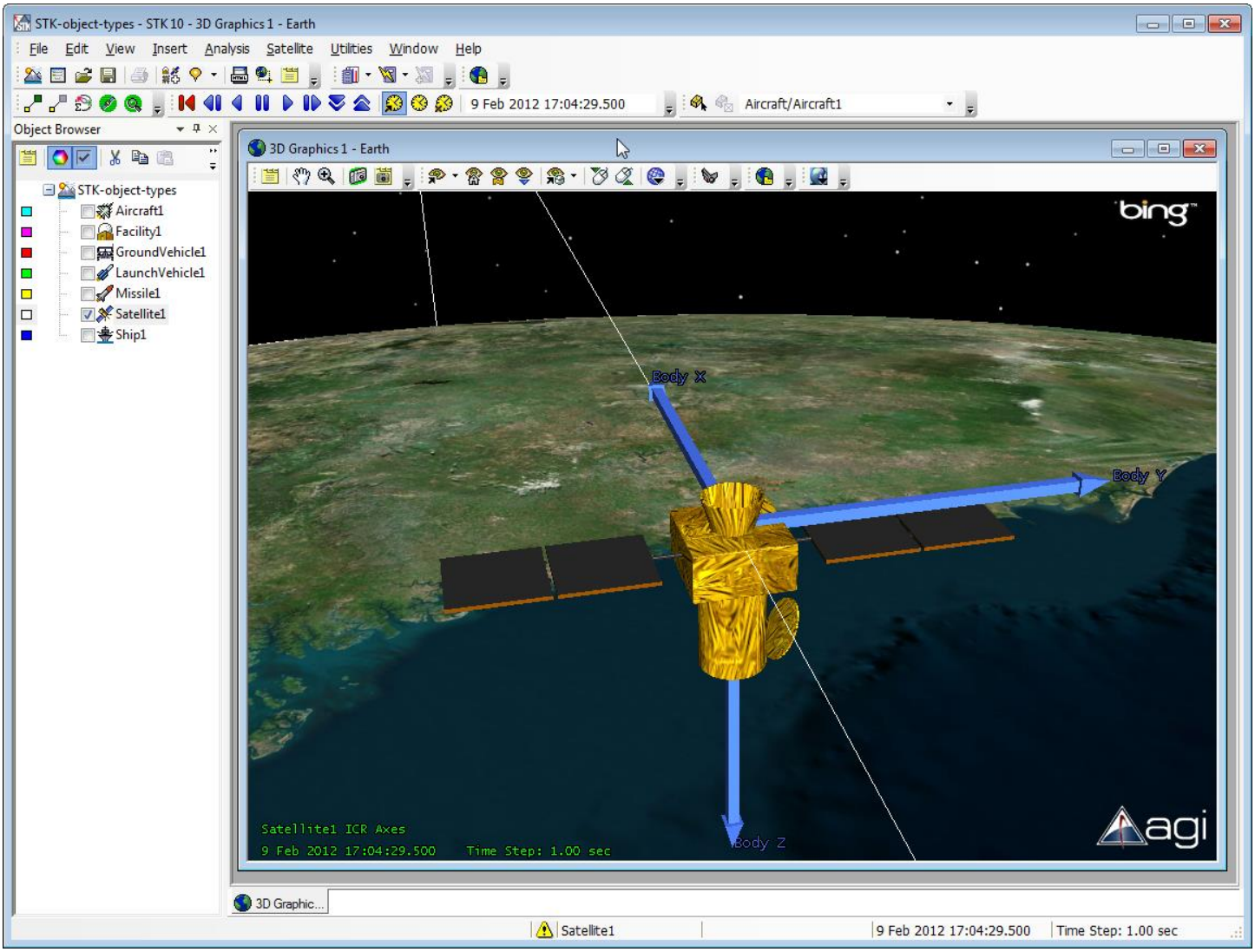
Satellite axis alignment. X is aligned with the velocity vector, Z axis points nadir.
Currently our' panels are not aligned correctly, as they should be in line with the Y axis.
Here you can see that they are aligned with X.



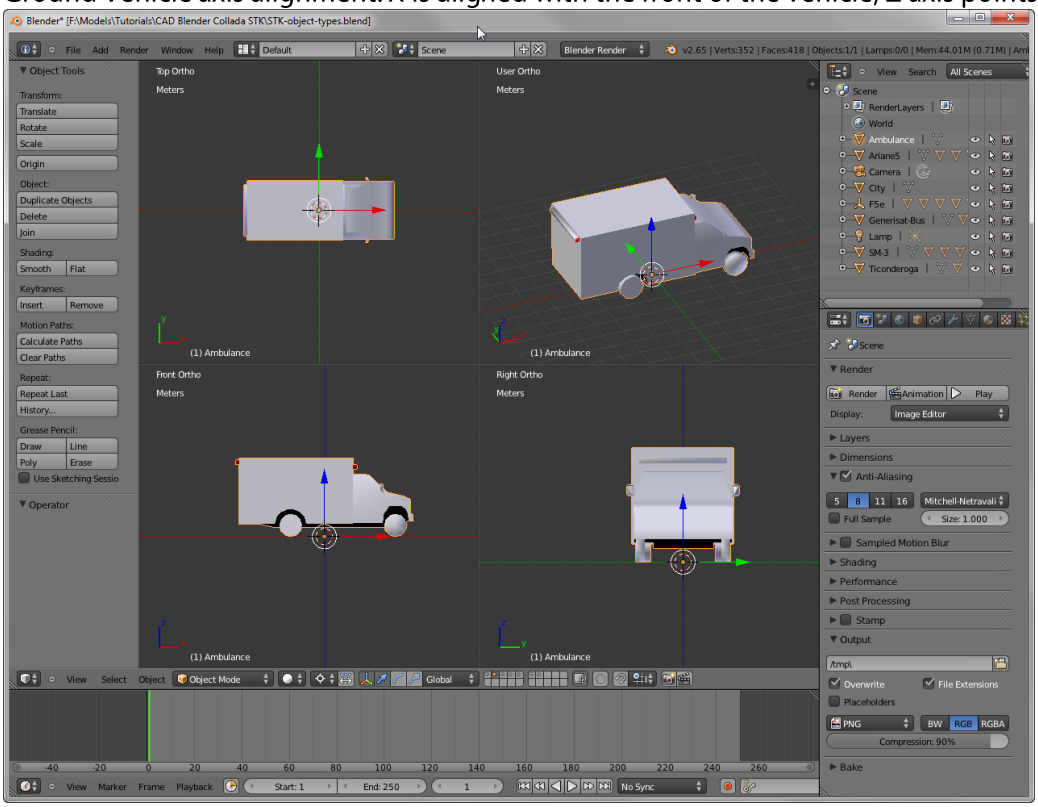
Select the layer that contains the panels. We are going to rotate them around the Z axis by 90 degrees.
Move the cursor to the upper left view (Top View). Using "r", rotate the panels so that they are in line with the Y axis.
Hint: *Use the Ctrl key to rotate in 5 degree increments.* The result should look like this.

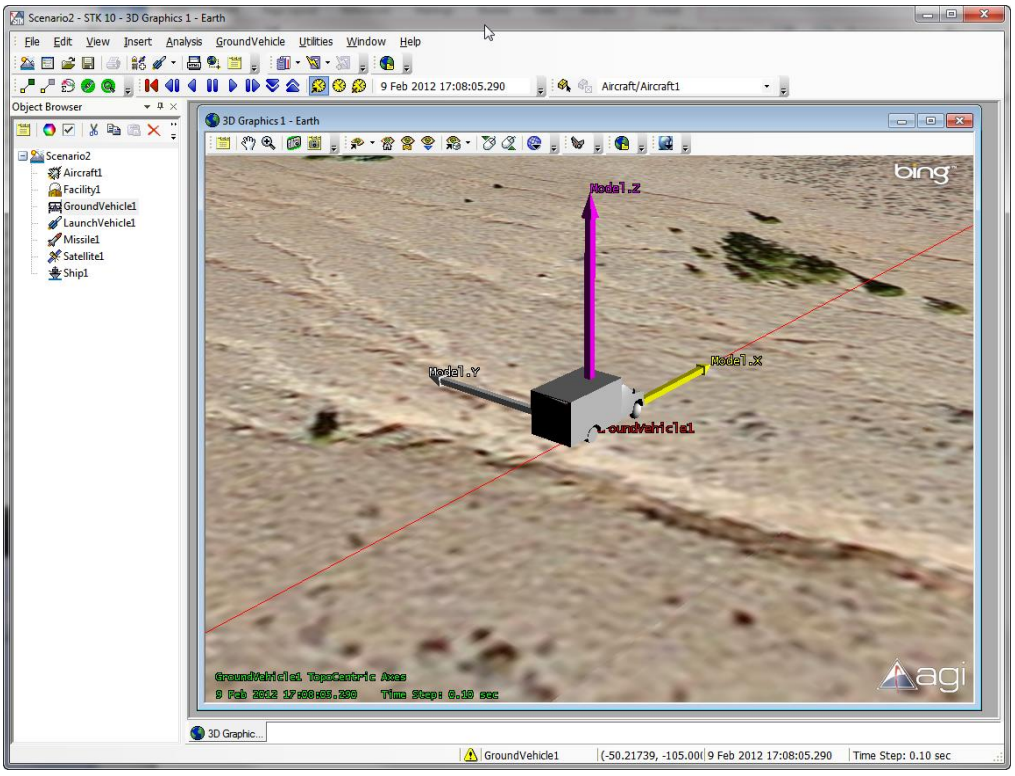


Make sure that you use "Ctrl+a" to reset the rotation in Blender.
Save the file as generisat_c.blend


STK-object-types - STK 10 - 3D Graphics 1 - Earth
File Edit View Insert Analysis Satellite Utilities Window Help
9 Feb 2012 17:04:29.500
Aircraft/Aircraft1
Object Browser
STK-object-types
Aircraft1
Facility1
GroundVehicle1
LaunchVehicle1
Missile1
Satellite1
Ship1
3D Graphics 1 - Earth
bing
Body X
Body Y
Body Z
Satellite1 ICR Axes
9 Feb 2012 17:04:29.500 Time Step: 1.00 sec
agi
3D Graphic...
Satellite1
9 Feb 2012 17:04:29.500
Time Step: 1.00 sec

Ground Vehicle axis alignment. X is aligned with the front of the vehicle, Z axis points to zenith.





Facility axis alignment. X is aligned with the front of the building, Z axis points to zenith.
Launch Vehicle axis alignment. X is aligned with the nosecone, Z axis points to zenith.
Missile axis alignment. X is aligned with the front of the missile, Z axis points to zenith.
Ship axis alignment. X is aligned with the bow, Z axis points to zenith.

Once you're finished aligning your model depending on its STK object type, be sure to select Ctrl-a, to reset any rotations.

ARTICULATION SETUP

WHAT'S NEEDED?

A DECENT TEXT EDITOR.

NotePad++ is a good, all around editor with support for XML documents.
http://notepad-plus-plus.org/
We are going to use it to edit both the Collada and ancillary file.

CORRECT ALIGNMENT BASED ON MODEL TYPE

For example, the "Roll" articulation for an aircraft is around the X axis.
But, if the fuselage is aligned with the Y axis and then you perform a Roll articulation, the model will look like it is pitching.

CORRECT CENTER OR PIVOT LOCATION.

The pivot and or center must be in the location from which the articulation will take place.

AN ANCILLARY FILE – WHAT IS IT AND WHY DO I NEED IT.

The AGI Collada ancillary file is a support document that allows for common attributes normally found in MDL files, to be used by Collada files.
Collada files can be exported from many of today's 3D modeling applications, but they can't export Collada files with built in articulations and the like.

This is what the ancillary file is for.
It allows those using other common DCC apps like, 3D Studio Max, Maya, LightWave 3D, SketchUp and Blender to set up their Collada file so that it does contain those attributes normally found in MDL files.

Those attributes include;

Articulations: Articulations are components (e.g., rocket stages, landing gear, doors, and propellers) that move in various ways. The motion is controlled via an articulation script.

Pointable Elements: Pointable elements are components (e.g., radio dishes, cameras, and sun tracking panels) that can automatically point to and target other objects.

Solar Panels Groups: Solar Panel Groups are components defined as solar panels and assigned an efficiency value.

Effect Attach Points: Effect attach points are used for attaching vapor trails to the 3D model.

Sensor Attach Points: Sensor attach points are used for attaching STK sensors to the 3D model.

