

Recette

M2 Image et Multimédia – Chef d’œuvre

Encadré par Charly Mourglia et Valentin Roussellet

Rédigé par : Fabien BOCO, Anselme FRANCOIS, Dimitri RAGUET

Table des matières

Présentation du logiciel..... 4

Travail réalisé 5

 Rappel de la milestone n°1..... 5

 Résultat**Erreur ! Signet non défini.**

 Rappel de la milestone 2 7

 Résultats 7

Présentation du logiciel

Le but de ce chef d'œuvre était d'implémenter un moteur physique en se servant de l'approche «Position Based Dynamics » décrite dans une publication de NVidia.

En soit ce moteur (comme son nom l'indique) n'est pas un logiciel à part entière mais bien une bibliothèque qui pourra être utilisé par n'importe qui sur un moteur de rendu quelconque.

Afin de tester le fonctionnement de cette bibliothèque, nous avons utilisé le moteur de rendu déjà fonctionnel d'Anselme François autour duquel a été développée une petite interface graphique avec Qt, permettant quelques interactions pour tester le moteur physique.

Vous pourrez trouver les différentes actions faisables dans le manuel.

Travail réalisé

Le développement du projet était divisé en plusieurs milestones, la première représentant le minimum livrable.

Rappel de la milestone n°1

Une scène doit pouvoir contenir des particules. Une particule sera représentée par une sphère avec une position et un rayon. Pour créer la scène, il faut également qu'il soit possible d'ajouter manuellement des particules ou bien d'en charger à partir un modèle.

La scène de test consistait en une scène dans laquelle 1000 particules sont contenues dans une "boite" invisible. Il devra paraître visuellement évident que les contraintes sont respectées, à savoir que les particules ne se rentrent pas dedans et ne sortent pas de la boite.

Résultats

Les objectifs de ce premier jalon ont été largement atteints. Nous sommes à ~35 FPS avec 1000 particules et le solveur réglé sur 3 itérations. Augmenter un peu le nombre d'itérations à ce stade ne fait pas chuter les performances, pas plus qu'elle améliore la physique de manière notable.

L'interface de test développée permet comme prévu d'ajouter le nombre de particules souhaité ou de les charger à partir d'un modèle (ce qui n'a pas vraiment d'intérêt étant donné que les particules sont indépendantes et se dispersent tout de suite).

