Cours sur Unreal Engine 4

1.0 Introduction :

Date de création : 09/08/18  
Dernière modification : 10/08/18   
Version compatible avec ce cours : Toute les versions supérieur à la 4.0  
Nombre de chapitre du cours : 3

Unreal Engine est un moteur de jeu, créé par Epic Games, permettant de développer des jeux sur différent support et avec plusieurs possibilités de création. Il est orienté pour les jeux à la 1ère personne même s’il tant à se diversifier avec la 3ème personne et la 1ère personne sur une voiture (même s’il est possible de créer un jeu à la 1ère personne avec une voiture, non-dépendant du contrôleur). Il est très connu pour avoir créé des jeux comme Fornite ou Unreal Tournament. Il peut s’associer à tous les styles de jeux vidéo : combat ou action comme Fornite ou horreur comme Outlast 2. Egalement, il est adapté pour la création de jeu 3D mais un peu moins pour la 2D.

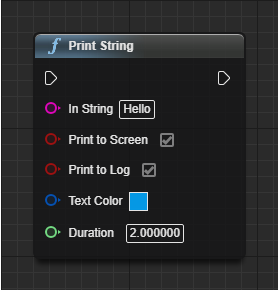
1.0.1 Support d’utilisation :

Le moteur de jeu est disponible sur Linux (Ubuntu), Windows (version supérieur à Windows 7) et Apple (supérieur à OSX 10).

1.0.2 Histoire du moteur

Unreal Engine est à la base fondée sur une startup, elle-même crée en 1991 sous le nom d’Epic Games par Tim Swenney. L’éditeur est publié le 22 mai 1998 par GT-Interactive.

2.0 Principe de programmation

Le langage n’est ni plus ni moins un script, L’UnrealScript crée afin de permettre a ce langage la possibilité d’être plus simple et plus robuste. Le support de création des composants est appelé Blueprint.   
Le code se présente comme ceci :   
  


Cette exemple permet de voir comment s’organise un blueprint ;  
Le signe premier > indique qu’il doit être relié à un brun. Si l’on veut déclencher l’événement « Print String » (écrire sur l’écran), ce brun doit être relié à un événement quel qu’il soit.   
Le deuxième > permet de relier cette événement à un deuxième événement. L’analyse de « Print String sera plus détaillé dans sa partie qui lui sera consacré »

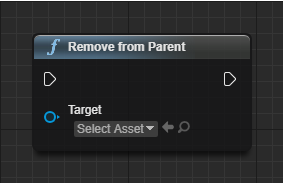
2.1 Programmation Orienté Objet :

La POO est une méthode de programmation qui permet de rassembler des objets avec leurs parents, de faire communiquer des classes avec des objets et vice-versa, de faire communiquer des tableaux avec des objets et de faire communiquer des objets entre eux. Justement dans Unreal Engine, le travail est mâché : grâce à une communication par chevron « < ; > » les objets sont directement reliés entre eux. Mais il est impossible dans certains cas de faire communiquer des parents leurs enfants ou héritage de classes. Du coup une seconde méthode est crée et il s’agit de la méthode « Parents To Child » qui grâce a un blueprint rassemble Parents et Enfant et simplifie l’héritage de classes.

2.2 Programmation Parents To Child :

La PPC est une méthode secondaire dans Unreal Engine qui permet de faire communiquer Parents et Enfants entre eux ainsi que l’aide à l’héritage de classe.   
L’exemple le plus flagrant est la fonction « Remove From Parents » qui permet de supprimer un enfant depuis sont parents dans la classe Widget.

Ici la fonction Target permet de faire communiquer la classe Widget (Celui de notre choix) avec l’événement « Remove from parents » et de supprimer le Widget



La programmation Parents To Child permet donc de donner une action à un, voir des enfants depuis le parent.   
/!\ A noter qu’un enfant ne peut avoir une action sur son parent excepté la demande d’information ainsi qu’un héritage possible de classe que s’il s’apparente à cette dernière. Il ne peut ni modifier ni supprimer son parent. /!\

3.3 Programmation numéraire :

La programmation numéraire (qui s’utilise dans Unreal Engine 4) est une méthode de programmation qui revient à mélanger les classes numéraire et non linéaire. Dans ces classes, on retrouve les « Floats », nombre à virgule, les « Integer », nombre entier et les « byte », 0 ou 1. Il reste également les coordonnées qui sont numéraire mais ce sera pour plus tard. Donc cette méthode de programmation revient à la possibilité de les convertir entre eux toujours avec un Blueprint qui leur correspond.

La première chose à savoir est qu’Unreal à un code couleur pour les classes de données. Les float sont coloré en vert très clair, les integer sont coloré en vert foncé et les bytes en vert très foncé. Le code est compliqué mais on si fait très facilement.