JOGL (Java OpenGL) - Textures

`@for` Developers
`@author` Kai Ruhl
`@since` 2009-04

Textures the Classic Way

This article is an outsourced excerpt of my JOGL tutorial; it is concerned with textures. Instead of using the `TextureRenderer`, you can also use textures the classic way (i.e. as it is done in pure OpenGL). Doing so is split into the `init()` and the `display()` stage.

You need the following tools at your disposal:

- An `int` in an `int[]` (array) to store the texture id.
- A PNG image containing the texture.
- A `GL` object to operate on.

With that, go to the `init` stage:

1. have an `int[]` array for texture identifiers (1 `int` per texture)
2. load image data into a `byteArray`, but do not do anything with it yet
3. announce to OpenGL (`GL`) how many textures you want ever in this app:
   ```
   glGenTextures(<textureCount>, <fittinglySizedIntArray>);
   ```
4. per tex, announce to OpenGL what type it is:
   ```
   glBindTexture(GL_TEXTURE_2D, textureArray[<position>]);
   ```
5. per tex, load the image `byteArray` into graphics card (where `texId == textureArray[<position>]`):
   ```
   gl.glTexImage2D(<texId>, <always-0>,
                   <dataToUse: RGB, wid, hei, border>,
                   <dataSrc: RGB, UNSIGNED_BYTE, byteArray);
   ```
6. globally tell OpenGL what to do when zoom in (`MAG`) or zoom out (`MIN`):
   ```
   glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_LINEAR);
   glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_NEAREST);
   ```
7. globally tell OpenGL to enable 2D textures:
   ```
   glEnable(GL_TEXTURE_2D);
   ```

And in the `display` stage:

1. announce (before `glBegin()`) which texture you want to use:
   ```
   glBindTexture(GL_TEXTURE_2D, textureArray[<position>]);
   ```
2. draw only quads:
   ```
   glBegin(GL_QUADS);
   ```
3. for each quad, announce texture corner position before quad corner position:
   ```
   glTexCoord2f(0.0f, 0.0f);
   glVertex3f(-1.0f, -1.0f,  1.0f);
   ```

The above is certainly not a complete, tutorial-like description of how to use textures the native way. For details, may I refer you to the NeHe tutorial, which explains it much better. I hope this small excerpt persuaded you to use the JOGL `TextureRenderer`, as described in my JOGL tutorial. And with this, have a good time!

EOF (Apr:2009)