

Blender

[Blender](#) is an open source 3D content creation suite that has built-in support for the file format COLLADA that the HPL2 engine uses to import 3D models.

Program setup

The great thing with Blender is that it will work great without any additional settings, plugins or similar.

How To - The very basics

1. When Blender launches it creates a default cube.
2. Select it and press F6 (Texture menu for materials).
3. Set Texture Type to Image and in the image menu click load, go to your HPL2 directory and select a texture (the .dds/.jpg/.tag not the .mat file).
4. Press TAB to go to edit mode and make sure the whole object is selected, Press U and select UV - Unwrap.
5. Go to the UV/Image editor view mode and in the pop-up menu right of the UVs menu, select the previously loaded texture.
6. Go back to 3D view, change draw mode to Textured and you can see that the model is textured and ready.
7. With the object selected go to File → Export - Collada 1.4 and set the following options to On(all others off) (for Blender 2.49):
 - Triangles
 - Only Export Selection
 - Disable Physics
 - Only Current Scene
 - Use Relative Paths
8. Click the “...” button by the Export file: path and export the file as “filename.dae” in a folder within HPL2, for testing do it in the same folder as the texture you selected.
9. Using the [model viewer](#) you can now view the newly created good looking cube.

Additional Information about the above steps

The above is intended to quickly give a sure step by step process to successfully export an object and get it to display within the game engine. It is not intended to be a lengthy tutorial or be suited for advanced, novice or any specific type of user. It's just a quick how to. In step 6 you do not actually have to select the object or make those specific export settings, but they create the smallest file with the least amount of waste information as well as minimize the risk of that something is not working due to an export problem.

Important information about images

Textures(.dds, .jpg, .tga and other general image formats) and material files(.mat, the engine specific file) - The material file is an XML file that contains information on what images is used in the material to create the texture and other properties. When the game loads a 3D model it searches for the name of the image used as a texture for the model(the .dds etc) and then replaces that name with the extension .mat to know what material file to use. Because of this it is important that the texture used has the same name as the material file, with the exception of the extension.

Blender to .Dae, How to get it right

This article is written by [yasar11732](#) in forums.

First thing to take into account when exporting is, **correctly U.V mapping the texture**. Adding texture to material without U.V mapping will NOT work, it will just crash your model viewer.

Also, before adding texture to your models, save your model (blender file) to where textures resides (It is better if they all in one place). And use relative paths when adding textures instead of absolute paths, then export your .dae to same place. So that you can easily port your models from one place to another (Always keep .dae and textures together). If you don't do that, you will probably fail at some point when trying to move your models from one place to another.

Sometimes, when you export your model and see it in model viewer, you may see that texture is shown "inside" of the model, not the outside (look really weird)! That is because normals of your model is mixed up. You should go back to blender and flip the normals and export again. This should solve your problem.

And also, one should remember always to use textures with sizes of exponents of 2 (256×256, 512×512, 128×1024 etc.) Otherwise it will either crash your model viewer, or just look awkward.

One last point to remember is that, materials you create might be mixed up when you create it and one material accidentally get linked against 2 different geometries (or datablocks, or vertex groups... whatever one calls it). If this happens without proper care (accidentally), your model will not export correctly. Will look funny in model view.

You can export rotation/location/scale animations without problem, however, I couldn't export shape animations correctly into .dae format (because blender's collada export sucks! Not actively developed.) If anyone can do it and post the method, it would be great to learn about it.

And also, do not try to export armature/bones, because as far as I can say, that doesn't work either. If someone could do it, I also want to hear about how to do it.

Notes on Blender 2.5

When exporting to .dae, there will be a few issues you may notice upon importing your work into the Model Editor of Amnesia. These issues include your model appearing to be missing "pieces," being

incomplete and full of holes. The COLLADA plug-in of Blender 2.5 automatically exports everything as triangles. When it comes to converting quads (i.e. faces with four sides) it does a bad job at it. In Blender 2.5, when you convert a quad into a triangle (Ctrl + T) you'll notice the quad becomes two triangles forming a shape of a quad. However, the COLLADA plug-in when it faces a quad and converts it to a triangle it'll only provide one triangle in the place of the quad (or screw up the vertices of the second triangle). To avoid this issue, you must convert all your quads to triangles in Blender before exporting to .dae format.

Another issue you might run across is when you import your mesh into the Model Editor of Amnesia you notice that your object(s) are rotated (though in Blender it looks perfectly fine). This is caused by the local axis of the object(s) conflicting with the global axis of the environment. To fix this, in Blender, in Object mode, select the object you want to correct and in the menu on the bottom of the 3D View go to Object > Apply > Rotation. This will align the local axis of the object to the global axis of the object. Now when you import your mesh again in the Model Editor everything will be on their proper axis. Do the same for location and scale if need be.

Exporting of object names is also not currently supported in Blender 2.5. This may cause some issues when creating your own entities from your custom models. When attaching a body to your model in the Model Editor of Amnesia and the object doesn't have a name, the game will not be able to properly attach the model to the body. This may result in half of the model located at the exact center of the map, while another part having been properly attached to the model's body. The current workaround is to edit the .dae file and manually add names to your meshes. Under `<library_visual_scenes>`, each mesh from your model will be represented by a node tag. Simply edit the node tag by adding a name attribute with the value being the object's desired name.

For example:

```
<node id="Cube" type="NODE" name="Cube">
```

There are also issues with armatures, but full armature export support is scheduled for Blender 2.6.

Blender 2.59.4

Blender 2.6 is capable of exporting armatures to where the ModelView and Model Editor doesn't crash; however, this only after one minor tweak to the exported .dae file. It would appear that the HPL engine is not capable of finding the bones of the controller (armature) when they are parented to an armature node in the .dae file. Current workaround is to either remove the armature parent node or change its type from "NODE" to "JOINT". Do note, though, that this only prevents the ModelView from erroring out with "Bone '%s' does not exist;" there is no guarantee that your model will look and work exactly as it does in Blender.

Blender Tutorials

Does Blender seem like a good program for you? But you have not modeled much before or not modeled much with Blender? Here are some useful sites that have tutorials and information to help you out.

- [Wikibooks should keep you occupied](#)

- [Tutorials listed on the Blender site](#)
- [Video tutorials 1](#)
- [Video tutorials 2](#)

From:
<https://wiki.frictionalgames.com/> - **Frictional Game Wiki**

Permanent link:
https://wiki.frictionalgames.com/hpl2/third_party_tools/3d/blender?rev=1318287943

Last update: **2011/10/11 00:05**

