Université de Montpellier
Master 1 - IMAGINE

# Moteur de jeux

Projet informatique — HAI819I

**Étudiants :**
M. Marius JENIN
M. Alexandre LANVIN

**Année :** 2021 - 2022

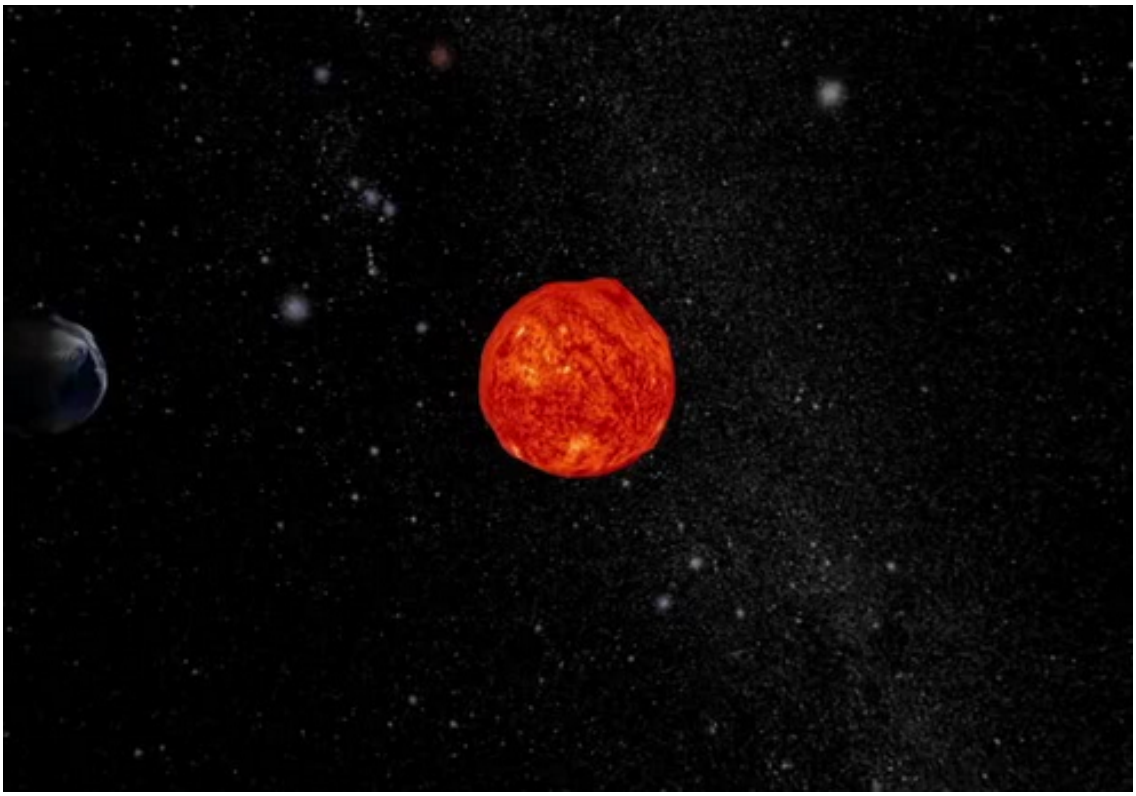
# Table des matières

# Chapitre 1

# Graphe de Scène

## 1.1 Structure du graphe

Le graphe est composé de différents noeuds, un noeud a un noeud parent et une liste de noeuds enfants. Il possède aussi une transformation locale et une transformation globale. Ces transformations sont stockées dans des mat4 elles contiennent les informations de rotation, translation et mises à l'échelle du noeud. La transformation locale sert à appliquer une transformation au noeud sans l'appliquer aux noeuds enfants de celui-ci.



# Chapitre 2

# Moteur physique

## 2.1 Mouvement

La physique est mise à jour indépendamment du taux de rafraîchissement. On a donc besoin d'équations qui approchent les résultats idéaux. Comme nous voulions garder notre moteur de jeux le plus paramétrable possible nous avons ajouté la possibilité de calculer les positions des RigidBodyVolumes de différentes manières un utilisant différentes ode *(Ordinary Differential Equation)* telles que Euler, Verlet ou encore Runge-Kutta 4.