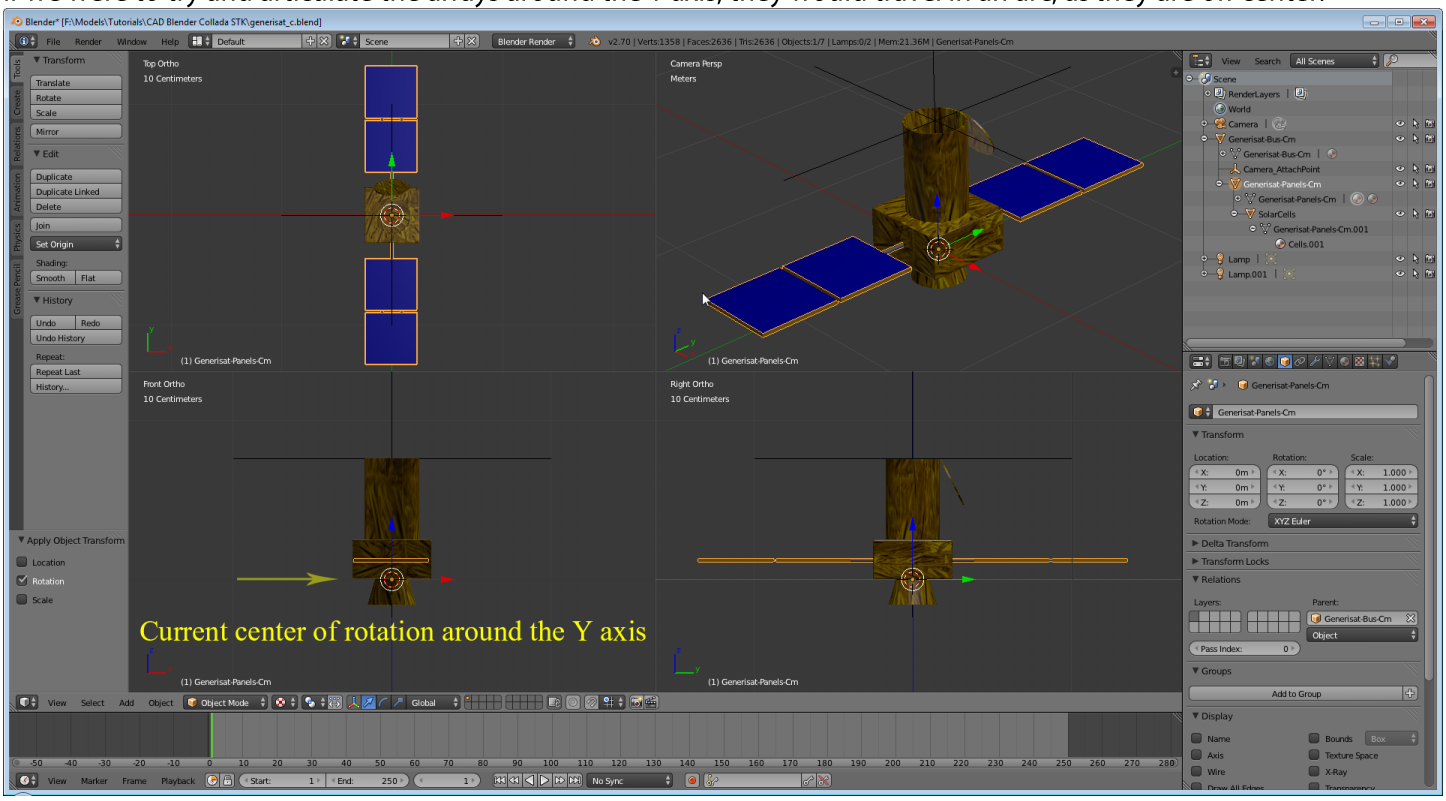
Non-Obscuring Material: Non-obscuring materials are components (e.g., secondary mirrors, and radar domes) that are invisible to sensors. This is useful when running the STK sensor obscuration tool.

A BASIC UNDERSTANDING OF THE COLLADA FORMAT.

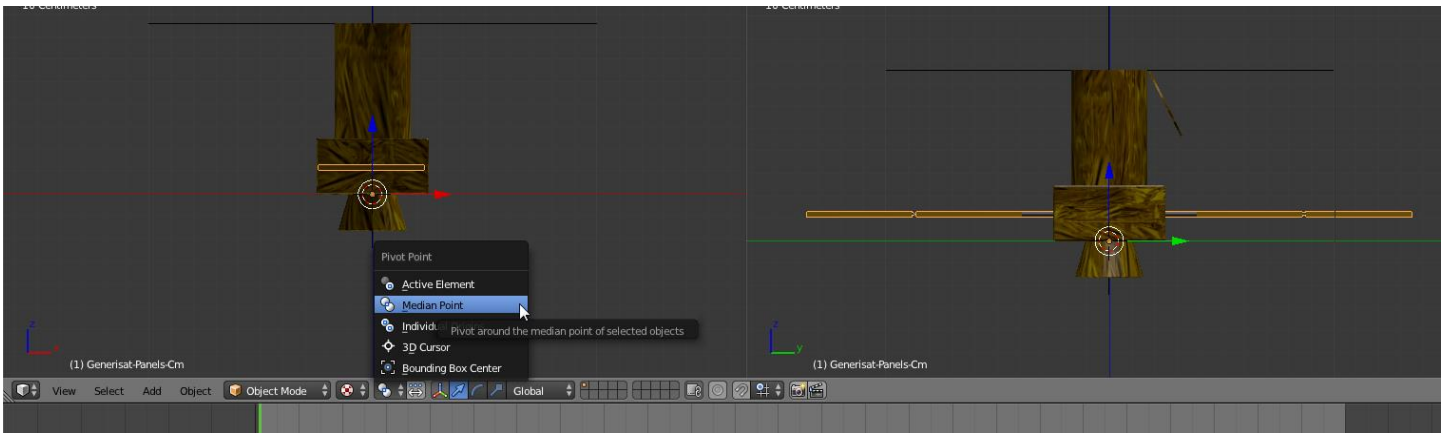
Collada files are formatted using container elements. Those elements are called Libraries.
There are Libraries for Cameras, Lights, Images, Effects, Materials, Geometry, Visual Scenes and user defined data.

Core libraries include.
Images. This lists textures applied to the model file
Effects. These are surface attributes such as face color, transparency, specularity, shininess etc.
Materials. Named list of materials used in the model file.
Geometries. Lists vertices, normals and texture coordinates for each mesh layer.
Visual_Scene. This where all the layers in the model file are brought together. This is the section that we will concentrate on the most. This section lists the "Nodes" that are needed for the ancillary file that will be used to articulate the model. The nodes name is the same as the layer name in Blender.

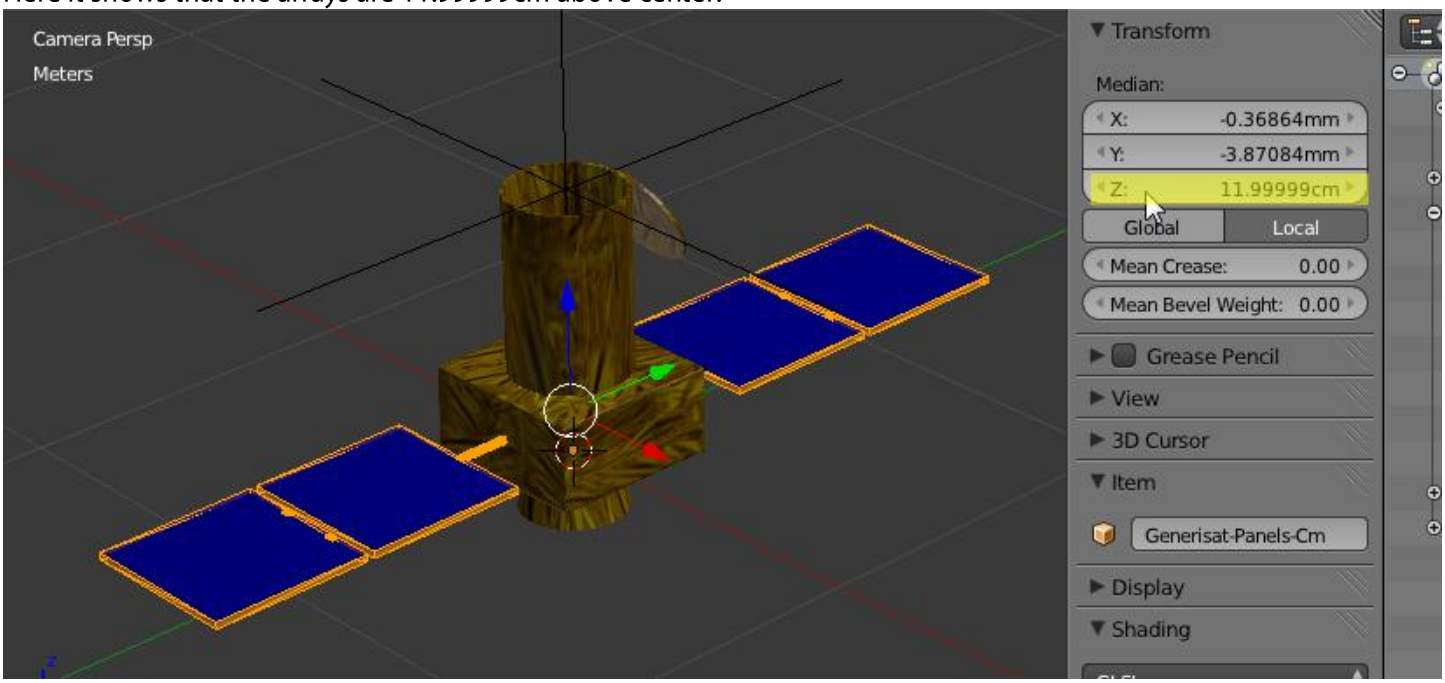
Before we get into the creation of an ancillary file, let's prepare the panels for correct rotation.
If we were to try and articulate the arrays around the Y axis, they would travel in an arc, as they are off center.



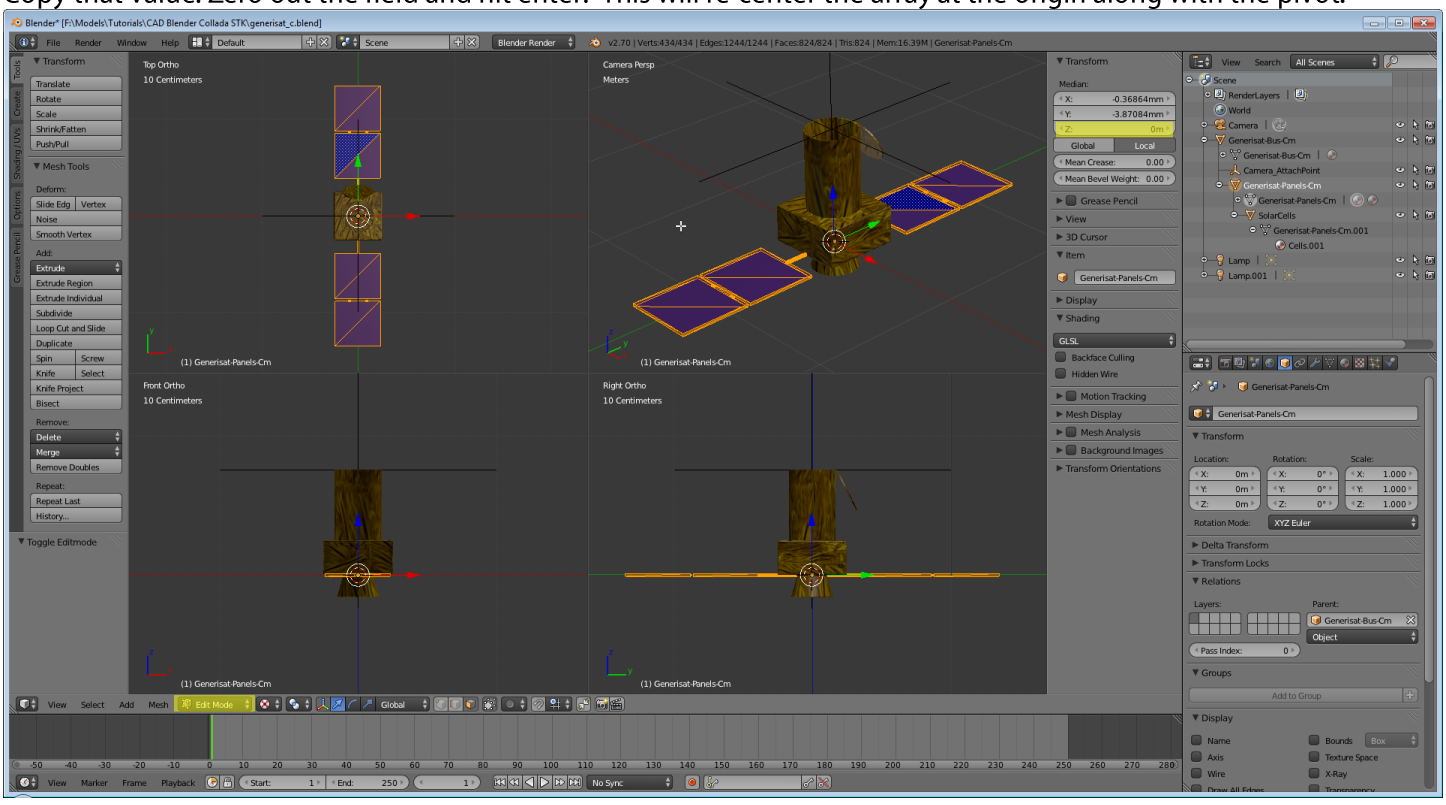
Let's look at how the array can be re-centered about the pivot point.
Make sure that the "Pivot Point" is set to "Median Point"



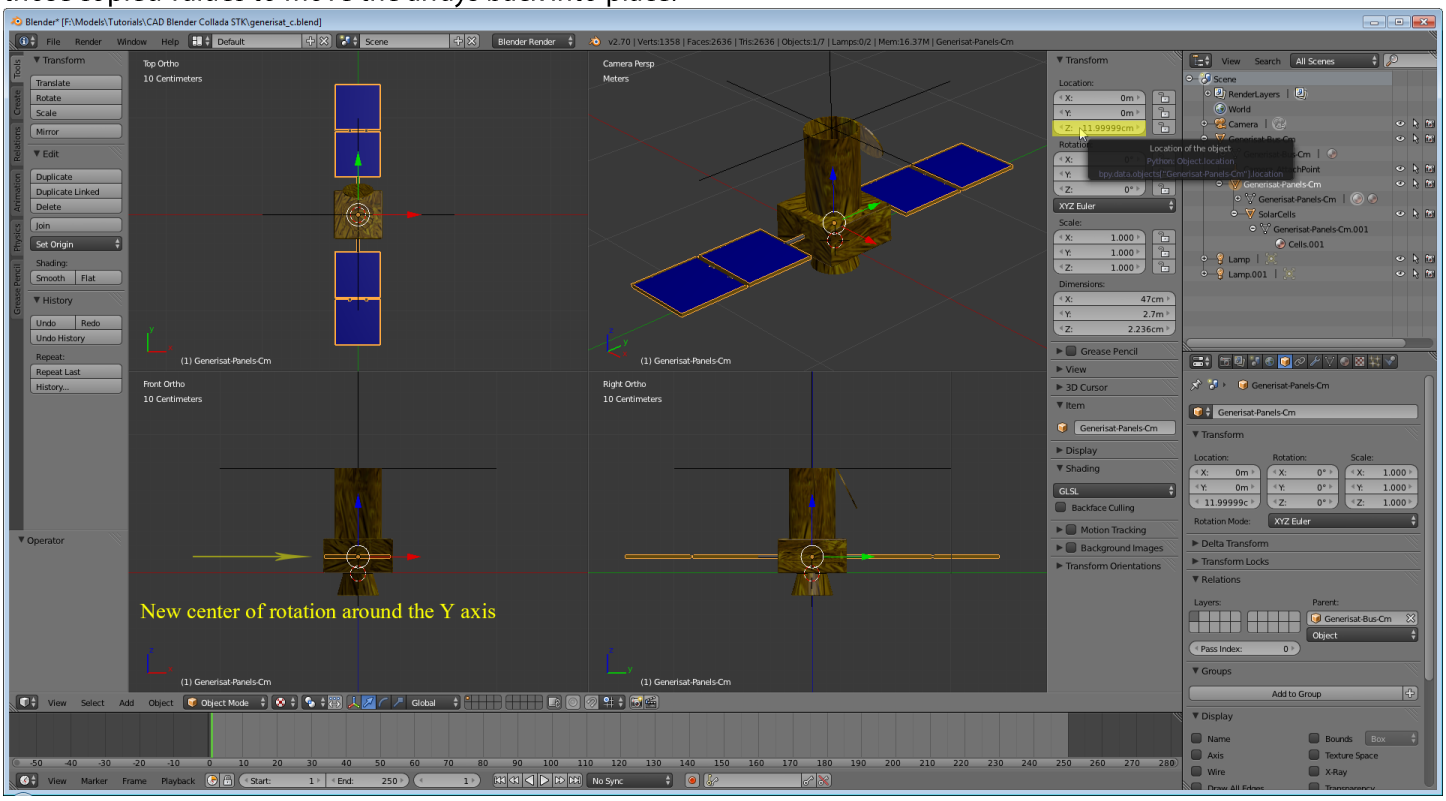
With the “Generisat-Panel-Cm” layer highlighted, hit the “Tab” key to go into “Edit Mode” and use “a” to select all.
The numeric panel on the right hand side gives us the info that we need.
(If hidden, hit the “n” key)
Here it shows that the arrays are 11.99999cm above center.



