Since games need to be able to process operations as fast as possible, traditional rendering techniques (specular, ray tracing reflections and refractions) cannot typically be processed fast enough for a game. For this reason, textures need to be mapped differently. There are also times when you may want to use mapped textures in an actual render. Blender does this through traditional UV Texture Mapping and a new system called GLSL Shading. There is a lot that can be done through both of these methods “beyond the basics” that will be discussed here. For more details, check out the Blender wiki.

UV Texture Mapping

Think of UV Mapping like taking a box and cutting it to lay flat. The texture needs to match each side of the box. It is difficult to do that with traditional materials and textures. Let's say you want to model a dinosaur. The texture changes on various parts of his body and need to be mapped correctly. This is where UV Mapping come into play. For this example, I am going to map the following texture I made in GIMP on a basic cube:

This is just a jpeg image that would be impossible to map as a standard material/texture.

To begin, start with a basic scene with a cube and change your viewport shading type from Solid to Textured. This is the shading used during game play. You will notice that, by default, textures are effected by the lighting so add some lights to illuminate your scene better and switch to the “UV Editing” window layout. This will give you one 3D viewport and one UV Editor viewport. At the bottom of the UV Editor viewport, hit the Image-Open Image menu option and find the texture you wish to use.

Now, enter Edit Mode for the cube and switch to selecting Vertices to Faces since this is a face applying process. You can select individual faces and put UV textures on that way, but let's “Unwrap” the cube to match our texture. This can be done for any mesh, but we need to mark the seams where we want a split to occur. If we look at the picture, we can see where seams should go. In order to mark seams, we need to switch from Face select to Edge select.

Select the following edges (Shift-RMB) and click “Mark Seam” in the Shading/UVs tab section in the Tool Shelf. These will be the unfold edges. It should match the box layout.
Chapter 23- UV Texture Mapping

Now go back to Face select mode, hit “A” for All twice to select all faces. Type “U” to bring up the UV Mapping options in the 3D window. You have several option. We want “Unwrap”. You will now see the unwrapped faces in the UV Mapping window. You can select these vertices as you would for any other Blender object and move, scale or rotate them. You will also see the texture on the cube. By pressing “P” you will see the texture in game play. (in object mode) Adjust the vertices so it looks good on the cube.

UV Textures in an Animated Movie:
Just like game physics can be written into an animation curve, UV can be used with materials and textures. After going through the steps above, add a material and texture to the object in the Internal Renderer.

Select “Image or Movie” for the texture type.

Select the picture you used in the Image panel (it will be listed un the drop-down. You will not need to go out and open it from your folder again).

Under the “Mapping” panel, choose “UV” in the Coordinate box. Pressing F12 should give you a rendered image of the map.

UV in Cycles:
This is the same as applying an image texture as discussed before with only a few adjustments. After creating the unwrapped UV map, go to Cycles Materials panel, use a Diffuse node, set Color to Image Texture. Select the image from the drop down option, and set Vector to UV.
GLSL Shading

GLSL shading is relatively 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 in the internal render engine. 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.
Shadows in GLSL:
Another nice effect allowed in GLSL is the ability to cast shadows in game play. Right now, ray tracing is not supported so your only option is to use a Spotlight with a Buffer Shadow setting (refer to the lighting chapter for more details on setting the buffer shadow).

Currently, there is a lot of development in GLSL with new features constantly being added. There is also a lot of work in Baking settings to improve performance. There is work on support for indirect lighting, fluids, soft bodies and many more. For up-to-date information, follow the Blender wiki, YouTube, and the forums.

World Settings:
While some World settings work in Multi-texture mode and some work on GLSL Shading, some features do not work in either at the time (stars for example). In Multi-texture shading, you can get a nice effect with the Mist settings to give a "foggy" feel to your game, but works a bit differently in GLSL. World Horizon and Zenith colors work differently as well.
Scenario:
You work for an advertising firm and have been asked to create a commercial featuring your client’s new cereal brand. Create a table scene featuring the product.

Using the UV texture mapping features, create your scene. Search the internet for an unwrapped box pattern of your favorite cereal to use or scan your own box. Here is the texture used in this example:

Create your box from a cube, apply the UV texture maps, and add other elements to your scene.

Rather than trying to unwrap the entire box, try selecting a single face to unwrap using the “U” command in edit mode. You will have a single face with corner vertices in the image mapping window to work with. Move those points around to match the side you want to use. You may also need to rotate the face. Continue until finished.

Add other elements to your scene to make it look nice and realistic. Add appropriate lighting and set a good camera angle. Render and save an image of your work.

** Call the instructor when finished **
Challenge Task & Chapter 23 Reflection

For your challenge task, open your maze game file. Add appropriate UV textures to the elements in your game. Add a world that uses mist in your maze for a dreary effect. Feel free to add and texture any other elements you wish.

** Call the instructor when finished **

Chapter 23 Reflection and Wrap-up:

UV Texture Mapping

UV maps are used in most professional movies and advertisements. Please take a few moments to reflect on these questions.

1. Take a look at a 3D animated animal (bird, dinosaur, etc.). Look at the variations in the texture of the animal. Why would it be difficult to get a texture to map correctly using the traditional image texturing in Cycles and the internal render engines discussed earlier in the book? Why would UV mapping be better? Explain.

2. Research the internet for how you would unwrap an animal, like a dinosaur, and UV texture it in Blender. What did you find? Explain the process.