

# **SDL\_glSetRelativeKebabMode**

Image et Son (mais surtout Image)

CHAMPROY Lilian

DUBROCA Axel



Licence 3 Informatique

Groupe IN601A2

Avril 2016



FIGURE 1 – Tribute to Minecraft, parce que cubes et OpenGL, vous voyez ?

# Table des matières

<b>1</b>	<b>Préambule</b>	<b>3</b>
<b>2</b>	<b>Le programme</b>	<b>4</b>

# Chapitre 1

## Préambule / SDL, OpenGL, GLEW et GLM

Il m'a semblé nécessaire de faire un préambule, afin de prévenir de ce qui va suivre.

Cet exercice a été réalisé sur une configuration différente, puisqu'il n'a pas été fait sur les ordinateurs du CREMI, pour la simple et bonne raison que c'étaient les vacances. Nous avons passé plus de 4 heures<sup>1</sup> à essayer de faire fonctionner OpenGL sous Windows. Pourquoi avoir tenté sous Windows ? Une des forces d'OpenGL, c'est d'être compatible sous toutes les plateformes. Mais ces 4 heures ont suffi à nous faire abandonner l'idée. Nous avons réussi à faire fonctionner SDL, OpenGL, mais GLEW et GLM, c'était impossible (tout du moins, pas trivial). Nous sommes donc passés sous Linux.

Mais alors que nous pensions que ce serait plus simple - l'installation l'était - il a fallu que les choses soient plus complexes également. Les fichiers sources, et corrections ne compilaient pas. Pourquoi ? Parce que la configuration du CREMI est entièrement personnalisée. Comment, nous n'en savons rien. Quelle solution avons-nous adoptée ? Nous avons tout refait depuis zéro. Nous avons appris à développer avec OpenGL en C++, puis nous avons tout traduit en C, au moins pour tout ce qui était possible. La seule chose que nous ne pouvions pas traduire était GLM, qui est pensée avec le C++. Heureusement, le compilateur étant g++, nous avons pu, ignoblement, insérer des instructions C++ dans nos fichiers sources C. Rien que de le dire, j'en ai envie de vomir. Mais nous n'avions aucun moyen d'outrepasser le namespace glm.

Le résultat que vous avez ici peut donc ne pas compiler sur votre machine. Nous n'en savons rien. Nous mettrons donc des captures d'écran, au moins pour que le rapport se tienne à lui-même. Si vous parcourez le code, vous verrez également que le tout est bien plus massif que ce qui était fait en TP. Pourquoi, je n'en sais rien<sup>2</sup>. Mais nous avons tout de même, en quelques jours, appris l'OpenGL de zéro, tout en traduisant le tout en C. Et ça... On en tire une certaine fierté. Non pas d'un résultat parfait, ni correct, mais de savoir que nous nous sommes au moins approchés du résultat final, malgré tout.

Le prochain chapitre - très court - traitera de l'utilisation du programme par l'utilisateur, et le suivant répondra aux questions soulevées par l'exercice.

---

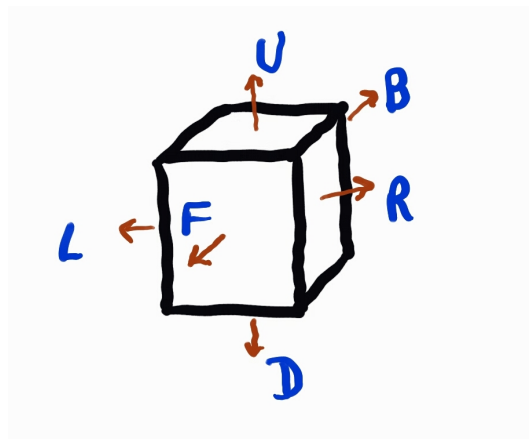
1. Et il n'y a aucune exagération, vraiment, cet exercice a pris, au total, plus de 25 heures à être réalisé.

2. A vrai dire, je pense savoir, puisque nous n'avons assisté à aucun des cours sur OpenGL...

## Chapitre 2

# Le programme / Interaction avec la scène

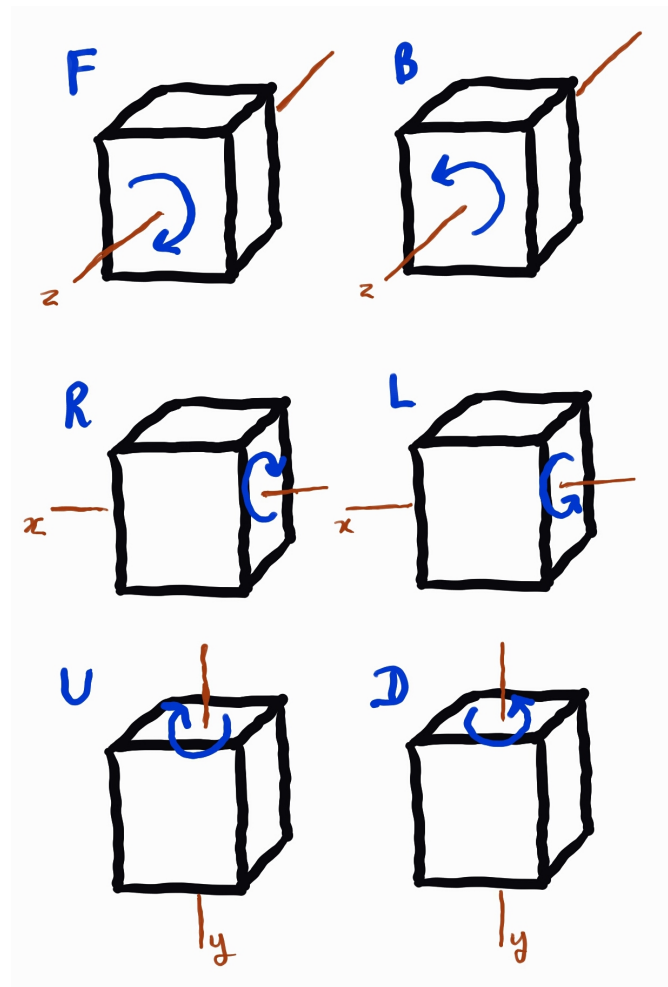
Commençons par une notation pour qualifier les mouvements des cubes. Nous tirons cette notation des Rubik's cubes - c'est ainsi qu'on écrit les algorithmes de résolution du cube. Vous allez comprendre.



(Ne vous inquiétez pas... C'est le fond de l'image qui n'est pas blanc, mais bleuté)

La notation est la suivante : U pour UP, D pour DOWN, F pour FRONT, B pour BACK, L pour LEFT et R pour RIGHT. Simple, non ? Il suffit de considérer que vous voyez, d'une certaine manière, le cube de face. C'est également ce qui va poser notre choix de caméra pour la scène finale.

Du coup, comment décrire les mouvements du cube ? Eh bien, le mouvement "F" indiquera que, si on se place devant la face F, on fait tourner le cube de  $90^\circ$  dans le sens horaire. "B" décrira le mouvement inverse (si on se place devant la face B, et qu'on la fait tourner sans le sens horaire, cela revient à faire tourner le cube dans le sens antihoraire si nous sommes face à F). Voici un schéma récapitulatif :



C'est avec cette notation qu'on décrira les interactions que l'utilisateur aura avec la scène. Actuellement, voici l'association entre les touches du clavier et les mouvements décrits précédemment :

Touche	Mouvement
A	Back
E	Front
Q	Up
D	Down
Z	Right
S	Left