Copy that value. Zero out the field and hit enter. This will re-center the array at the origin along with the pivot.

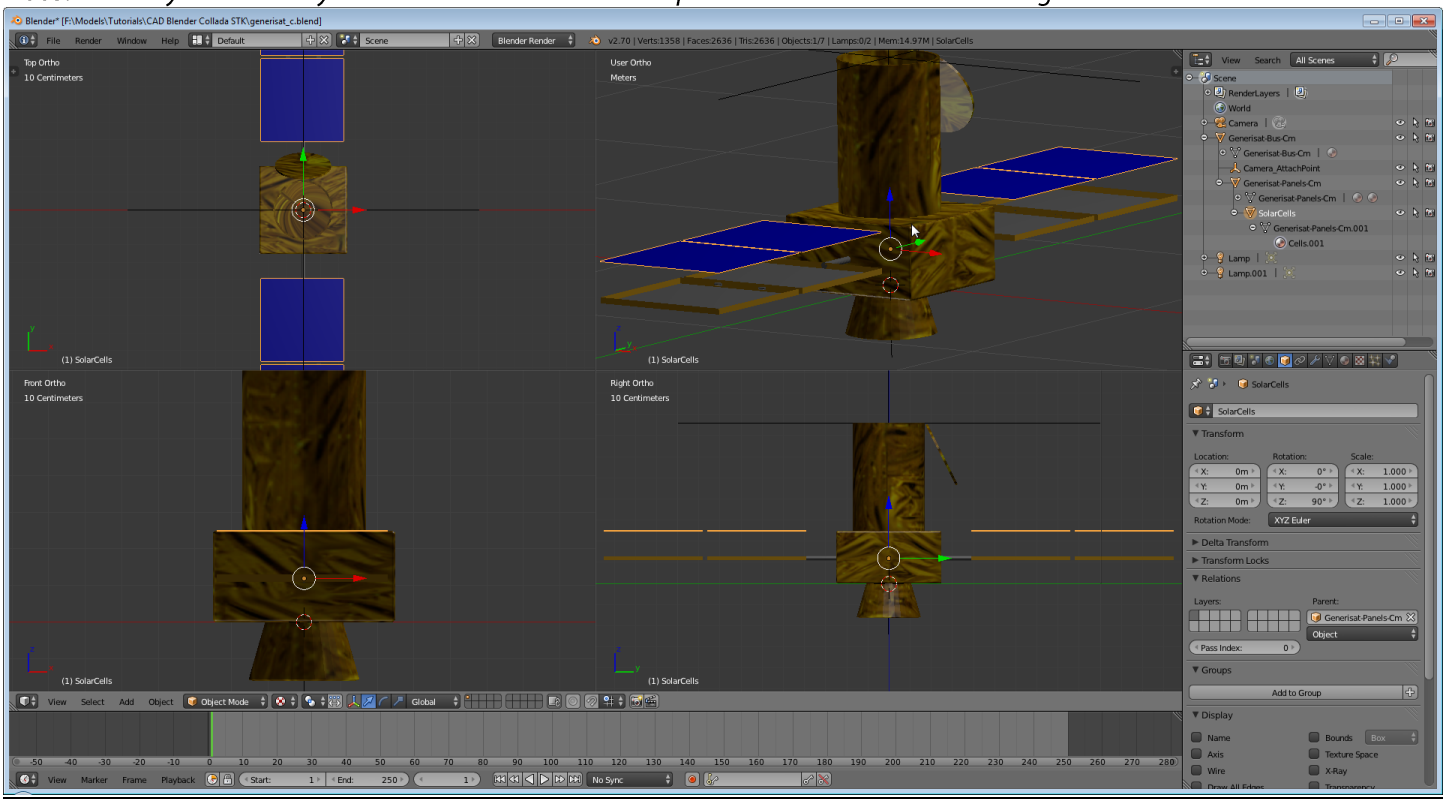


Go back into “Object Mode” (Tab key), in the “Transform” section, select “Location” from numeric panel and paste in those copied values to move the arrays back into place.

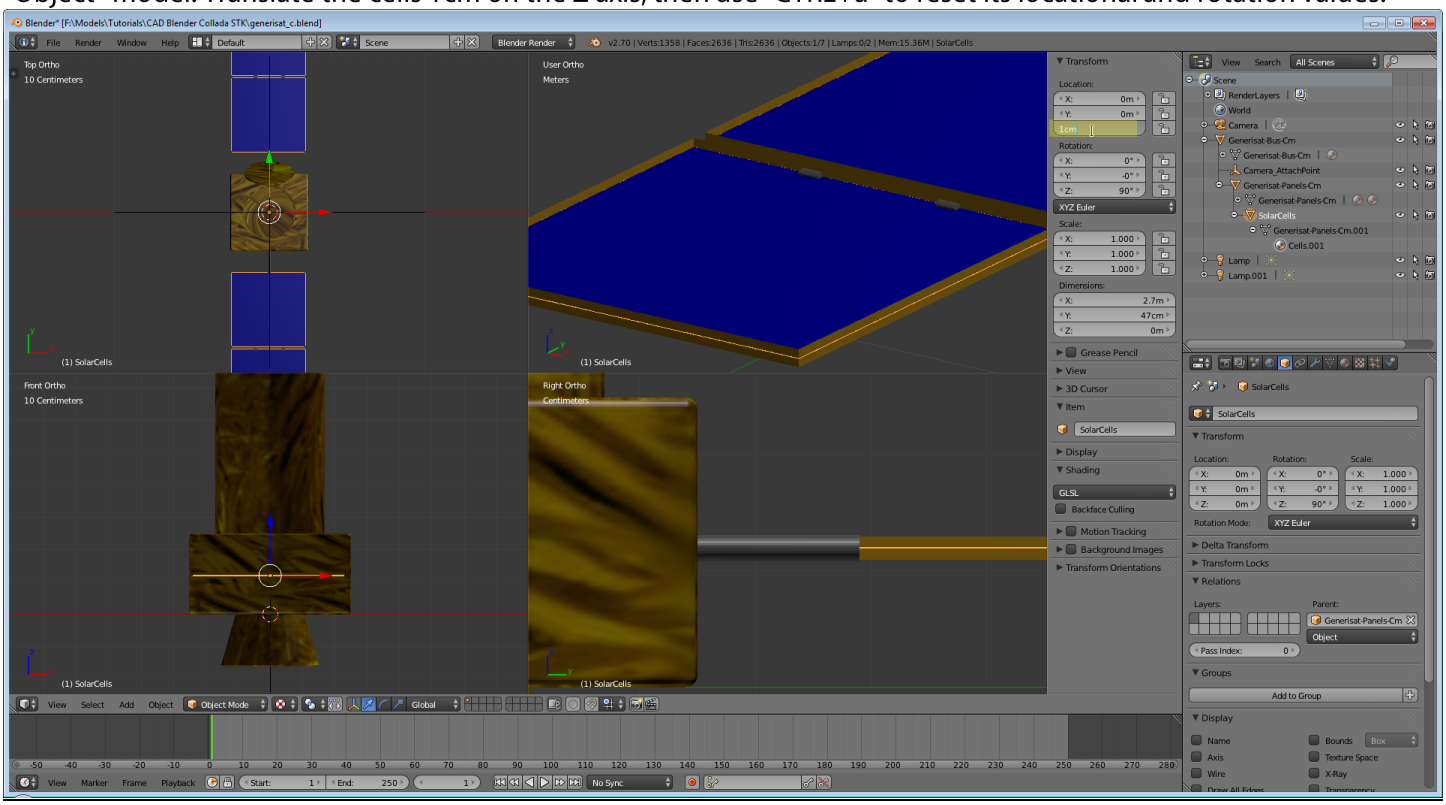


This translates the arrays back to their original position.
Important: *Do Not use “Ctrl+a” to reset the location or we’ll be right back where we started.*

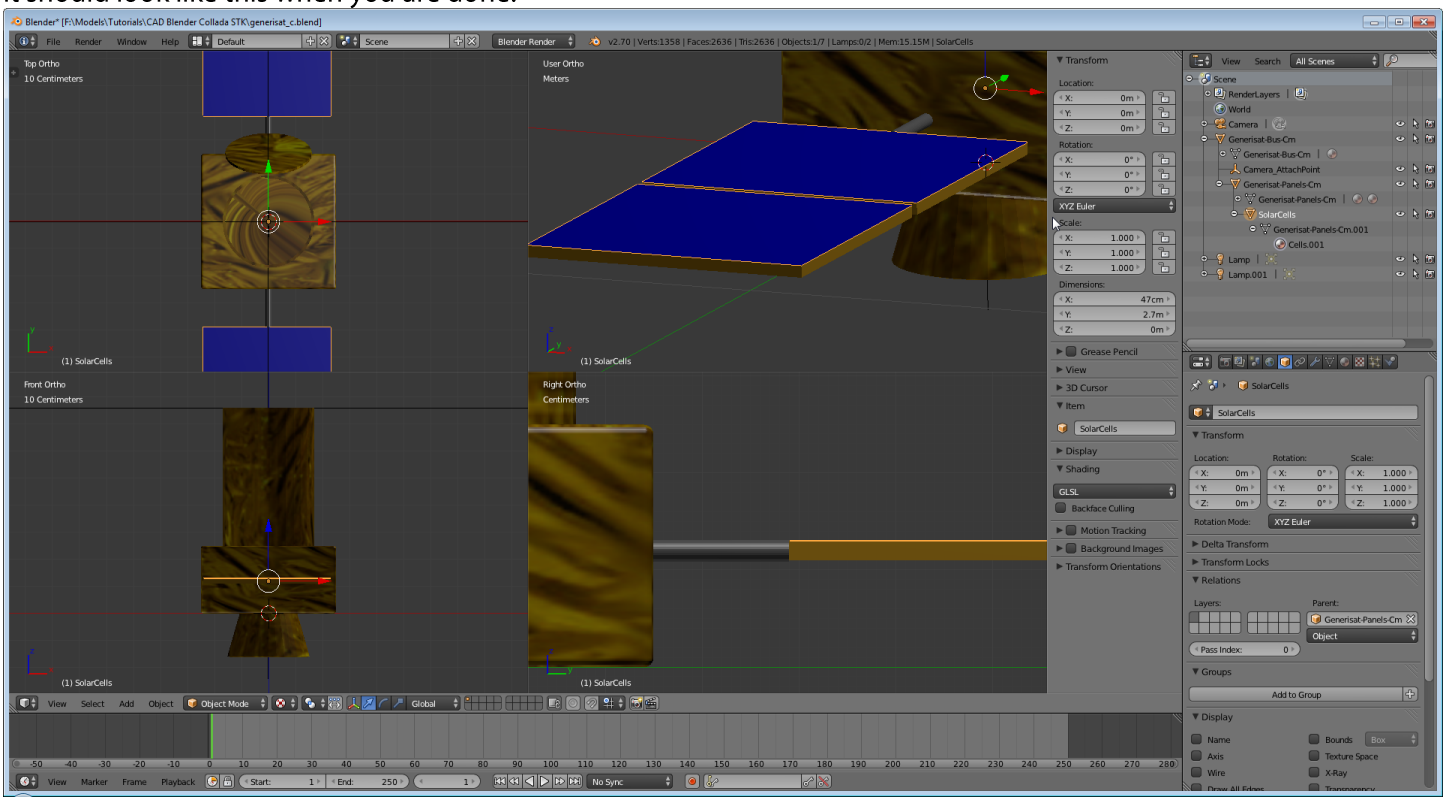
Note: *You may notice that you’ll also have to zero out the position of the “SolarCells” using the same method.*



Go back into “Edit” mode and use “a” to select all. In the “Transform” section, the positional values for the panels are 13cm. We only want to translate them12cm, the same amount as the arrays. So zero out the Z value and go back to “Object” model. Translate the cells 1cm on the Z axis, then use “CTRL+a” to reset its locational and rotation values.



It should look like this when you are done.

