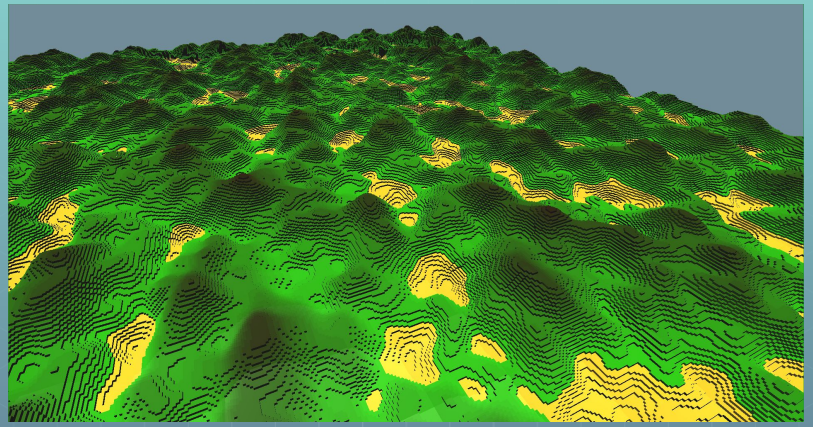
Moteurs de jeux

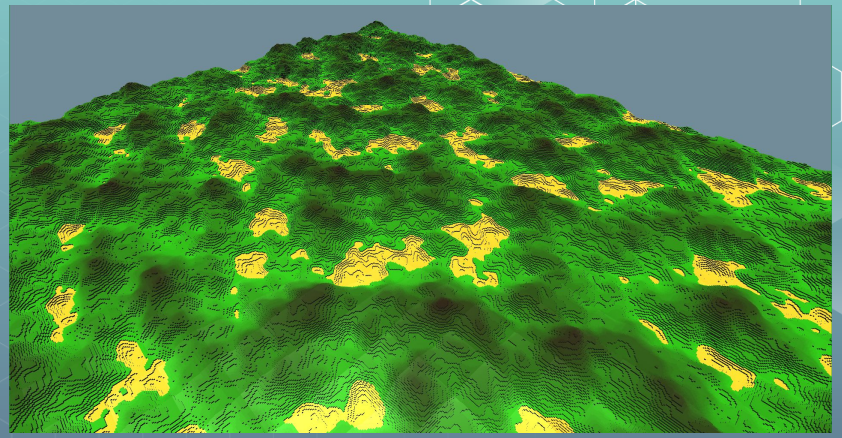


Voxel Engine

Créé et présenté par

ISNEL MAXIME et ODORICO THIBAULT

Enjeu du projet



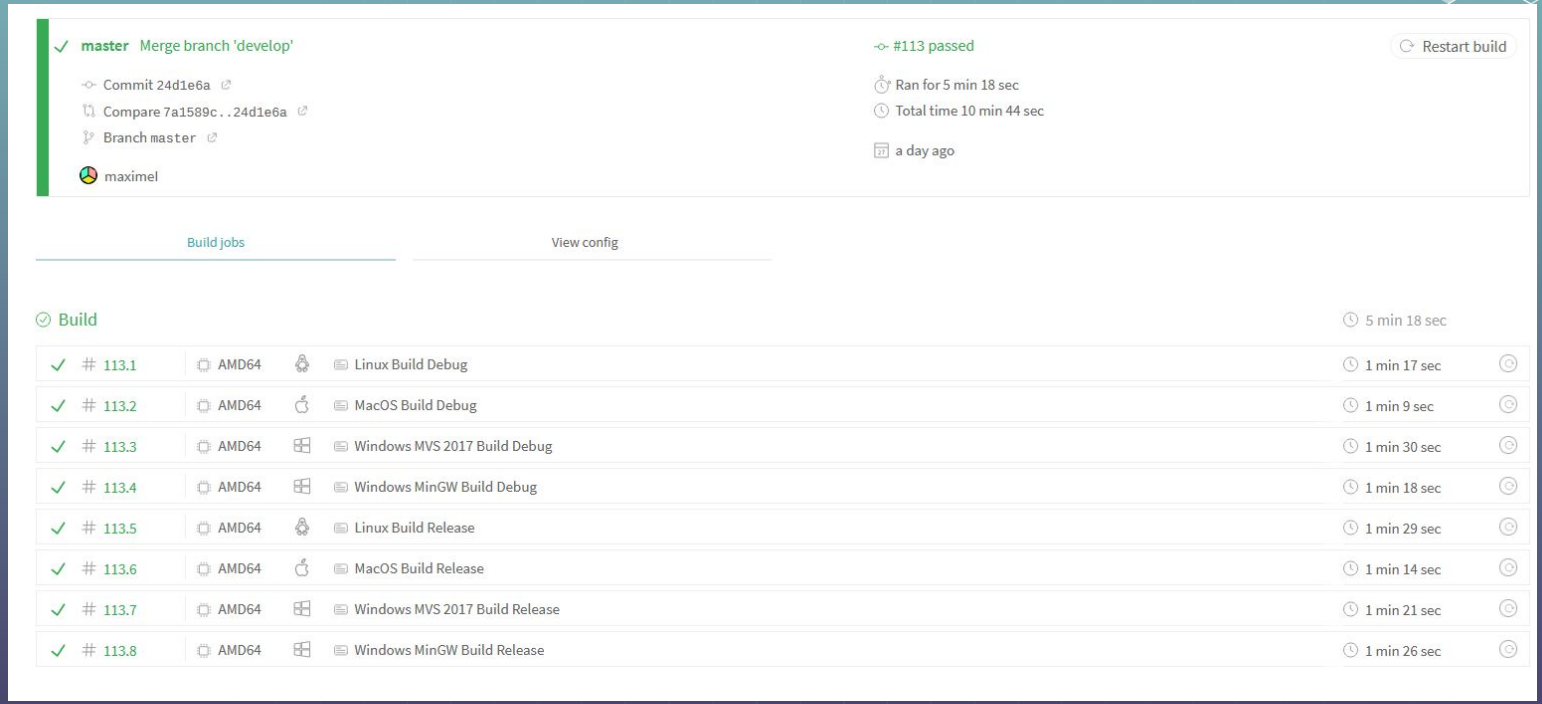
- Faire un moteur de jeu adaptatif
- Faire un moteur à base de voxel
