



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: Speedlight

Contribution le : 31/8/2011 13:55:35

Citation :

Dans cette page fort intéressante, il utilise gl. mais lorsque je fais pareil, j'ai un message "gl is not defined".

["target=" blank">http://www.highspeedracer.webou.net/tuto.php?id=21](http://www.highspeedracer.webou.net/tuto.php?id=21)

Ton message d'erreur signifie que la variable "gl" n'est pas définie, entre autre, qu'elle existe pas parce que tu ne l'as pas créée.

La variable "gl" est souvent utilisée dans le bge pour remplacer le bge.logic (2.5) ou le GameLogic (2.49b)

```
gl = GameLogic cont = GameLogic.getCurrentController()
```

Citation :

Pourquoi un Actuator Motion utilisant "Force" au lieu de "Loc" ne fonctionne que sur des objets sans Parent ?

L'actuator motion utilisant force s'applique sur un objet n'étant pas dépendant d'un autre objet, ce qui altère ces propriétés physiques.

Un parentage force un objet à toujours rester à telle ou telle distance de l'objet. Si c'est en Loc, blender va modifier la position du child pour le parentage.

Citation :

Puisque l'appel d'un script se fait au niveau du Controller, comment appeler un Actuator Motion dans ce script,

alors que justement la présence du Controller de type Python se passe de liaison vers un Actuator ?

Faut-il le créer dynamiquement ?

Pourquoi veux-tu te passer de liaison ? Avec liaison:

```
gl = GameLogic cont = gl.getCurrentController() Actuator =  
cont.actuators["nom_de_l'actuator"]
```

Sans il suffit d'utiliser l'objet. Par exemple:

```
gl = GameLogic Objs = gl.getCurrentScene().objects Actuator_Recherché =  
Objs["L'objet"].actuators["L'actuator_Recherché"]
```

Mais je ne suis pas sûr de la méthode pour lister les objets, je ne l'ai jamais utilisé.

Citation :

La commande "Run script" provoque donc une erreur, qui ne sera pas présente lorsque le script sera

exécuté en mode Game.

C'est méchant, mais c'est nécessaire de le savoir. Et tant pis si je suis le seul (pardon, étais le seul) ici à l'ignorer.

Oui car la commande "Run Script" est propre à Blender, c'est donc une API carrément différente du BGE qui est attendue.

Citation :

2) le mouvement

Dans une boucle for..., je défini à chaque itération la nouvelle position de mon objet. Si

je l'effectue du même coup

un print(position) pour voir les résultats dans la console, tout semble correct, mais mon objet est affiché

uniquement à sa position de départ et d'arrivée. Pas de mouvement visible, donc, mais un saut unique.

J'imagine qu'il manque une commande de rafraîchissement ou quelque chose dans le genre (mais si c'est le cas, alors pourquoi l'objet est quand-même visible à sa position d'arrivée ???)

En fait, lorsque tu utilise un script, il est exécuté entièrement avec ces boucles et tout le tralala, puis laisse la main à blender.

Moment 1:

Position de l'objet = [0, 0, 0]

Moment 2:

Execution du script:

____ Boucle:

_____ Pos = Pos[x]+1

_____ Print Pos

____ Fin

Moment 3:

Une nouvelle frame se lance, actualisation de blender, et des coordonnées de l'objet.

En fait, la position de ton objet a pris toutes les valeurs d'un coup, s'est arrêté à la dernière, et blender a repris la main après l'exécution du script.

Pour bien te rendre compte, fait des tests avec le module time.

Tu fais une scène animée, puis tu rentres ce script avec relié au controller python une brick "always" "true pulse on":

```
import time
time.sleep(0.5)
```

Le script demande juste qu'à chaque exécution du script, il fasse une pause de 0.5 secondes.

Citation :

3) Keyboard Sensor

Le lancement d'un script par l'enfoncement d'une touche effectuée en fait deux fois le script,

Une fois à l'enfoncement et une fois au relâchement. Pour éviter cela il suffit d'activer l'option Tap du Sensor.

Mais il semble que cela ne fonctionne pas toutes les fois (encore à vérifier).

Ou alors tu créé ton script en fonction, et tu englobe le lancement de celle ci dans une condition telle que:

```
gl = GameLogic Sensor = gl.getCurrentController().sensors["Nom du sensor"] if Sensor.positive == True: laDitFonction()
```

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