

Multitexturing

Un dels plugins que et proporcionem és **multitex**. El que fa aquest plugin és, bàsicament:

- Al mètode **onPluginLoad**, crea un VS i un FS senzills per usar textures. El VS calcula de forma automàtica coordenades de textura (s,t) a partir de les coordenades dels vèrtexs i del radi d'una esfera contenidora del model (el radi el rep amb un uniform). El FS simplement assigna al color del fragment el color mig de dues textures (sampler0 i sampler1). El mètode també té el codi necessari per carregar dues imatges (obrirà un diàleg per que trieu les imatges) i definir un parell de textures. La classe QImage de Qt simplifica molt aquesta tasca. Observa que el codi està carregant la primera textura a la texture unit 0, i la segona a la texture unit 1 (crides **glActiveTexture**). Els identificadors OpenGL de les textures (textureId0,textureId1) són atributs de la classe, ja que més endavant seran necessaris per activar aquestes textures. La funció més rellevant és **glTexImage2D**, la qual defineix una textura a partir d'un buffer que conté les dades de la imatge que hem carregat:

```
glTexImage2D( GL_TEXTURE_2D, 0, GL_RGB, image.width(), image.height(), 0,  
              GL_RGBA, GL_UNSIGNED_BYTE, image.bits());
```

- Al mètode **preFrame**, s'activen el VS i el FS, i es defineixen els uniforms que contenen els texture units de cada sampler:

```
program->setUniformValue("sampler0", 0); // texture unit del primer sampler  
program->setUniformValue("sampler1", 1); // texture unit del segon sampler
```

També s'activen les textures corresponents:

```
glActiveTexture(GL_TEXTURE0); // texture unit 0  
glBindTexture(GL_TEXTURE_2D, textureId0);  
glActiveTexture(GL_TEXTURE1); // texture unit 1  
glBindTexture(GL_TEXTURE_2D, textureId1);
```

En aquest exercici només et demanem que provis aquest plugin amb el viewer. Hauries d'obtenir imatges similars a aquestes (depenent de les textures que carreguis):