- Génération de terrain procédural
  - terrain infini
- Optimisation pour être temps réel
- Avoir un gameplay qui permet d'explorer

Outils utilisés

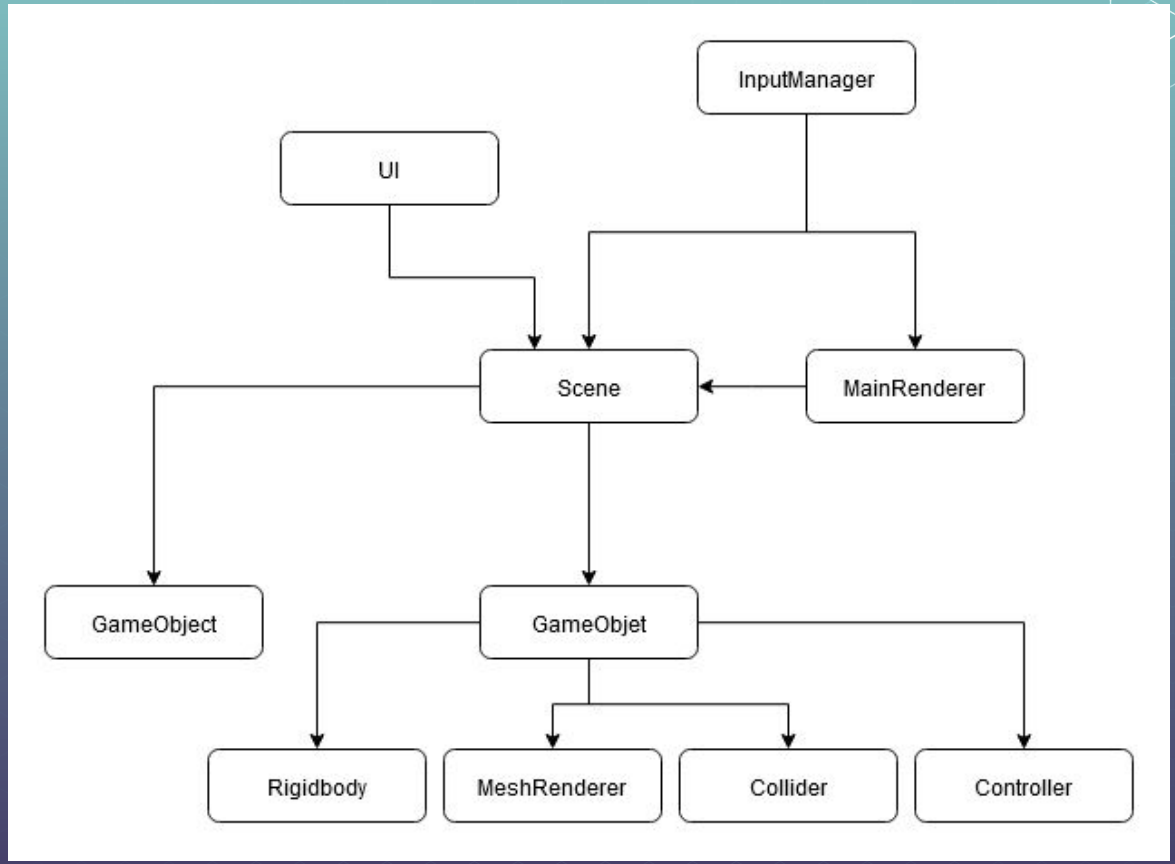
- Langage c++ (version >= c++11)
- OpenGL avec glad
- GLM
- GLFW3
- ImGui
- Cmake
- Git (dépot sur Github)