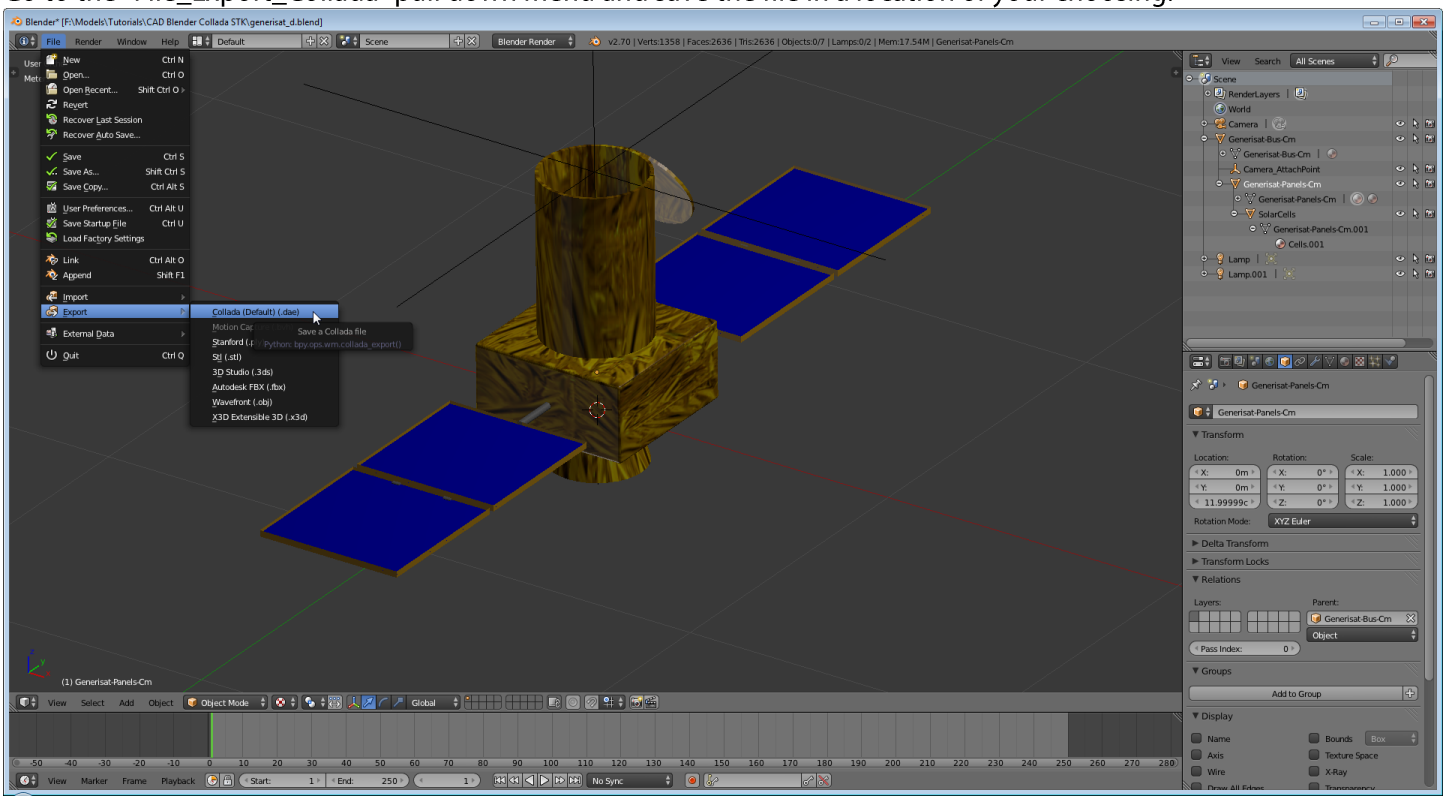


Now the arrays will articulation correctly around the newly centered pivot point.

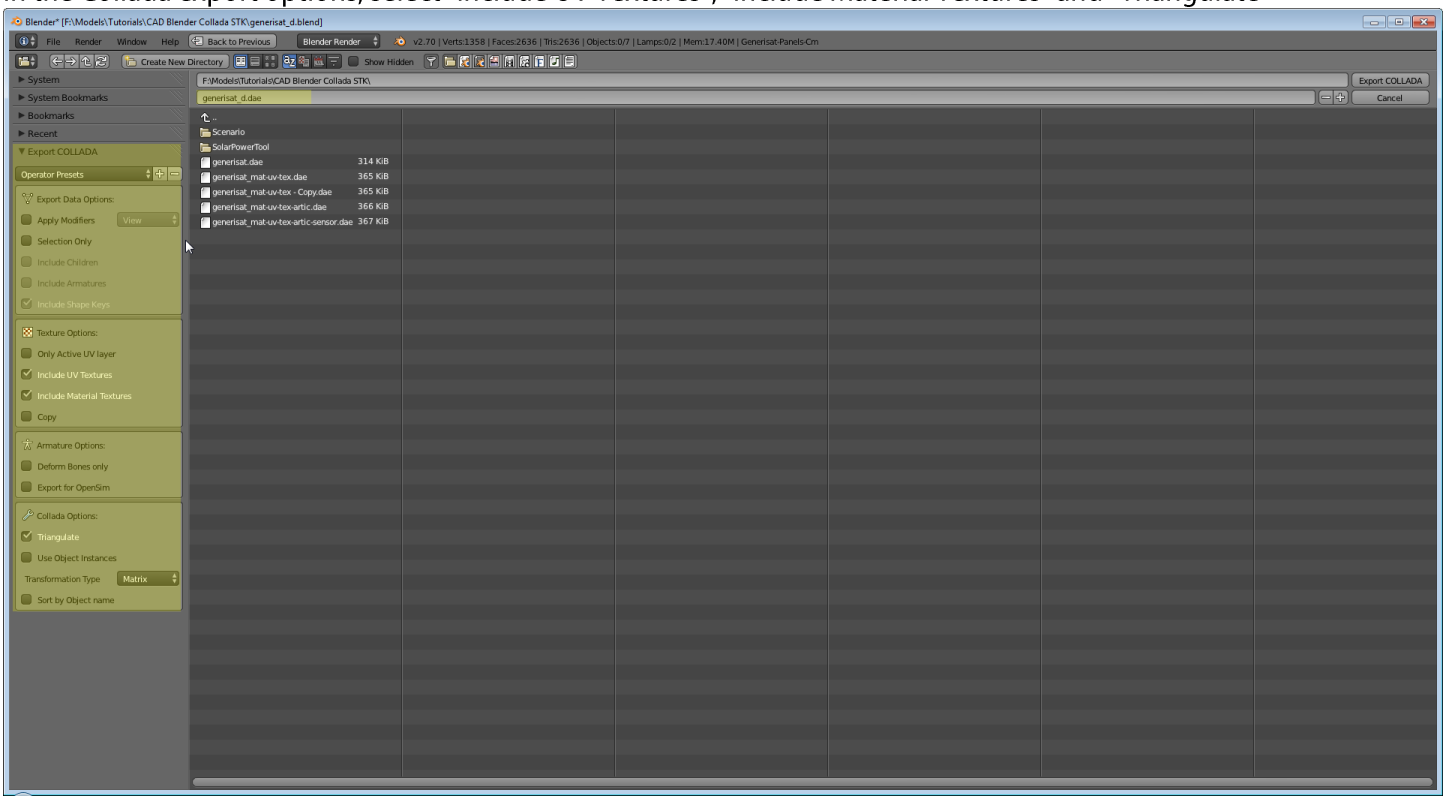
Save the blend file as generisat_d.blend

EXPORTING TO COLLADA

Go to the “File_Export_Collada” pull down menu and save the file in a location of your choosing.



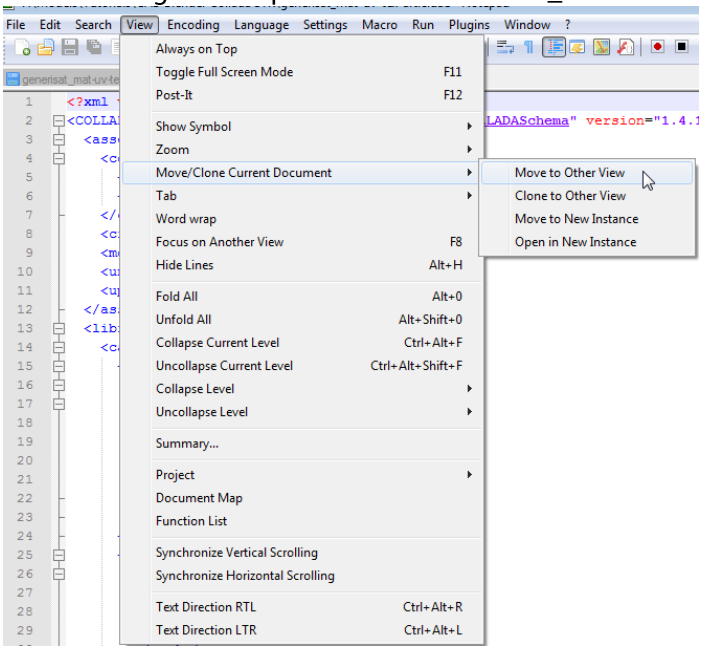
In the Collada export options, select “Include UV Textures”, "Include Material Textures" and “Triangulate”



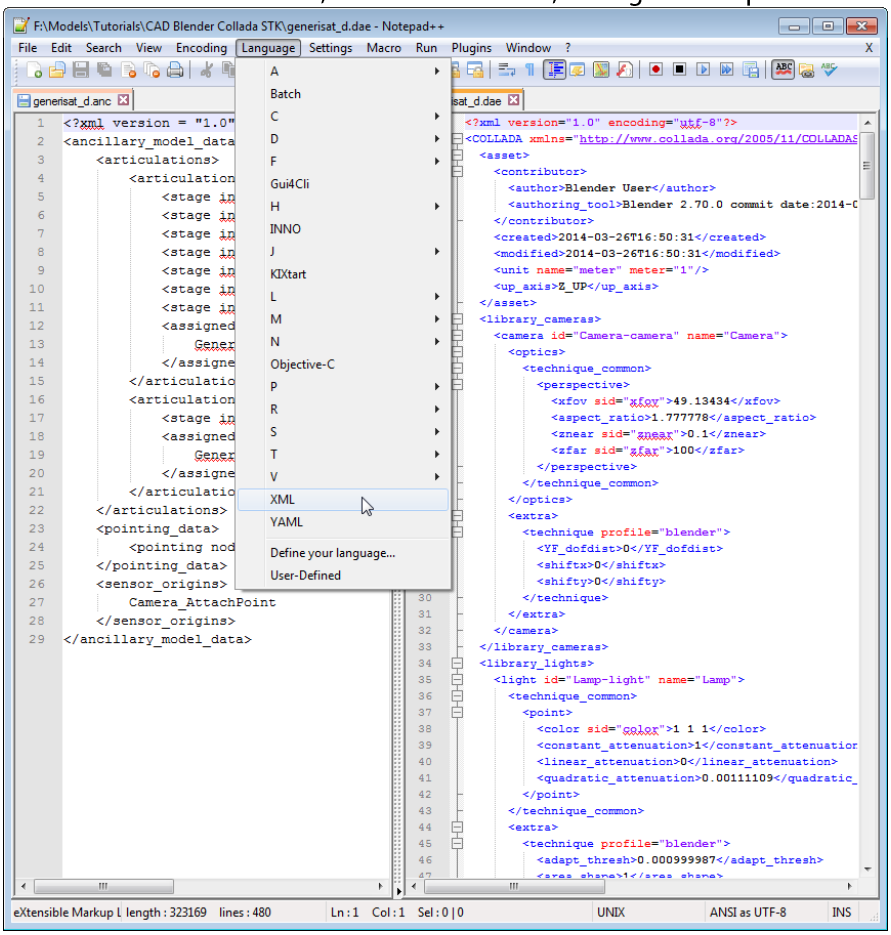
Give the file a name, i.e. generisat_d.dae and a location on the drive where you want the file stored.
Load the file into “STK” or "Modeler".

ANCILLARY FILE CREATION

We need to import both the Collada file and an ancillary file into Notepad++ so that we can setup articulations.
Notepad++ can be downloaded here http://notepad-plus-plus.org/
There is a master ancillary file that is included, but I have made one just for this model.
Open Notepad++
Import the files, generisat_d.dae and generisat_d.anc
By default, you can only see one file at a time.
To see both go to the pull down menu “View_Move/Clone Current Document_Move To Other View”



Now that we can see both, select the DAE file, then go to the pull down menu “Language_XML”



This allows us to view the Collada file in a tree like structure.
Beginning with the first library “cameras”, collapse the structure.
Then the “lights”, “images”, ”effects”, “materials” and “geometries”
At last we arrive at the library called “visual_scenes”
This is where we will find Node names that correspond to the layer names in Blender.
Skip past the first two, <node id="Camera" and <node id="Lamp"
The third one, <node id="Generisat-Bus-Cm" is the bus of our satellite and being the main parent, usually gets all the default articulations.
The next node, <node id="Generisat-Panels-Cm" is our arrays that we just reset the pivot of in Blender.
It is only going to need a rotation around the Y axis.
The next node is <node id="SolarCells". These are the panels that we separated into their own layer so that we can use them with STK’s Solar Panel Tool.
The last node is <node id="Camera_AttachPoint". This is the Null that we used to specify the origin of our sensor.

```
<?xml version="1.0" encoding="utf-8"?>
<COLLADA xmlns="http://www.collada.org/2005/11/COLLADASchema" version="1.4.1">
  <asset>
  <library_cameras>
  <library_lights>
  <library_images>
  <library_effects>
  <library_materials>
  <library_geometries>
  <library_controllers/>
  <library_visual_scenes>
    <visual_scene id="Scene" name="Scene">
      <node id="Camera" name="Camera" type="NODE">
      <node id="Lamp" name="Lamp" type="NODE">
      <node id="Generisat-Bus-Cm" name="Generisat-Bus-Cm" type="NODE">
        <matrix sid="transform">1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1</matrix>
        <instance_geometry url="#Generisat-Bus-Cm-mesh">
        <node id="Generisat-Panels-Cm" name="Generisat-Panels-Cm" type="NODE">
          <matrix sid="transform">1 0 0 0 0 1 0 0 0 0 1 0.1199999 0 0 0 1</matrix>
          <instance_geometry url="#Generisat-Panels-Cm-mesh">
          <node id="SolarCells" name="SolarCells" type="NODE">
            <matrix sid="transform">1.94707e-7 -1 0 0 1 1.94707e-7 0 0 0 0 1 0 0 0 0 1</matrix>
            <instance_geometry url="#SolarCells-mesh">
          </node>
        </node>
        <node id="Camera_AttachPoint" name="Camera_AttachPoint" type="NODE">
      </node>
      <node id="Lamp_001" name="Lamp_001" type="NODE">
    </visual_scene>
  </library_visual_scenes>
  <scene>
    <instance_visual_scene url="#Scene"/>
  </scene>
</COLLADA>
```

Let's look at the ancillary file.
We are going to use this ancillary file to write out the desired elements and assign the appropriate "node".
This may look different from what you are used to seeing in an MDL file, but it is the same stuff just written differently.

