

L'Ingenieur Costumier Modelisation Project

Contents

1. Structuration du projet	1
2. Modele 3D	1
2.1. Les bases d'un objet 3D	1
2.2. Le lien de parenté	2
2.3. Le mesh	2
2.4. La structures des os	2
3. Le moteur Unity avec le modele 3D	3
3.1. importation du fichier excel	3
3.2. Gestion de la caméra	3
3.3. Transformation de l'avatar	3
3.3.1. Structure de données des Os	3
3.3.2. Modification de l'avatar	4
3.4. Autre objet dans Unity.	4
3.4.1. Le sol	4
4. Ressources liens et logiciel	4

Les objectifs sont de modéliser un avatar definis sur mesure afin de pouvoir visualiser comment irait un costume sur la personne concerné. Ici nous nous focaliserons sur la modelisation et création du logiciel qui permettra la visualisation du costume sur la personne. Celui ci sera réalisé avec le logiciel Unity. Il y a aussi l'utilisation du logiciel Blender pour la modelisation 3D.

1. Structuration du projet

Le projet est composé du projet Unity dans le dossier LIngenieurCostumierUnity

Des modèles 3D et export dans le dossier Blender.

Et d'information sur le projet dans le dossier Documents.

Et tout ce projet est hébergé sur [git-hub](#)

2. Modele 3D

Dans cette partie nous expliquerons comment est formé le modèle et afin de pouvoir l'utiliser correctement. Il est principalement composé de 2 objets : Le mesh et le squelette.

Le mesh permet l'affichage de l'avatar, et le squelette nous permet de faire du mouvement et des modifications sur le mesh

2.1. Les bases d'un objet 3D

Un objet 3D en informatique qu'elle qu'elle soit possède 3 interactions fondamentales :

- Sa position
- Sa rotation
- Sa taille (scale en anglais)

C'est trois choses permettent d'interagir (le déplacer, le tourner et l'agrandir ou le retrecir)

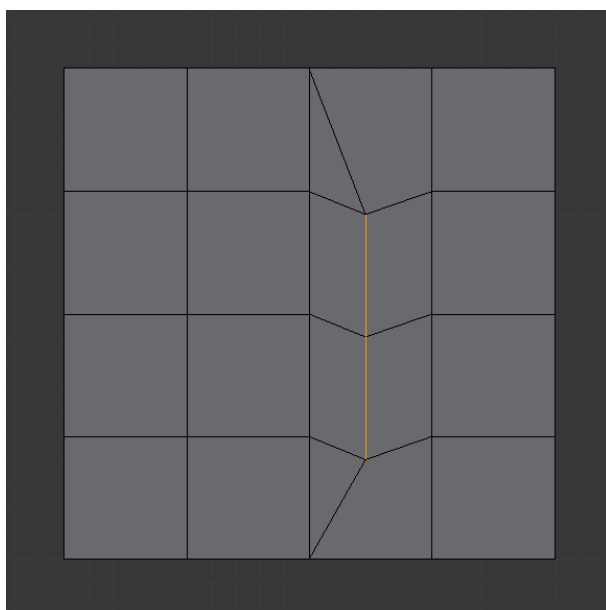
2.2. Le lien de parenté

Il est possible de faire un lien de parenté (pere enfant) entre les objets 3D. Ce lien de parenté fait en sorte que si on applique une transformation fondamentals au pere alors l'enfant subis la meme transformation

2.3. Le mesh

Le mesh est la structure d'un modèle 3D

- Un mesh en informatique est composé de plusieurs structures la première est le vertex et correspond à un point dans l'espace.
- La seconde est l'arete reliant 2 vertex dans le but de faire un "trait" ou une "droite".
- Enfin la dernière structure est la face celle-ci relie plusieurs arêtes entre elles afin de former une surface qui est une face (il en faut au minimum trois pour former un triangle).



sur cette image nous avons un petit exemple d'une structure avec au niveau des intersections des vertex qui sont reliés par des arêtes formant des carrés (et 2 triangles).

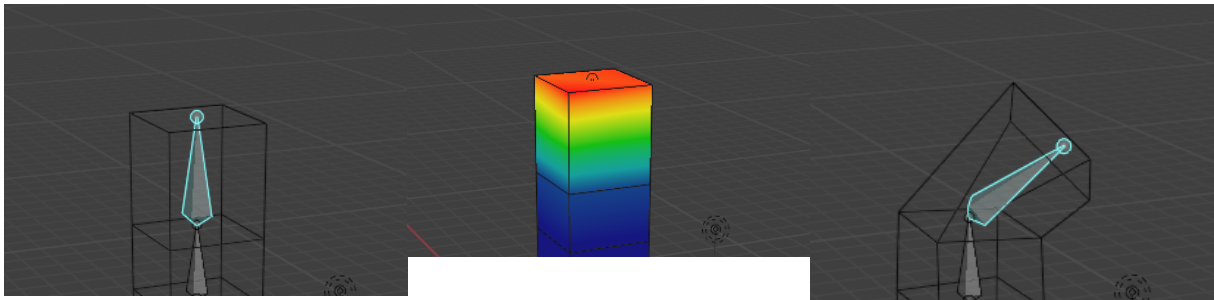
2.4. La structures des os

Un os est une autre structure permettant de modifier un mesh 3D plus facilement et répondant a des contraintes d'animations. Prenons un exemple : Imaginons nous voulons modifier un mesh 3D assez complexe (ce que nous voulons faire) alors pour cela il faut pour chaque vertex enregistrer les nouvelles position qu'il aura et cela va vite devenir complexe si notre modèle possède beaucoup de vertex. C'est pour cela que les os interviennent.

