

TP Ogre 3D



Démarrage du TP

Récupérer sur Moodle les fichiers :

OgreQT5-master.zip

kermit.png

Dans la VM : remplacez les fichiers du répertoire **Ogre_1_9/OgreQT5-master** par ceux fournis dans le fichier .zip

Ajouter le fichier **kermit.png** dans le répertoire : **Ogre_1_9/media/materials/textures**

Activez QtCreator en ouvrant le fichier **OgreQT5.pro** dans le répertoire **Ogre_1_9/OgreQT5-master**

Checkpoint #1

Vérifions ensemble le visuel obtenu et passons en revue les commandes

La souris fait bouger la caméra, le clavier manipule la scène et les objets. Nous allons voir qu'agir sur la caméra ou sur les objets n'est pas un choix anodin !

Avec la souris : regardons ensemble les effets, c'est très intuitif.

Avec le clavier :

P – mode rendu plein / mode wireframe (maillage visible)

D – visualisation du repère local des SceneNode

B – visualisation des BoundingBoxes des SceneNode

F et G : rotation roll() – Axe Oz

H et J : rotation yaw() – Axe Oy

K et L : rotation pitch() – Axe Ox

Utilisez la documentation Ogre : <https://www.ogre3d.org/docs/api/1.9/> pour trouver des informations

Revue de code

La majorité de vos interventions se fera dans le fichier **QTOgreWindow.cpp**, et plus précisément dans la méthode : **void QTOgreWindow::createScene()**

L'objet central est **OgreSceneManager**.

Faisons les commentaires en direct dans le fichier pour comprendre ce qu'il se passe

A vous de jouer !

Etape 1 : ajoutez un Ninja dans la scène, au-dessus de Tux le pingouin (choisir une coordonnée y autour de 100 pour `position()`), appliquer la même mise à l'échelle `scale()` que pour Tux – Dans le graphe de scène, le SceneNode associé doit être un nœud fils du SceneNode associé à Tux le pingouin.

Les fichiers de maillage sont dans le répertoire `Ogre_1_9/media/models`

Les fichiers de texture sont dans le répertoire `Ogre_1_9/media/materials/textures`

Vous devriez constater que le Ninja disparaît en tout ou partie quand vous interagissez avec la scène.

Point de validation n°1 – 5 Points

Pourquoi le Ninja disparaît-il ? Corrigez ce problème, dont l'origine se trouve dans la méthode : `void QTOgreWindow::initialize()`, au niveau des initialisations de la caméra.

Etape 2 : ajoutez un deuxième Ninja dans la scène, 2 fois plus petit que le premier, à gauche de Tux le pingouin. Le SceneNode associé à ce deuxième Ninja doit être un fils du nœud de scène root.

Point de validation n°2 – 5 Points

Montrer la scène, interagir avec la souris, expliquer le visuel obtenu avec les interactions de rotation F,G,H,J,K,L

Etape 3 : Changeons de cible

Dans la méthode `void QTOgreWindow::createScene()`, nous pouvons choisir quel nœud de scène sera affecté par les rotations effectuées avec les touches F,G,H,J,K,L : utilisez

```
cameraManager->setTarget(nœud_de_scene_cible);
```

et constatez les différences de comportement

Etape 4 : Maillages, textures, coordonnées de texture

Il est temps maintenant d'utiliser nos propres maillages : plutôt que d'utiliser un fichier de maillage pour les objets, nous allons utiliser un maillage 'manuel', ici un 'Terrain' avec 16 points.

Regardez le code fourni dans le fichier `Terrain.cpp` :

La méthode `createKube()` vous permettra, d'ici quelques minutes, de tester le maillage que vous avez préparé en TD.

Pour le moment :

Dans la méthode `void QTOgreWindow::createScene()`, créer le maillage de terrain en appelant correctement la méthode `createTerrain()`, qui est une méthode statique. Le fichier de données à utiliser en premier paramètre se nomme « `default.dat` ». Pour les curieux, ce fichier se trouve dans le répertoire `Ogre_1_9/build-OgreQT5-Unnamed-Debug`

Une fois ce maillage créé, vous pouvez créer une `Ogre::entity` à partir de ce maillage en fournissant le paramètre « `terrain` » au lieu d'un nom de fichier

Ajouter un matériau avec juste une texture (comme pour Tux et les Ninjas), en utilisant par exemple le fichier « **kermit.jpg** ».

Point de validation n°3 – 5 Points

Nous avons 9 kermit, un sur chaque quad créé pour le terrain – A vous de faire en sorte que la texture soit appliquée globalement une fois sur le maillage.