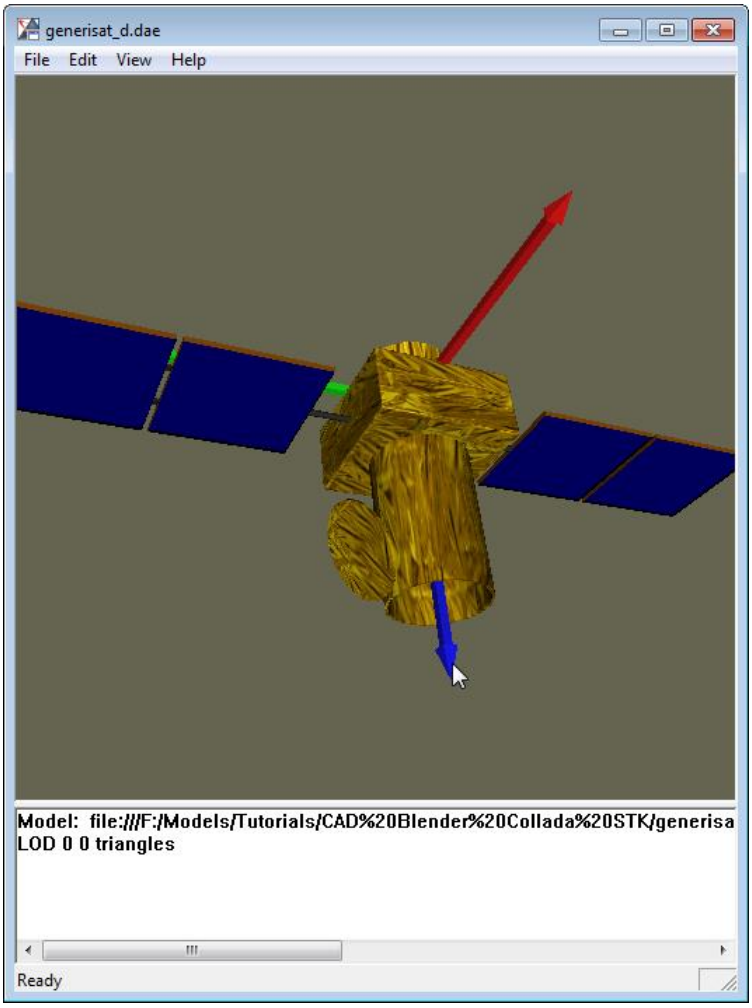
```
1  <?xml version = "1.0" standalone = "yes"?>
2  <ancillary_model_data version = "1.0">
3      <articulations>
4          <articulation name = "Generisat" type = "transform">
5              <stage init = "1" max = "1" min = "0" name = "Size" type = "uniformScale" />
6              <stage init = "0" max = "360" min = "-360" name = "Roll" type = "xRotate" />
7              <stage init = "0" max = "360" min = "-360" name = "Pitch" type = "yRotate" />
8              <stage init = "0" max = "360" min = "-360" name = "Yaw" type = "zRotate" />
9              <stage init = "0" max = "1000" min = "-1000" name = "MoveX" type = "xTranslate" />
10             <stage init = "0" max = "1000" min = "-1000" name = "MoveY" type = "yTranslate" />
11             <stage init = "0" max = "1000" min = "-1000" name = "MoveZ" type = "zTranslate" />
12             <assigned_nodes>
13                 Generisat-Bus-Cm
14             </assigned_nodes>
15         </articulation>
16         <articulation name = "Generisat-Panels" type = "transform">
17             <stage init = "0" max = "360" min = "-360" name = "Rotate" type = "yRotate" />
18             <assigned_nodes>
19                 Generisat-Panels-Cm
20             </assigned_nodes>
21         </articulation>
22     </articulations>
23     <pointing_data>
24         <pointing node = "Generisat-Panels-Cm" vector = "0 0 1" />
25     </pointing_data>
26     <solar_panel_groups>
27         <solar_panel_group efficiency = "28" name = "SolarPanels">
28             <assigned_nodes>
29                 SolarCells
30             </assigned_nodes>
31         </solar_panel_group>
32     </solar_panel_groups>
33     <sensor_origins>
34         Camera_AttachPoint
35     </sensor_origins>
36 </ancillary_model_data>
```

Line 4 is the "Articulations" that will be seen in "STK" or "Modeler". Here it is called "Generisat"
Lines 5-11 are the "Transformations" such as Roll, Pitch, Yaw, along with their rotational constraints.
Lines 12-14 define which "node" is to be used. In this case it is "Generisat-Bus-Cm", which we got from line 435 of the DAE file.
Note: *The ancillary file doesn't need the quotes for the "assigned_nodes"*

Next is the arrays articulation, which begins on line 16 of the ancillary file.
An articulation name of "Generic_Panels" is given.
Line 17 is the "Transformations" which allow for a rotation around the Y axis and is called "yRotate"
Line 19 defines the "node" to be used. In this case it is "Generisat-Panels-Cm", which we got from line 446 of the DAE file.

Next we've added a pointing command for sun tracking.
On line 24, we are still calling the same "node" for the arrays and we are giving it the Z vector to point on.

How do we know which vector to point along?
It is determined by looking at the model in Modeler.
Make sure that the "Axes" are showing. If they are not, go to the "View_Show Axes" pull down menu.
Then determine which axis is perpendicular to the face of the arrays.
In this case it is the blue Z axis.



Next up is the Solar Panel Group.
On line 27 we set the efficiency rating to 28% and define the "name" that will show up in STK's "Solar Panel Tool". Here, they are called "SolarPanel"
Line 29 defines the "node" to be used and in our case, it is "SolarCells", which we got from line 456 of the DAE file.

```
26      <solar_panel_groups>
27          <solar_panel_group efficiency = "28" name = "SolarPanel">
28              <assigned_nodes>
29                  SolarCells
30              </assigned_nodes>
31          </solar_panel_group>
32      </solar_panel_groups>
33      <sensor_origins>
34          Camera_AttachPoint
35      </sensor_origins>
36  </ancillary_model_data>
```

Lastly, on line 34 we define the "node" that will be used for our sensor attach point, which we got on line 467 of the DAE file.
Note: *The ancillary file must always be in the same directory and have the same name as the model.*

NOTES ON MODELER

Modeler is AGI's standalone model file viewer and is included with the installation of STK.
It can be found either from the Windows Start Menu listed as “Modeler 10”, or in the "bin" directory on the drive where STK was installed.
Typically on a PC this would be C:\Program Files (x86)\AGI\STK 10\bin
The name of the executable is AgMDE.exe
Modeler should be used only for viewing your model to make sure that the textures look good and to check the articulations of a model, if it has them.

Note: *Do not edit or save your model from within "Modeler".*
Those tools were designed to work with the old, original hand built models.
FYI: Yes, models used to be built by hand in a text editor one line at a time back in the day.

If any further editing is needed, go back to Blender and make your changes there.

TEXTURE PATHS

When the Collada model was exported, its associated images were given an absolute path in the Collada file.
Models tend to get moved around from one computer to the next.
We need to change this because models don’t stay in one place.
On line 174 is our texture reference

```
1   <?xml version="1.0" encoding="utf-8"?>
2   <COLLADA xmlns="http://www.collada.org/2005/11/COLLADASchema" version="1.4.1">
3     <asset>
13    <library_cameras>
34    <library_lights>
172   <library_images>
173     <image id="goldfoil4_tga" name="goldfoil4_tga">
174       <init_from>/F:/Models/Tutorials/CAD%20Blender%20Collada%20STK/goldfoil4.tga</init_from>
175     </image>
176   </library_images>
```

This directory path is where the textures will be looked for
We need to change this, so that no matter where the model is moved, the texture will be loaded.

Change line 174 to read:
<init_from>. /goldfoil4.tga</init_from>

Save the model from Notepad++ using Ctrl-s

Note: *It’s good practice to keep the textures in the same directory as the Collada model.*

Option: *If this is an STK installed texture, then this would be a path directly linked to STK’s textures directory and there would be no need to change it /C:/Program%20Files%20(x86)/AGI/STK%2010/STKData/VO/Textures/goldfoil4.tga*

COLLADA LIMITATIONS

The XML reading library libXML, which is used by the COLLADA DOM (which we use to ingest Collada), has a limit of 10 million characters within any xml tag.

Meaning, there must be less than 90,000 polygons per Component, aka Layer, aka Node.

The Collada specs are very wide and AGI's importer doesn't read every possible variation of that spec.

Results may vary.

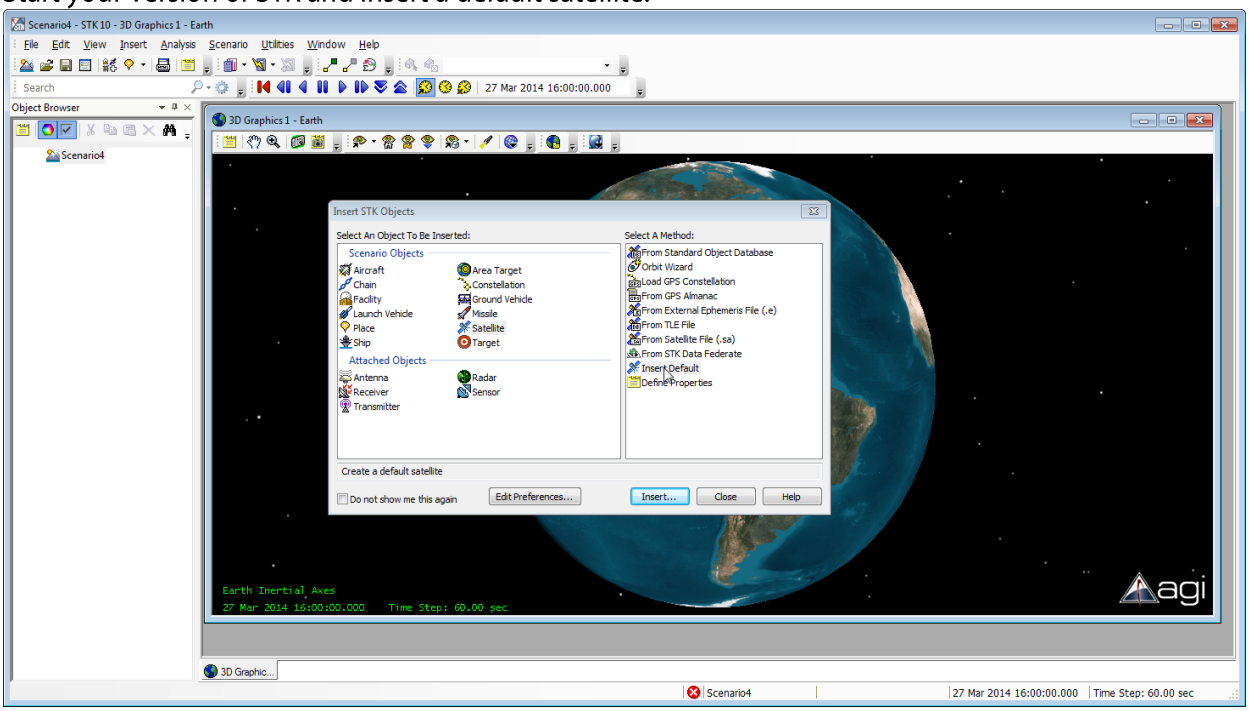
Any further questions and or comments should be addressed to support@agi.com
The subject line should include the word "Collada Help"

Scott Cross
Model Development Specialist
Analytical Graphics, Inc.
220 Valley Creek Blvd.
Exton, PA 19341-2380
www.agi.com
v. (610)981-8000
f. (610)981-8001

