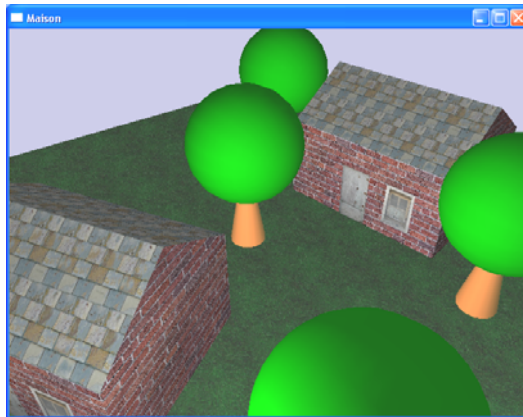


IN – Synthèse d'images - Rendu TP 2

Extrait des TP de Sébastien Thon, Université Aix Marseille

Nous allons maintenant appliquer différentes textures sur les surfaces de notre scène. Vous utiliserez comme textures les images **mur.tga**, **facade.tga**, **toit.tga**, **herbe.tga** se trouvant sur le site. Sur la même page, vous récupérerez les fichiers **tga.h** et **tga.cpp** qui permettent de lire des images au format TGA.



1. Chargement des textures

Déclarez en global autant de constantes que de textures :

```
#define TEXT_MUR      0
#define TEXT_FACADE   1
#define TEXT_TOIT     2
#define TEXT_HERBE    3
```

Déclarez un tableau d'indices de textures :

```
GLuint      textures[4];           // pour nos 4 textures
```

Demandez à OpenGL de remplir ce tableau :

```
glGenTextures(4, textures);
```

Chargez chaque image au moyen de la fonction **load_tga()**.

```
byte *img;
int  width, height;
img = load_tga("mur.tga", &width, &height);
if( img != NULL )
{
    glBindTexture(GL_TEXTURE_2D, textures[TEXT_MUR]);
    glTexImage2D(GL_TEXTURE_2D, 0, 3, width, height,
                 0, GL_RGB, GL_UNSIGNED_BYTE, img);
    delete[] img;
}
```

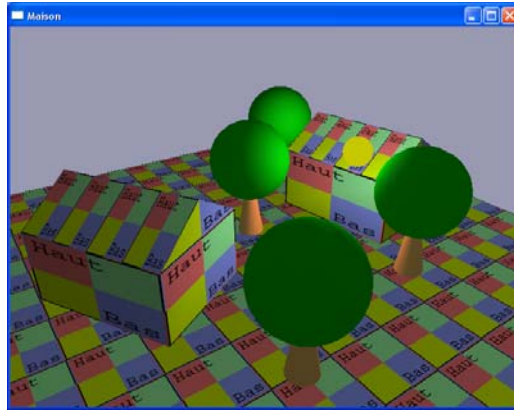
2. Plaquage des textures

Activez le plaquage de texture :

```
glEnable(GL_TEXTURE_2D);
```

Appliquez les textures sur les murs, le toit et le sol en spécifiant les coordonnées de texture avec **glTexCoord2f()** avant chaque déclaration de vertex faite avec **glVertex3d()**.

Pour vérifier si vous spécifiez correctement les coordonnées de texture, vous pouvez utiliser comme texture de test l'image **test.tga**



3. Mode de filtrage

Essayez plusieurs modes de filtrage pour les textures :

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, mode);
```

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, mode);
```

avec **mode** = **GL_NEAREST** (pas de lissage de texture), puis avec **mode** = **GL_LINEAR** (lissage de texture).

4. Modes de bouclage

Testez plusieurs modes de bouclage pour les textures :

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, mode );
```

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, mode );
```

avec **mode** = **GL_REPEAT**, puis avec **mode** = **GL_CLAMP**.

5. Mode de mélange

Testez plusieurs modes de mélange des textures avec la couleur :

```
glTexEnvf( GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, mode );
```

avec **mode** = **GL_DECAL**, puis avec **mode** = **GL_MODULATE**.

6. Mip-mapping

Des effets d'aliassage apparaissent sur les textures de mur et de toit. On va utiliser la technique de mip-mapping pour filtrer les textures et réduire ce problème d'aliassage. Utilisez la fonction suivante au lieu de **glTexImage2D()** lors du chargement de la texture à filtrer :

```
gluBuild2DMipmaps(GL_TEXTURE_2D,3,width,height,  
GL_RGB,GL_UNSIGNED_BYTE,img);
```

Puis indiquez :

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_LINEAR_MIPMAP_LINEAR);
```

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_LINEAR_MIPMAP_LINEAR);
```