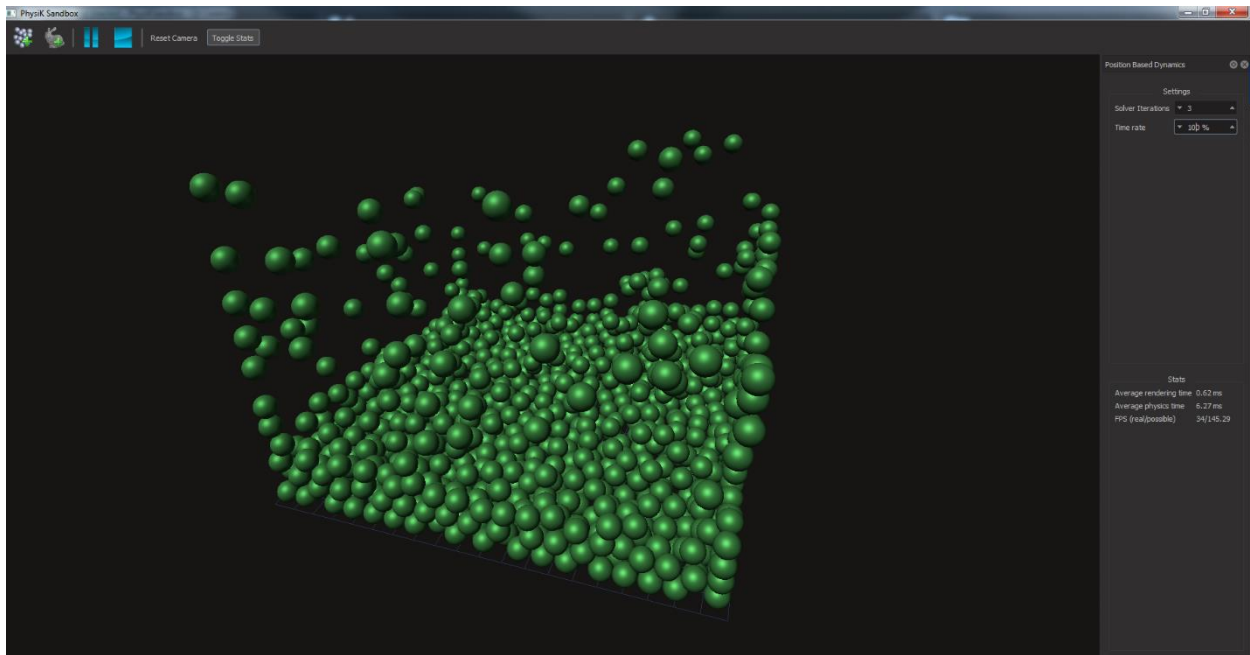


Figure 1- Scene "Piscine à boule avec 1000 particules"

Rappel de la milestone 2

La scène doit pouvoir contenir des triangles (donc des mesh) et gérer leur intersection avec des particules ou des segments.

Résultats

On devait pouvoir observer les réactions physiques entre 2 rigidbody. On ne peut pas considérer cette milestone comme terminée dans la mesure où de sérieux problèmes de performances (non résolu) se posent à la simple application de la gravité sur un seul rigidbody.

Cependant les rigidbody kinematic (qui ne sont soumis à aucune force ou impulsion) permettent d'observer le bon fonctionnement de la collision particules/triangle. On reste temps réel avec la scène suivante :

Contenu de la scène :

- ~1000 Particules
- "Rigidbody Sandbox" (boite dans la capture ci-dessous)
- Nombre d'itérations du solveur : 1

Performances : ~30 FPS

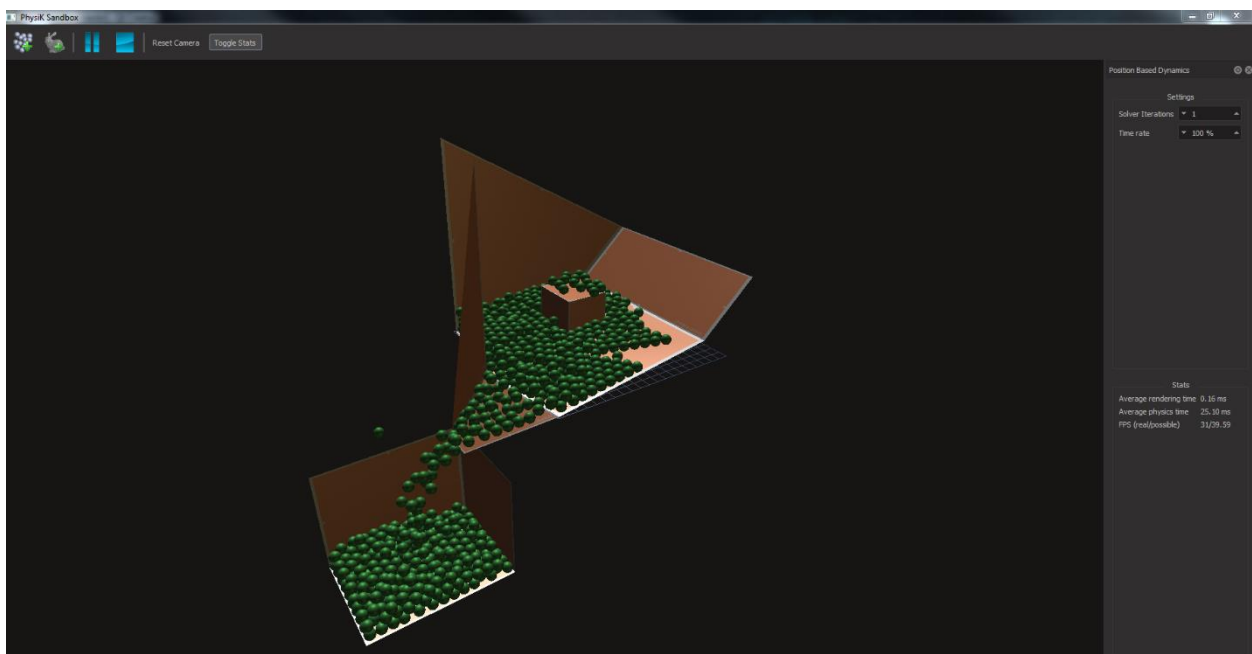


Figure 2- Particules & rigidbody

Milestone 3

On ne peut pas dire que la milestone n°2 ait été totalement complétée, cela dit, étant donné que la milestone n°3 consistait seulement à ajouter des contraintes de type pour simuler des « soft-body » nous avons commencé à les implémenter.