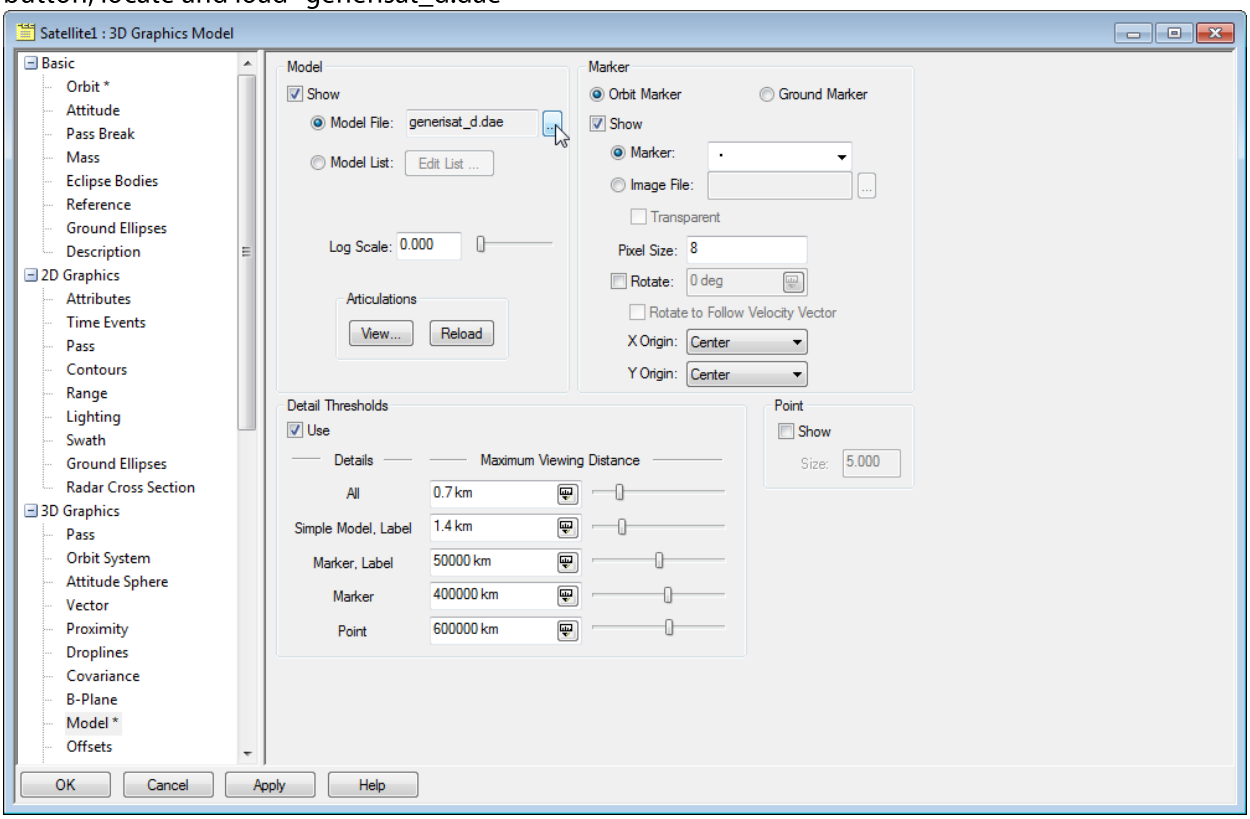
STK IMPLIMENTATION

INSERT THE SATELLITE

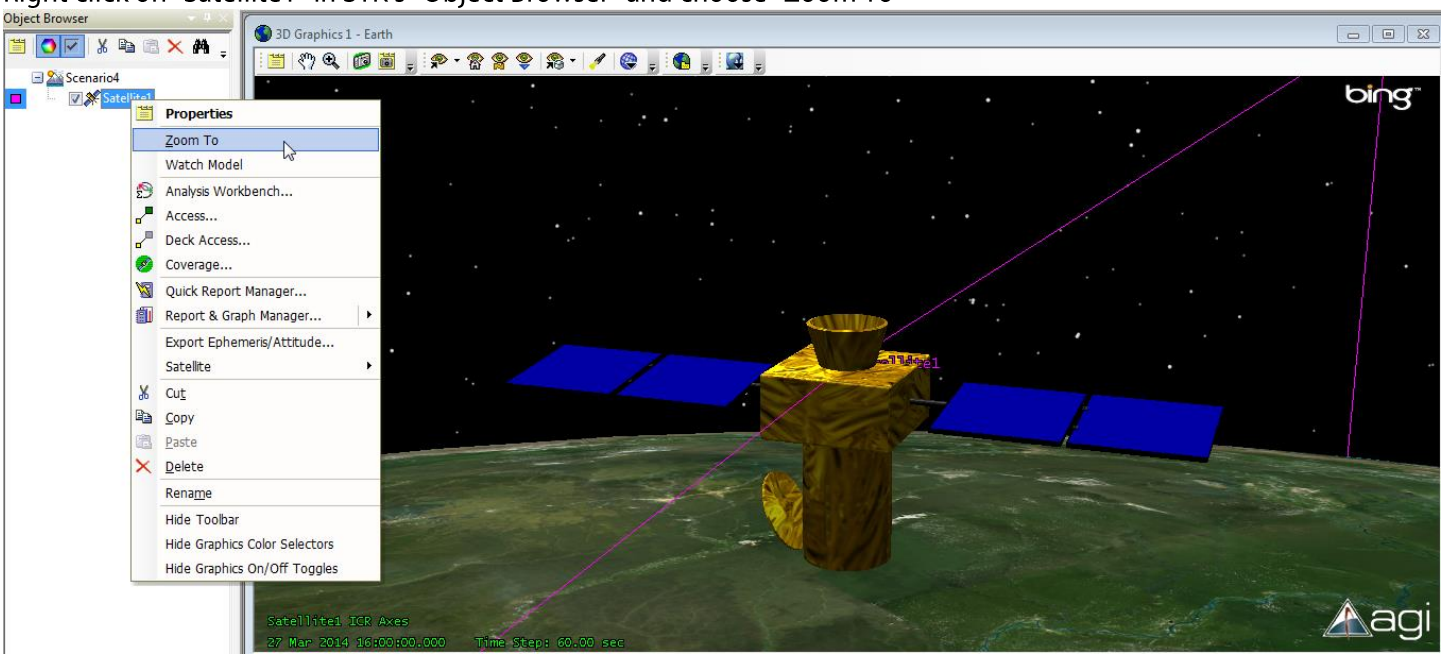
Now that the model is setup, let's load it up into STK, add a sensor and do some solar power analysis. Start your version of STK and insert a default satellite.



Now double click on "Satellite1" object and go to "3D Graphics". In that section select "Model" and click on the radio button, locate and load "generisat_d.dae"

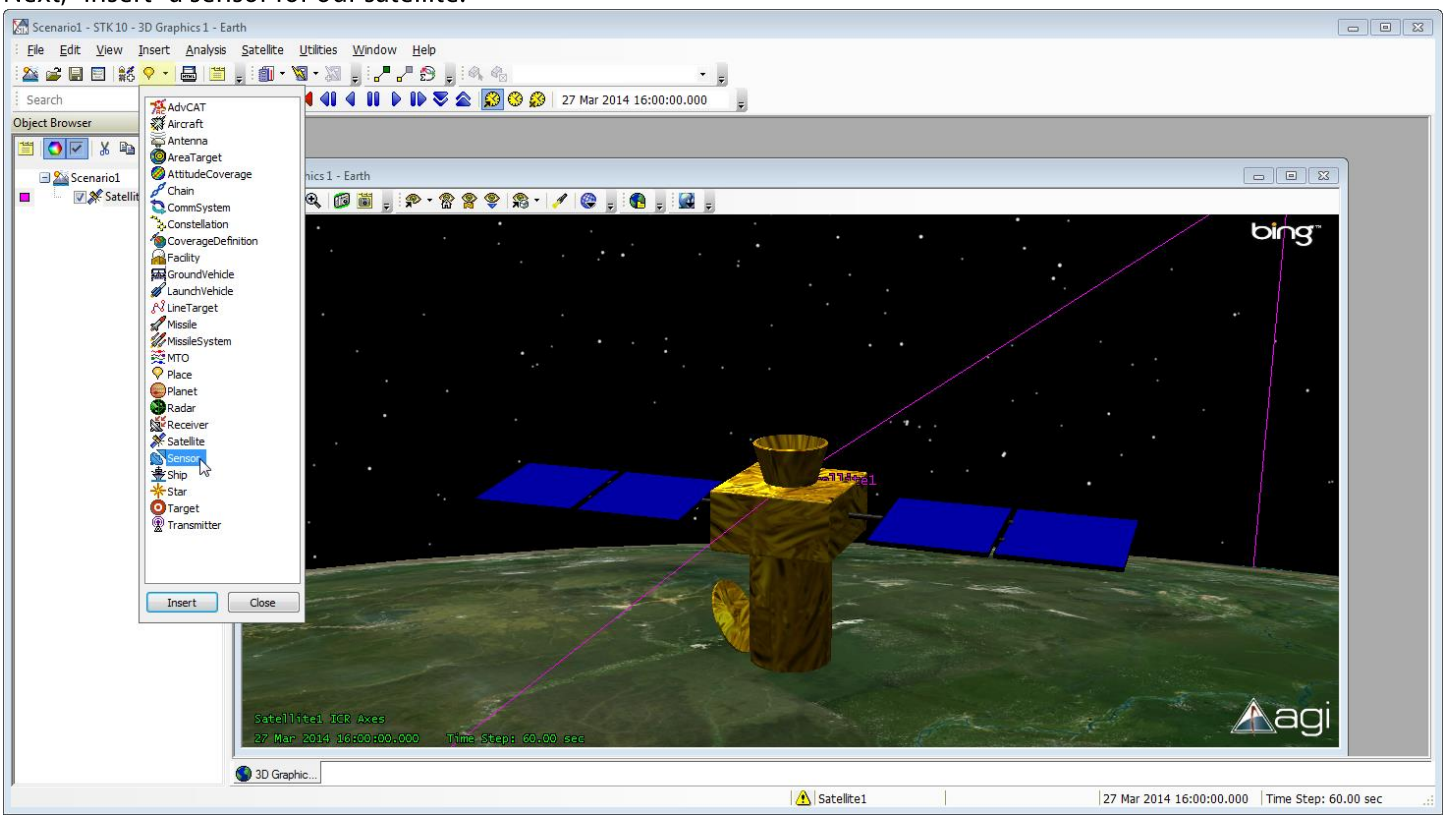


Right click on “Satellite1” in STK’s “Object Browser” and choose “Zoom To”

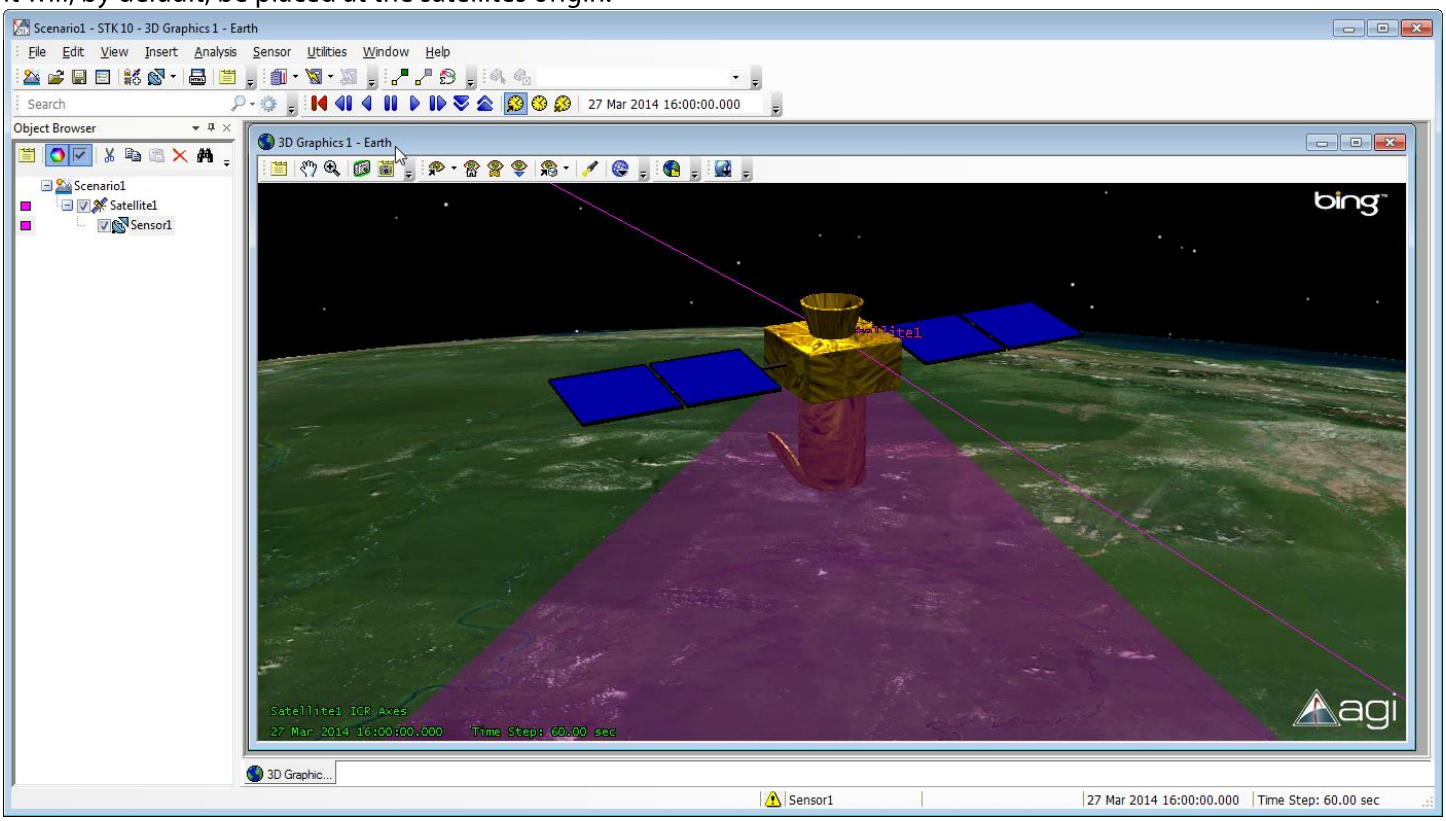


ADD THE SENSOR

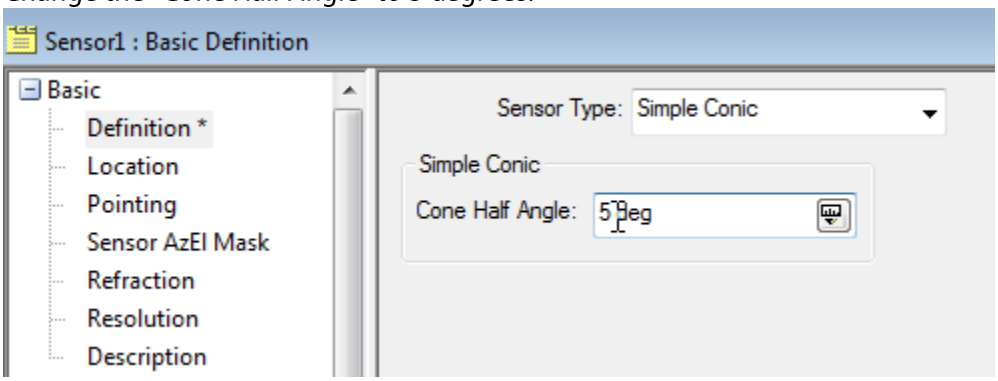
Next, “Insert” a sensor for our satellite.



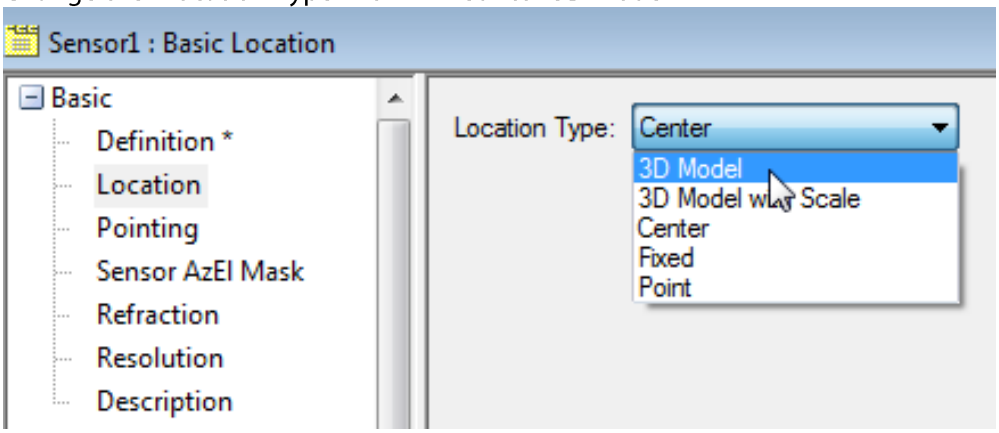
It will, by default, be placed at the satellites origin.