## 2.2 Collisions

Pour résoudre les collision nous nous sommes inspirés du game physics cookbook en utilisant des impulsions linéaires et angulaires pour affecter une translation et une rotation aux objets lors d'une collision. Nous avons aussi tenté de corriger les erreurs de collisions en utilisant une projection linéaire.

## 2.3 Comportements

Lors d'une collisions nous avons choisi d'avoir une fonction différente appelée en fonction des comportements que l'on aura affecté aux RigidBodyVolumes. Ainsi le comportement de "base", celui d'une réponse proche de la réalité à une collision correspond à la classe MovementBehavior mais nous avons d'autre comportements comme le SwitchColorBehavior qui modifie simplement le matériel affecté au RigidBodyVolume lors d'une collision.

# Chapitre 3

# Rendu

## 3.1 Modèle de rendu

Nous avons fait le choix d'utiliser le modèle de Blinn-Phong car celui-ci est simple d'utilisation et rapide à calculer. Il donne également des résultats satisfaisant si nous paramétrons correctement nos matériaux.

## 3.2 Matériaux

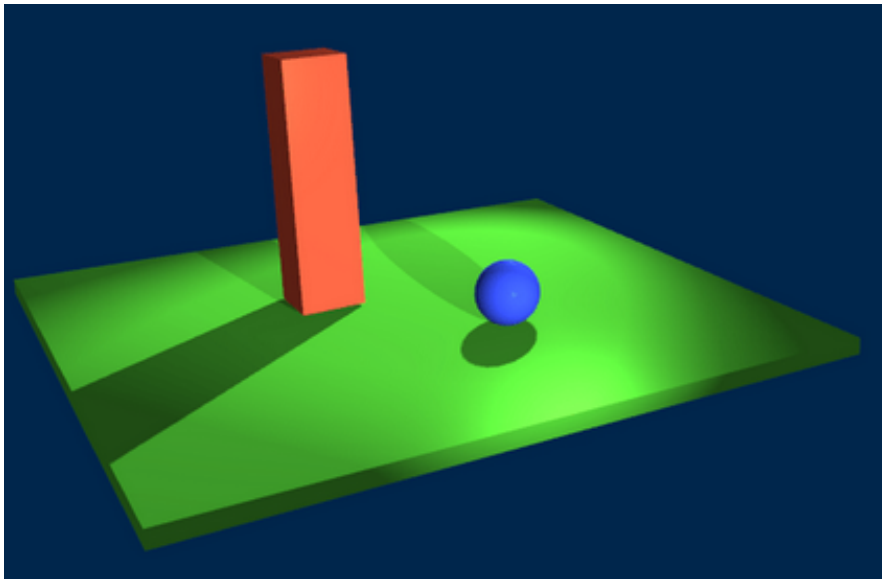
Nous pouvons avoir des matériaux unis avec des couleurs ambiante, diffuse et spéculaire différentes ou alors des matériaux texturés sur la composante diffuse et spéculaire.

## 3.3 Lumières

Il y a 3 types de lumières dans notre moteur. Tout d'abord les lumière ponctuelles, les lumières uniquement dirigées et les spots. Ces derniers sont des lumières avec 2 angles pour permettre d'avoir des contours adoucis.

## 3.4 Shadow Map

Avec des spots nous avons accès à des ombres générées par des shadows map. On effectue une passe par lumière dans la scène pour générer les cartes de profondeur. On utilise ensuite ces cartes comme des shadows map dans notre scène.



# Chapitre 4

# Interactions

Nous avons créé une scène "Laboratoire" permettant de se déplacer avec un bras robotique et d'ouvrir des portes en lançant des objets sur des boutons. Notre moteur permet beaucoup plus d'actions mais nous nous sommes concentrés sur les fonctionnalités du moteur plutôt que sur la création d'un réel jeu.