Cross compile

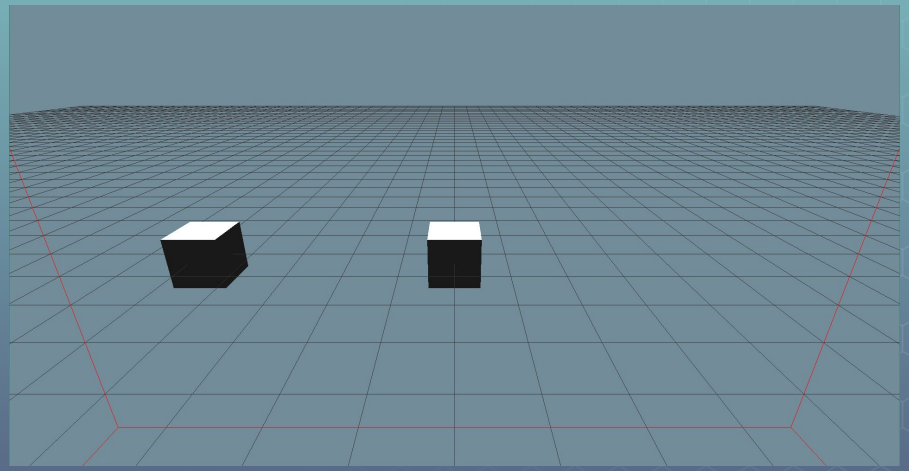
Travis-ci.com



Structure du moteur

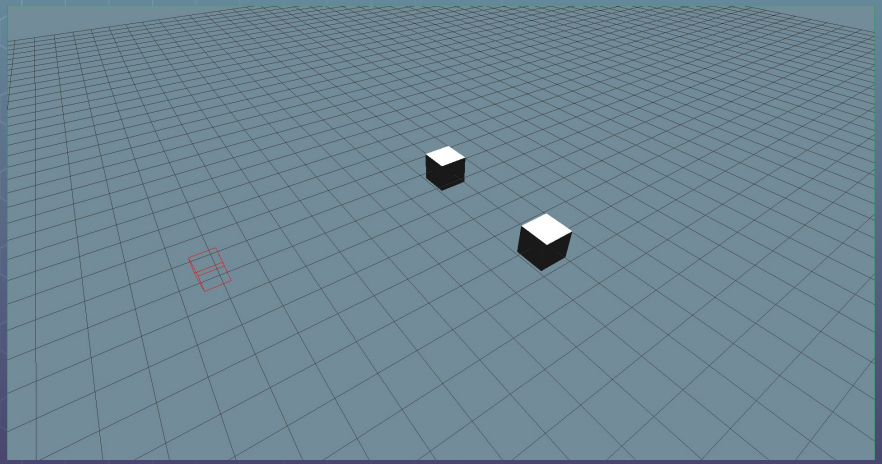


Mode éditeur et mode jeu



Vue du jeu

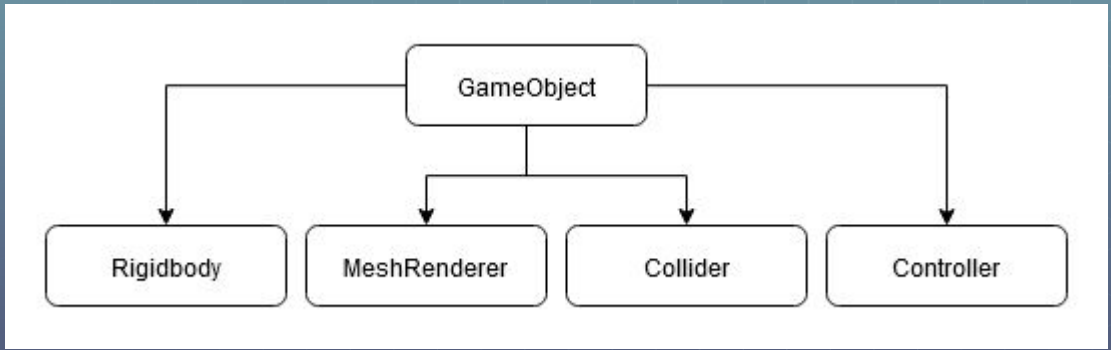
Vue du éditeur



Démonstration

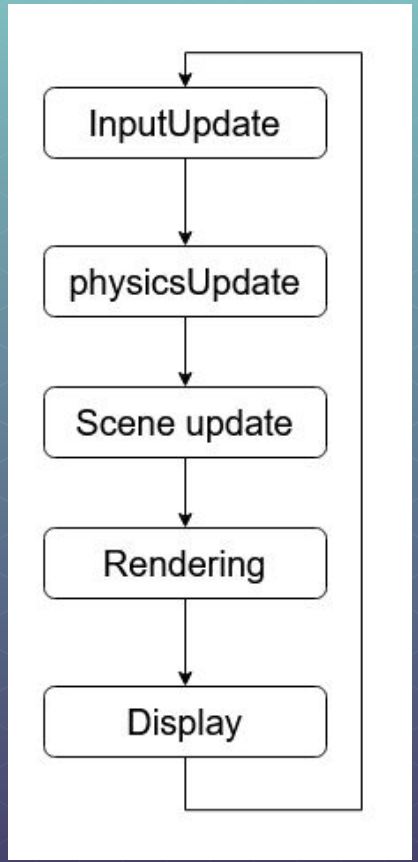