Le résultat nous donne un tapis bien soumis à la physique et qui se déforme sur le sol. Les performances sont bonnes (~45 FPS) mais les déformations sont très instables et le tapis a tendance à trembler et glisser tout seul sur le sol.

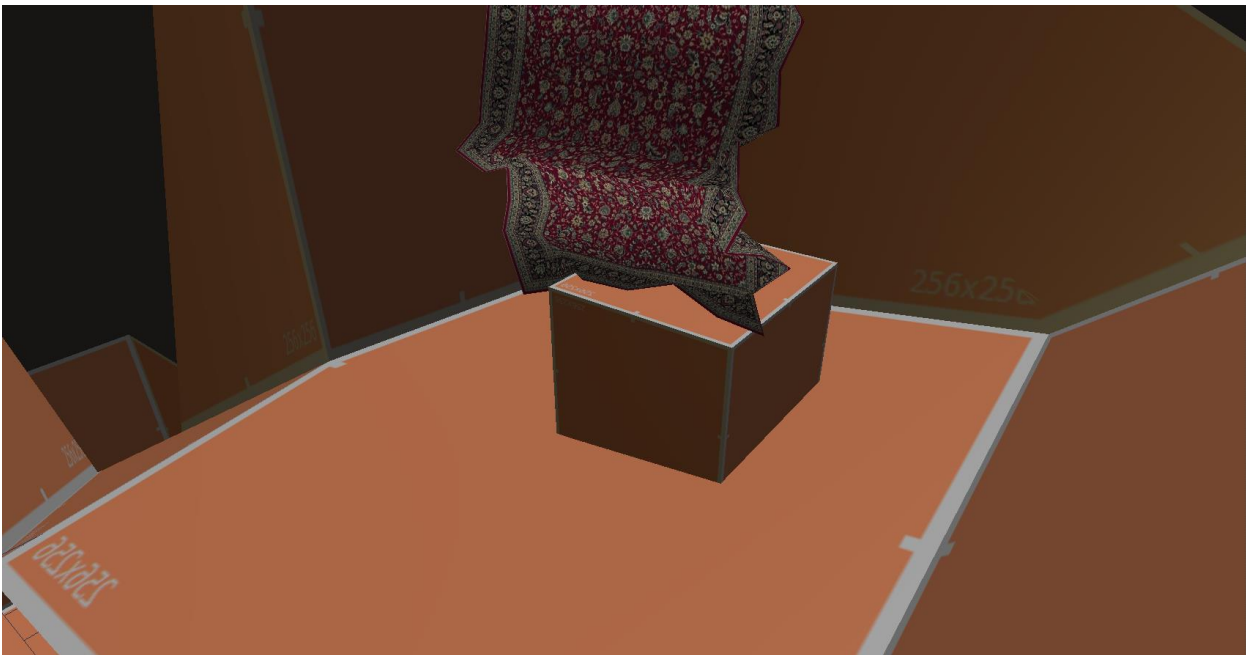


Figure 3- Tapis tombant

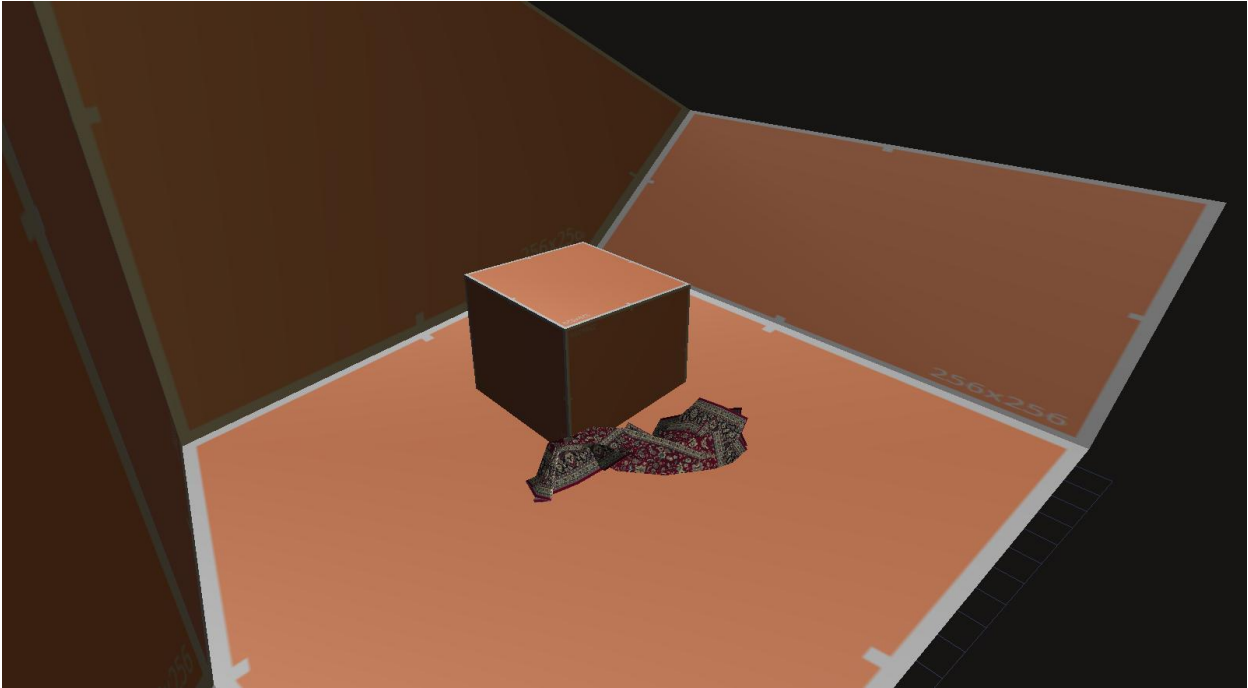


Figure 4- Tapis au sol

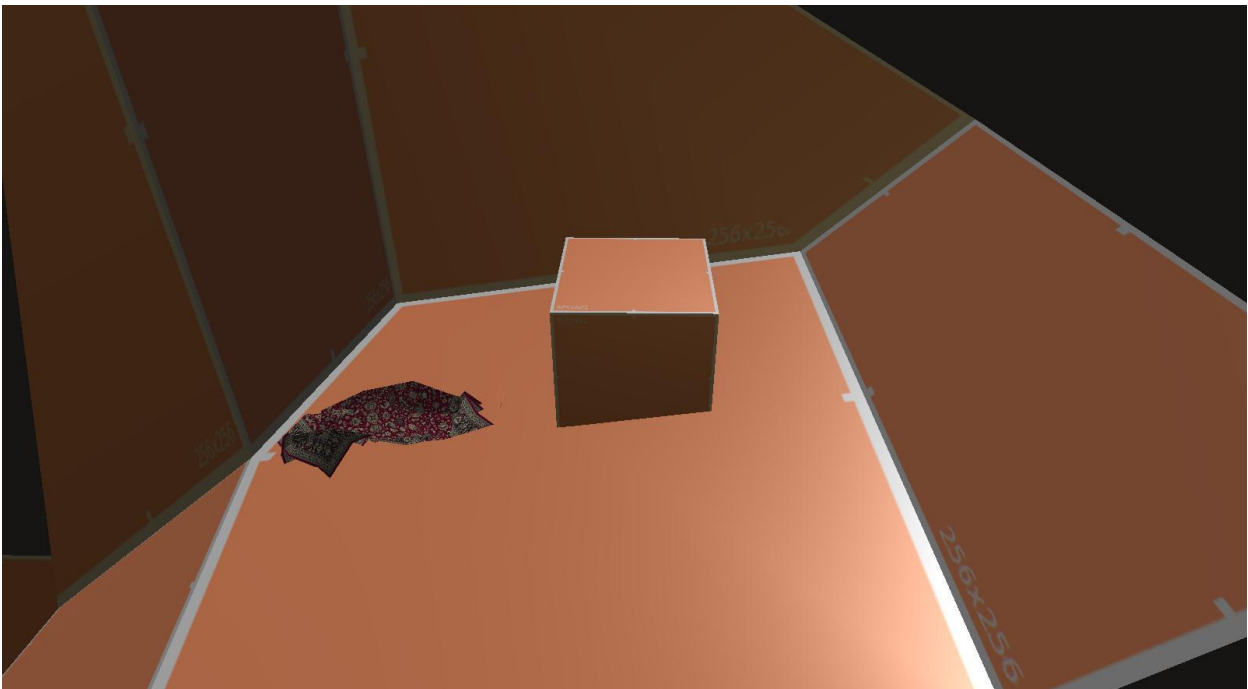


Figure 5- Tapis qui a bougé pour rien...