La representation d'un os est abstraite mais nous la representons principalement par des batons que nous plaçons sur le mesh et possède comme tout objet 3D une position, une rotation, et une taille. Pour chaque vertex et chaque os nous donnons une fonction (appellé fonction de poids) qui donne une valeur sur $[0, 1]$. Cette valeur representas l'influence de los sur ce vertex :

- Si la valeur est a 1 alors dès que l'os subira une transformation (déplacement, rotation, scaling) notre vertex fera la meme transformation
- Si la valeur est a 0 alors quelle que soit les transformations subis par l'os celui ci ne subira pas ces transformations

- Et si la valeur est entre 0 et 1 le vertex subira partiellement les transformations selon la valeur.



Sur les 3 images au dessus nous pouvons voir un mesh 3D avec un squelette (groupement d'os). Sur la seconde image nous pouvons voir la fonction de poids de l'os au sommet avec en rouge la valeur 1 et en bleu la valeur 0 (en vers et jaune nous avons les valeur entre 0 et 1). Et enfin sur la 3eme image nous avons le resultat de la roation de l'os sur le modèle.

Dans le but de pouvoir creer un modele pouvant prendre des proportions différentes nous allons utiliser cette structure de squelette car celle ci en changant la taille des os nous permet de modéliser les modifications que l'on peut faire sur notre avatar.

3. Le moteur Unity avec le modele 3D

Dans un premier temps nous importons notre modele 3D dans le moteur unity sous le format fbx qui nous permet d'importer les os et certaine animation et position que nous pouvons définir a l'avance sur Blender.

3.1. importation du fichier excel

Pour importer le fichier Excel nous utiliserons un package qui nous permettra de lire un fichier excel.

Le package que nous utiliserons est ExcelDataReader, de plus nous rajoutons le package systemencodingpage pour l'accompagner. Nous pouvons retrouver les .dll des packages dans Assets/PLugins

Ces packages nous permet d'importer le fichier excel et de le transformer en structure de données que l'on peut utiliser facilement dans le code

3.2. Gestion de la caméra

La gestion de la camera fonctionne da la manière suivante :

- La molette permet de zoomer et dezoomer
- Le clique gauche puis glisser de la souris permet de tourner autour du modèle

Le code de la camera est fait pour cibler un objet et tourner autour et l'objet choisit est un os au centre du corps de notre avatar comme cela si celui-ci est plus grand ou plus petit nous gardons toujours le centre de notre personnage au centre de l'écran

3.3. Transformation de l'avatar

3.3.1. Structure de données des Os

Avant de modifier les os nous les placons dans une structure de données ou ils sont assignés a un nom dans le but de faciliter l'accès lors des appels.

3.3.2. Modification de l'avatar

A la lecture du fichier excel nous recuperons les données et nous modifions la position des os afin que notre avatar puisse avoir les bonnes proportions. Le choix de modifier les positions et pas la taille des os est induit par l'effet de parenté du squelette. En effet dans un squelette tout les os sont parents dans le but d'avoir de bonne interactions lors des mouvements (si nous deplacons la jambes nous voulons que le pieds se deplace aussi). Si a la place de choisir la translation d'un os pour modifier certaine proportions de notre personnage nous decidions de changer la taille alors le changement de taille influencerai les os enfants (il est possible de le faire puis de retrecir les os enfants ensuite mais cela entraine des interactions particuliere entre la taille de l'os qu'il a et celle influencer par son père) Si par exemple nous voulons agrandir la cuisse il suffit alors de deplacer la jambes vers le "bas".

3.4. Autre objet dans Unity.

3.4.1. Le sol

Le sol est present pour avoir un point de repere solide pour l'avatar et qu'il ne flotte pas dans les airs et celui-ci est fils d'un des os du pieds comme ca lors des changement de taille le sol se déplaceras verticalement pour s'adapter a la taille du personnage.

4. Ressources liens et logiciel

- Git : <https://github.com/Lyx-Orl/LIngenieurCostumier>
- Unity : version 6000.1.8f1
- Blender : Version 4.5.2 LTS
- Modele 3D : Mixamo <https://www.mixamo.com/#/?page=1&query=mannequin&type=Character>
No license
- Package ExcelDataReader : <https://www.nuget.org/packages/ExcelDataReader/3.8.0-develop00498>
MIT License
- SystemEncodingCodePage : <https://www.nuget.org/packages/System.Text.Encoding.CodePages/10.0.0-preview.7.25380.108> MIT License