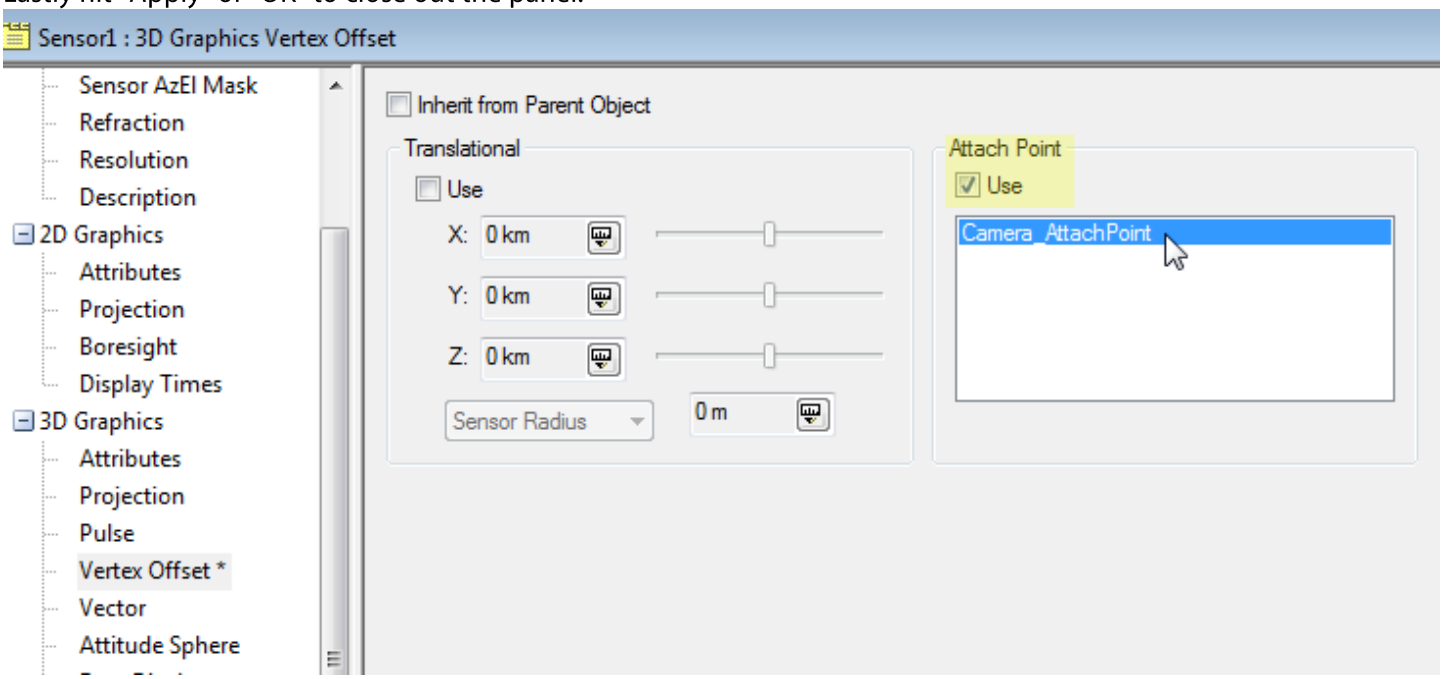
Let's attach "Sensor1" to the Null that we added to our satellite in Blender.
In STK's "Object Browser" double click on the object to bring up its properties.
Change the "Cone Half Angle" to 5 degrees.



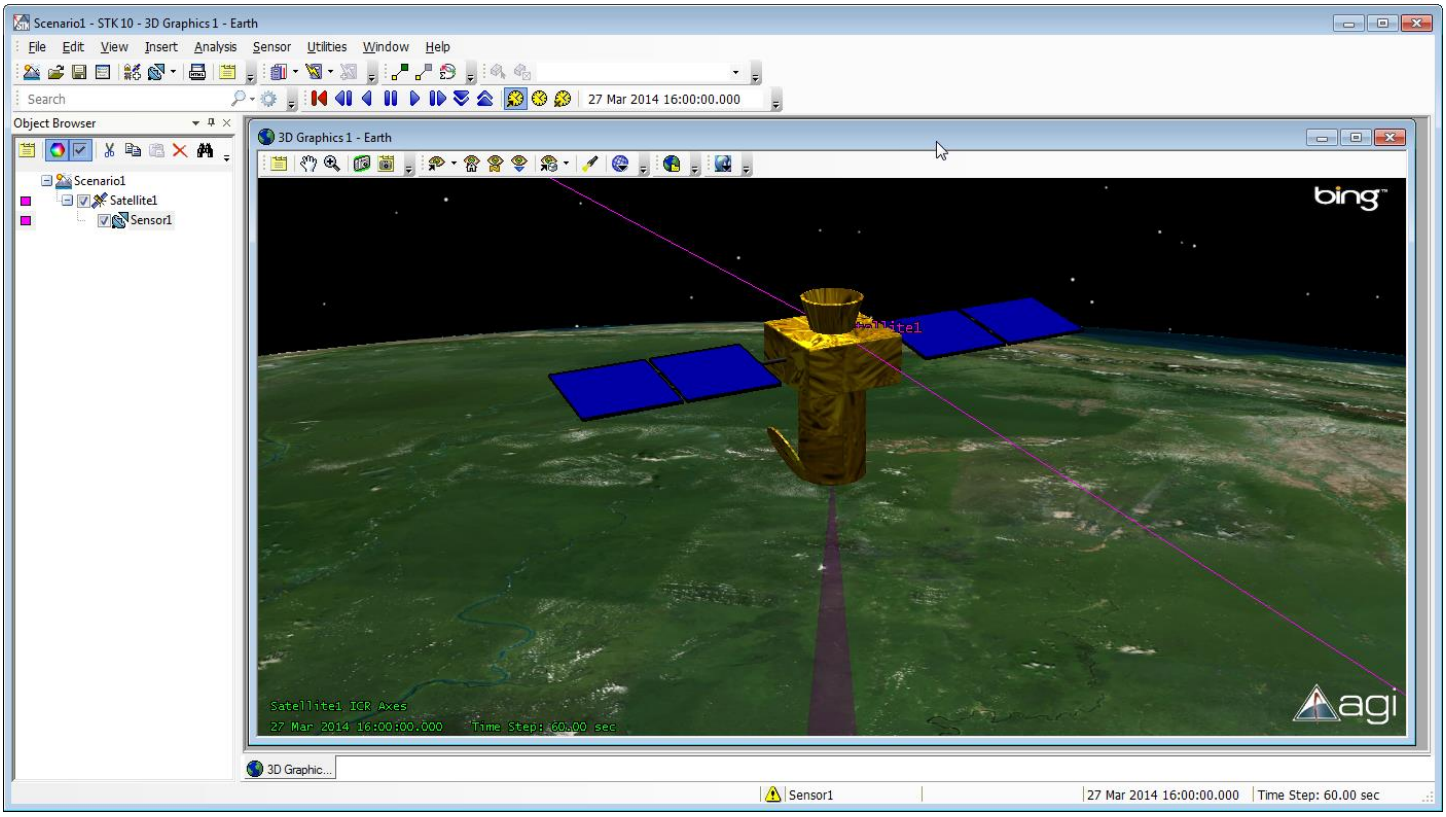
Change the "Location Type" from "Fixed" to "3D Model"



In the “Vertex Offset” section, check “Use” in the section titled “Attach Point”
Select the attach point that we named on line 34 of the ancillary file.
Lastly hit “Apply” or “OK” to close out the panel.



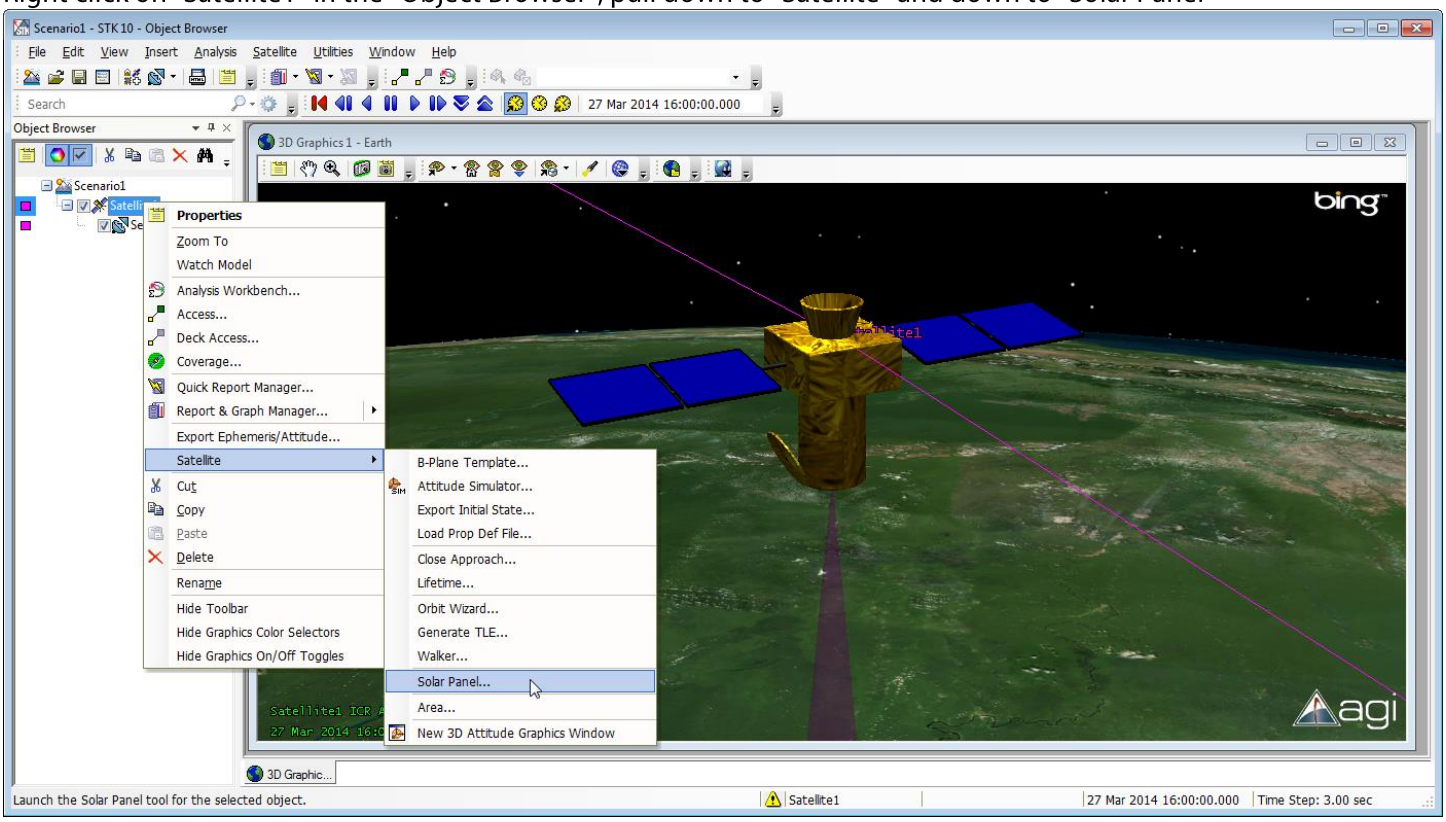
Here is the result.



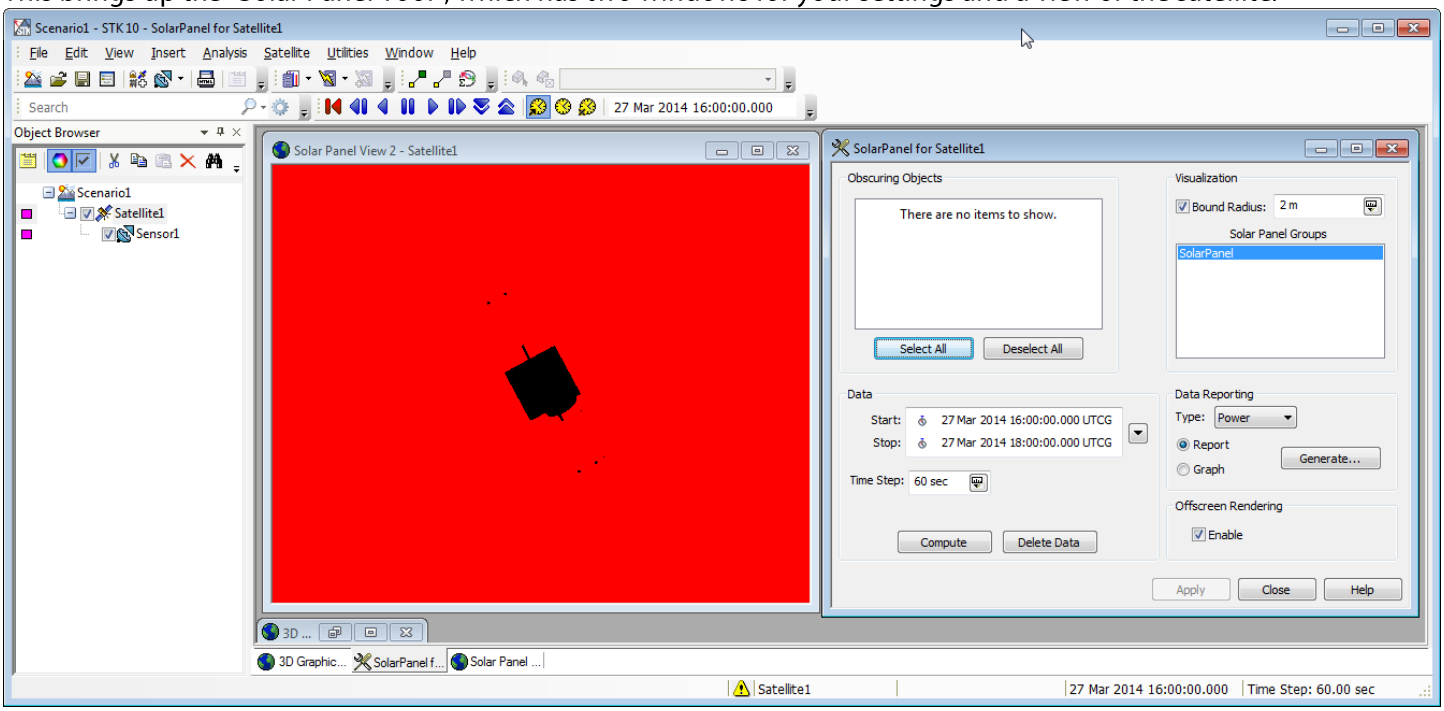
SOLAR PANEL TOOL

Reduce the "Time Step" to 5 seconds and play the scenario. You will notice that the arrays are already tracking the sun. This is due to the fact that one, we created an articulation for the arrays, two, set up a pointing command and three, set up solar panel groups. This won't happen unless these three conditions are met.

