

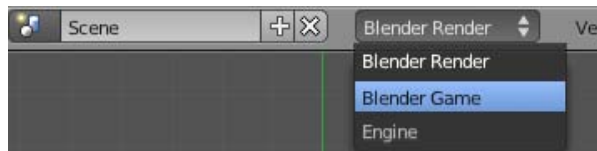
Chapter 22- Textures in the Game Engine

GLSL Shading

GLSL shading is very new to Blender and is an area seeing a great deal of development. It is an attempt to add many rendering-like features to the game engine, adding to a more realistic environment. As with the UV Mapping section, this unit will only cover the basics to get you started.

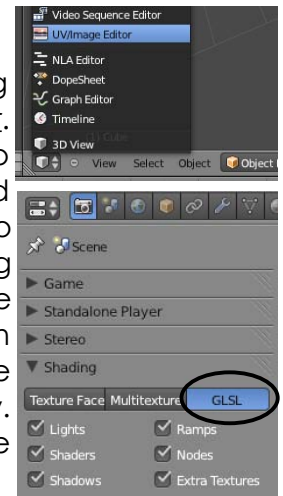
Not all video cards are supported for GLSL shading. Some machines will be unable to use this feature. See the Blender wiki for current specification.

To get started, we'll start a new scene and split the viewport, setting one to UV/Image Editor.



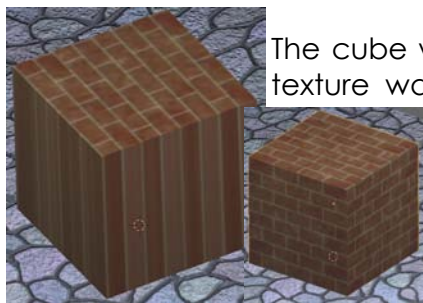
Set the Engine to "Blender Game" and viewport shading to "Textured". The last thing

you need to do is switch from "Multitexture" to "GLSL" shading in the Render panel. You are now ready to work with the GLSL features in the game engine. Not all texture and shading features are available in GLSL, but many are and others are being developed constantly. As mentioned before, we will only be looking at some of these features.

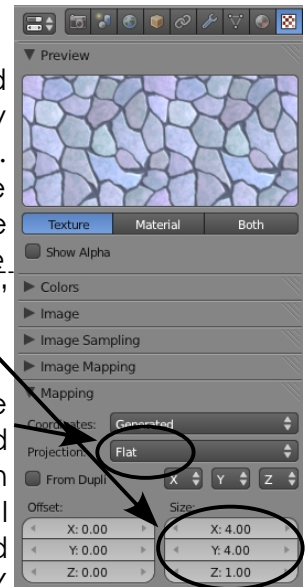


Adding Textures:

Basically, to add textures for the game in GLSL, you need to add materials and textures the way you do for any movie. For my example, I am working with a plane for the ground and a cube. I've added a material and texture to the floor, using a stone texture. The stones are too large so I need to repeat it a bit in the texture panel. You can't use the X and Y Repeat in the "Image Sampling" panel, but can change the size in the "Mapping" panel.



The cube was a bit more difficult. Because the texture wants to map as Flat by default and GLSL does not currently work with changing the mapping to Cube, I had to apply the texture as we did in the previous section using UV Texture Mapping and mapping the texture to each face. Remember to also switch the Mapping in the Textures panel to UV.



Another nice feature with GLSL is the ability to show Normal Geometry to give a texture depth. Looks best with a higher Specular.

