

TP SHADERGRAPH : Création de Shaders

Antoine Laroye - Thomas Barthélemy

18 Mars 2019

Chapitre 1

Création du Graphe

Le but de cette initiation à ShaderGraph est de prendre en main cet outil, afin de pouvoir créer des Shaders en ayant un feedback en temps réel. Pour cela, on met en relation des noeuds, qui ont plusieurs utilités au sein d'un graphe acyclique non orienté.

Le premier parti pris réalisé est de ne travailler que sur la composante "Emission" du master. Avant cela, on relie aux composantes Albedo, Metallic et Normal des emplacements pour des textures, et on permet à l'utilisateur de les modifier dans l'interface Unity. Ensuite, on choisit notre modèle pour travailler, et on va choisir un robot, nommé Kyle d'après l'Asset Store. Après avoir fait plus ample connaissance avec ce dernier, il est l'heure de se mettre au travail.

Kyle nous a dévoilé son plus grand complexe, il ne se sent pas crédible en tant que robot, il se trouve trop mignon et souhaiterait avoir l'air un peu plus inquiétant. Tout d'abord, nous allons lui apporter une sorte de halo rouge qui va parcourir son corps, comme s'il était méchant (nous en conviendrons, rouge c'est méchant). Ainsi, pour donner cet effet, on va créer un noeud "Effet Fresnel" qu'on multiplie avec une couleur rouge. D'autre part en combinant les noeuds "Time", "Twirl" et "Gradient Noise" on peut avoir un bruit distordu en forme de spirale qui évolue en temps réel. On multiplie le Fresnel avec notre bruit et on obtient alors notre halo rouge qui évolue en temps réel.

Ensuite, pour donner un côté plus "endommagé" au robot, on va ajouter au shaders des petits segments bleus animés, comme des fils d'énergie qui seraient continuellement alimentés. Pour cela, on part d'un noeud "Voronoi" auquel on applique deux Step différents, puis on les soustrait pour avoir des formes d'arcs de cercle, auxquels on applique une couleur bleue. D'autre part, on utilise une texture 2D de dégradé de noir vers blanc dont on augmente la fréquence et auquel on applique un noeud "Time" pour avoir un mouvement vertical. En multipliant la texture avec les arcs de cercle bleus, on obtient le résultat souhaité. Il ne reste qu'à sommer les arcs avec le halo rouge afin de l'appliquer au noeud "Emission" du master.

Après un retour sur expérience avec Kyle dans la fenêtre de Unity pour voir le résultat, il nous a félicités et est très heureux du résultat.

Chapitre 2

Remarques

2.1 Présentation

La présentation vidéo est très claire, il y a juste une petite difficulté qui peut apparaître la première fois qu'on lance ShaderGraph, une erreur renvoyée à cause de la pipeline PBR. Une petite recherche sur le net a permis de mettre en évidence qu'il faut préciser une pipeline de rendu dans Unity afin de pouvoir utiliser ShaderGraph. De plus, certaines version de Unity paraissent ne pas pouvoir faire fonctionner ShaderGraph, il faut préférer les versions 2.X. Dernière petite remarque sur un point qui m'a piégé, CTRL+S ne marche pas dans ShaderGraph, il faut obligatoirement cliquer sur "Save Asset", sinon le shader n'est pas rendu dans les fenêtres Game et Scene.

La difficulté de ShaderGraph réside dans le niveau de perfectionnisme de l'utilisateur. On peut faire des shaders plutôt corrects rapidement, mais il faut compter plusieurs heures pour faire quelque chose de vraiment soigné, avec des effets qui flattent la rétine. Le temps de suivre le tutoriel et de faire ce shader, nous avons dû prendre 4 à 5 heures (sans être en effort intense). De plus, il est aisé de se laisser prendre au jeu de vouloir essayer tous les noeuds, de les combiner, il faut donc réussir à un moment donné à se dire "Stop, j'arrête!". Mis à part cela, ce tutoriel était très agréable.