



**Forum: Moteur de jeu GameBlender et alternatives**

**Topic: Simulateur d'entrepôt, script et BGE : - urgentissime - comment modifier dynamiquement une action ?**

**Subject: Re: Simulateur d'entrepôt, script et BGE : comment faire bouger un objet ?**

Posté par: Bobibou

Contribution le : 31/8/2011 21:01:24

Hello mcBlyver !

Ça fait plaisir de te voir du côté BGE de la force.

Ton projet est intéressant, je regrette de ne pas avoir été là plus tôt pour le suivre et t'éviter cette cuisante solitude.

Je commence tout de suite par une remarque qui a déjà été faite mais qu'il est important de définir clairement car on commence tous par se tromper : Blender utilise deux API **totale**ment **indépendantes** en ce qui concerne Python :

- BPY qui sert à créer des add-ons.
- BGE qui est utilisée avec le BGE.

Elles ont chacun leur rôle et ne peuvent pas en changer, c'est pourquoi BPY ne fonctionne pas avec Run Game et BGE ne fonctionne pas avec Run Script qui est réservé aux add-ons. En effet, un script pour le BGE nécessite un contexte et ne peut par conséquent pas être lancé indépendamment du jeu, il pourrait y avoir des situations illogiques.

Ces deux modules sont d'ailleurs clairement séparés dans la doc :

[http://www.blender.org/documentation/blender\\_python\\_api\\_2\\_58\\_release/](http://www.blender.org/documentation/blender_python_api_2_58_release/)

Nous n'utiliseront ici que les Game Engine modules et les Standalone modules qui peuvent être utilisés dans les deux API. C'est important pour ne pas se mélanger les pincesaux mais il me semble que tu n'avais compris.

Pour le mouvement, je suis l'idée d'Oracle qui est d'interpoler grâce à une fonction mais attention : comme j'ai bien expliqué Speedlight, lorsque le script est exécuté, le jeu s'arrête. Donc ce qui se passe avec ta boucle, c'est qu'elle passe par toutes les étapes entre les deux rafraîchissements de l'écran ! Pour y remédier, il faut éliminer ta boucle et utiliser une variable globale servant de repère pour savoir à quelle étape du mouvement on est, une sorte de variable d'avancement. Une variable globale se définit comme ceci :  
bge.logic.variable = "une valeur" ou gl.variable si tu as choisi de remplacer bge.logic par gl en début de script grâce à la ligne « from bge import logic as gl ». (ou GameLogic.variable en 2.49)  
Une fois cette variable définie, on peut la réutiliser dans la formule d'interpolation comme étant le X. Mais attention, si tu fais gl.avancement = 0 dans ton script, cette ligne sera lue à chaque fois et l'avancement restera à 0. Il y a donc deux possibilités :

- Ou tu crées un script Initialisation.py (ou Init.py ou ce que tu veux d'autre mais c'est plus ou moins une convention) qui ne s'exécute qu'une seule fois, tout au début du jeu, et tu y places la ligne de définition de gl.avancement ;
- Ou tu utilises une condition dans ton script qui permet de ne définir gl.avancement uniquement s'il n'existe pas encore. Pour cela, on utilise la fonction hasattr() qui permet de vérifier si

un objet possède ou non un attribut (car une variable globale n'est ni plus ni moins qu'un attribut ajouté à l'objet bge.logic ou gl) :

```
if not hasattr(gl, "avancement") : gl.avancement = 0
```

Ensuite, cette variable globale peut être incrémentée à la fin du script pour que l'étape suivante fasse avancer l'objet.

Note :

Citation :

Note: "Sensor.NEGATIVE == True" n'existe pas, je pensais, mais non. Il se traduit simplement par "Sensor.POSITIVE == False"

Tu peux aussi utiliser « if not sensor.positive : », je trouve personnellement que c'est plus clair.