Right click on "Satellite1" in the "Object Browser", pull down to "Satellite" and down to "Solar Panel"



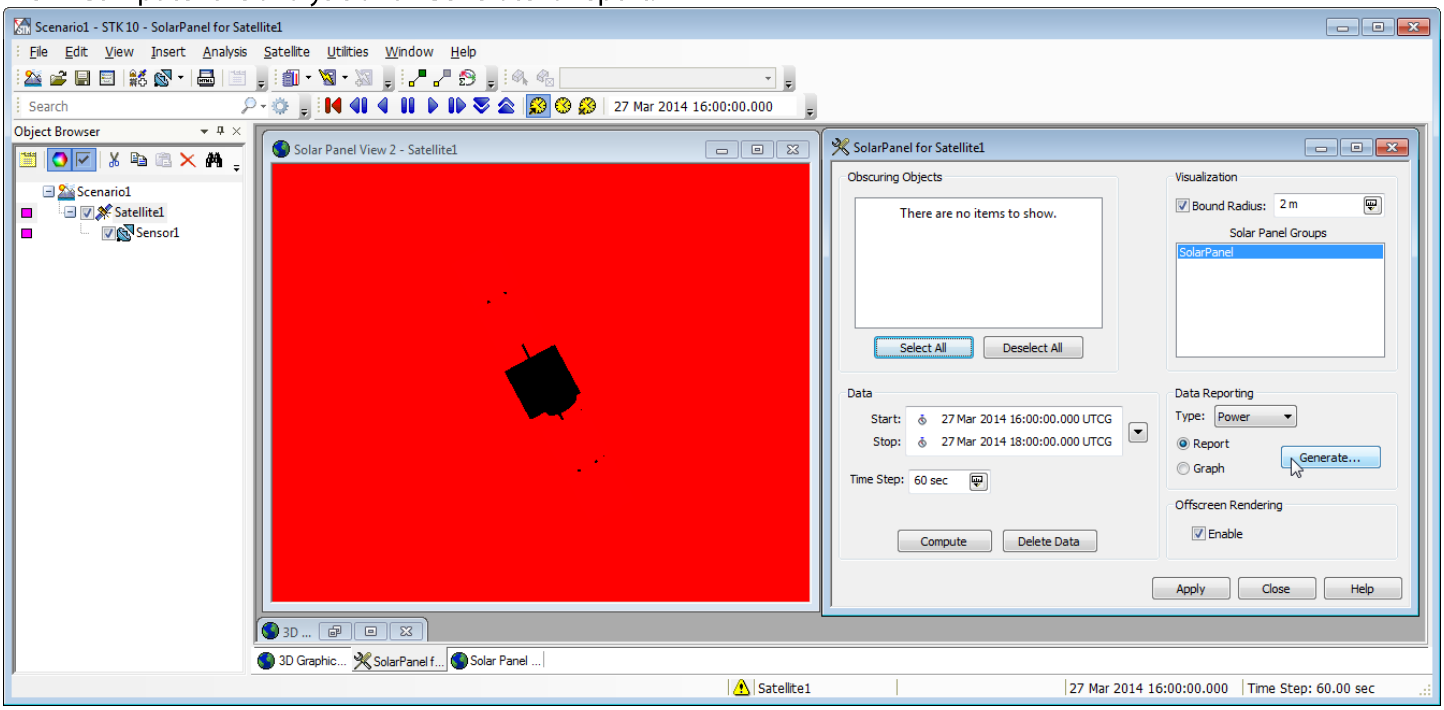
This brings up the "Solar Panel Tool", which has two windows for your settings and a view of the satellite.



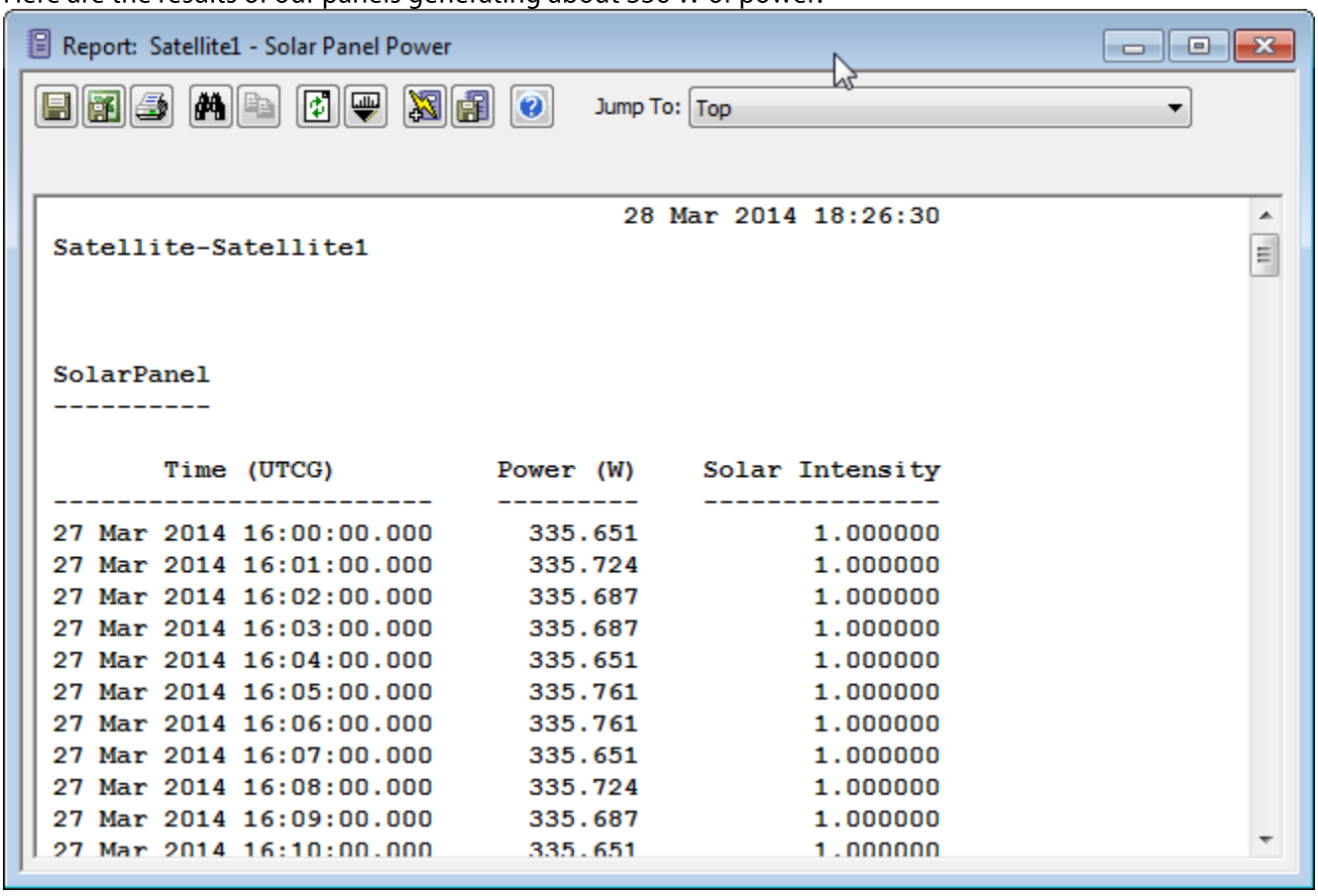
Check and set the "Bound Radius" to about 2m and hit "Apply". This will fill the view window with the satellite. The background of this window is red and when you select your "Solar Panel Groups", they will also turn read. This does make it difficult to see the panels, but rest assured that only the panels will be evaluated.

Setting the correct “Bound Radius” is important. The closer you are, the better, as this is a pixel counting tool. Being the case, within reason, the larger the view window is, the more accurate the results will be.
You may notice that “Offscreen Rendering” is checked, meaning that it’s ok to cover the view window.

For a test, I’ve set the “Data” for a 2hr period with a 60 sec time step.
We’ll “Compute” the analysis and “Generate” a report.



Here are the results of our panels generating about 336 W of power.



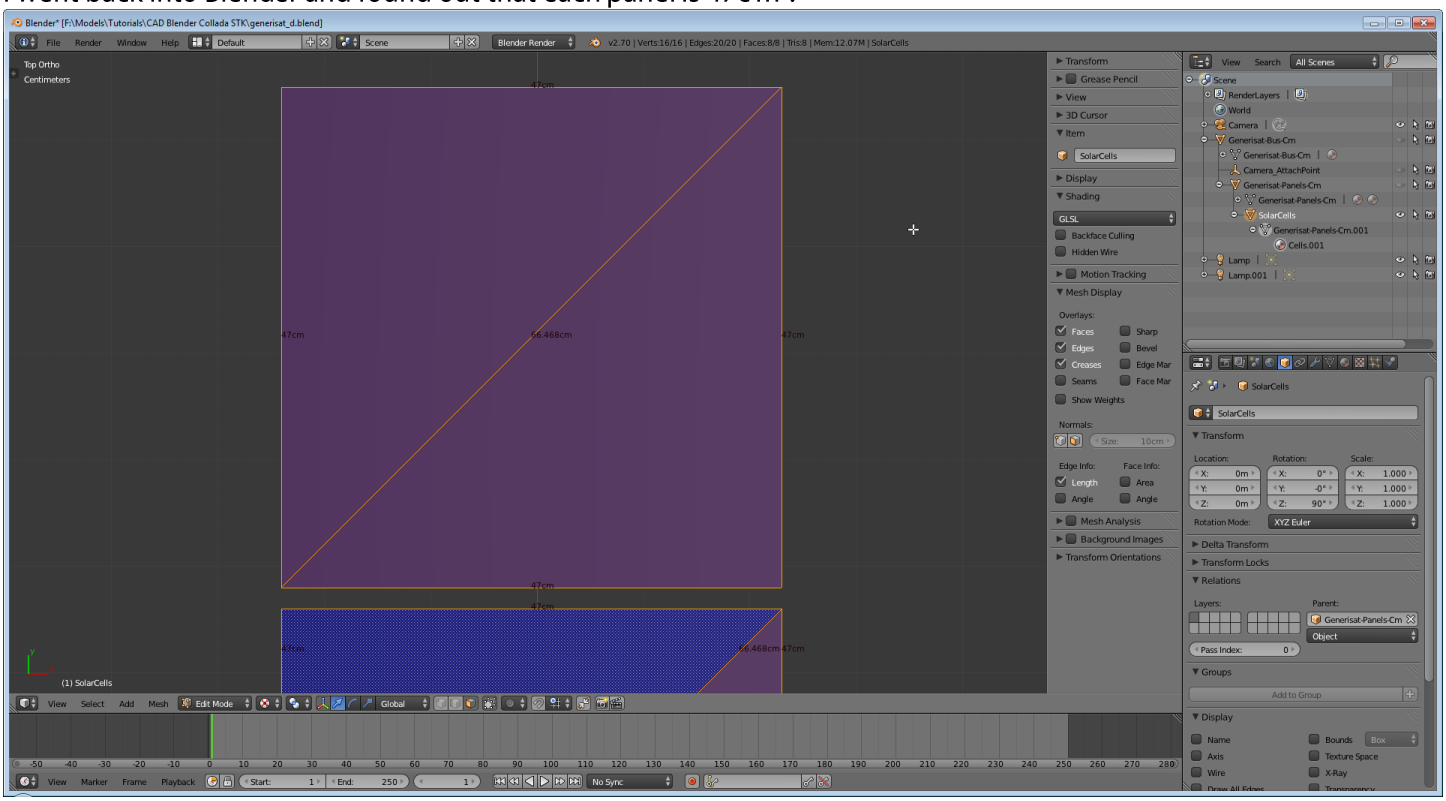
So, how did we arrive at these values?
The formula the STK uses to calculate power is Power = SPE x SI x Effective Area x 1358 W/m^2

SPE is the solar panel efficiency. A solar panel does not convert 100% of the sun's energy and degrades over time. SPE ranges from 0.0, a piece of cardboard, to 1.0, a perfectly efficient solar panel.

SI is the solar intensity. The value is 0.0 if the solar panel is in the earth's umbra, 1.0 in full sunlight, and between 0.0 and 1.0 in penumbra.

The value 1358 W/m^2 is the amount of W/m^2 that the sun provides near the earth.
More on this topic can be found here http://blogs.agi.com/insight3d/index.php/2008/04/16/shedding-light-on-the-solar-panel-tool/

Let's test this to make certain that it is correct.
The SPE we used in our ancillary file was 28%.
The SI from our report is 1 in full daylight and would be 0 at night.
Our Effective Area is the physical size of the panels.
The effective area is the area of the solar panel as seen from the sun. The area is used to determine how much of the sun's energy is reaching the solar panel.
Now, this value won't change over time, because our panels are tracking the sun and therefore always pointing directly at it.
I went back into Blender and found out that each panel is 47c m^2.



Because we have four panels, they add up to 0.8836m^2.
Inserting that into our equation gives us this 0.28x1x0.8836x1358 W/m^2=335.98 W/m^2
This is very close to the results that were reported out of STK's Solar Panel Tool.

Well, that about does it for this tutorial on converting CAD files through Blender and setting it up for use in STK.

If you have any comments or questions, send them to support@agi.com

Randall Scott Cross
Model Development Specialist
Analytical Graphics, Inc.
220 Valley Creek Blvd.
Exton, PA 19341-2380
www.agi.com
v. (610)981-8000
f. (610)981-8001