Structure à base de composants

- Séparer le code en fonctionnalités
- Éviter l'héritage infini
- Changer le comportement au runtime



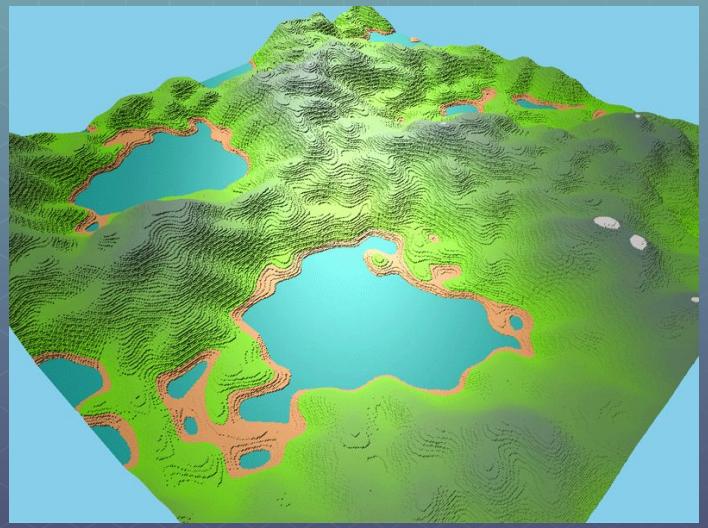
Démonstration

Game Loop


InputUpdate
physicsUpdate
Scene update
Rendering
Display

Voxel engine

- Monde à base de cube
- Génération procedurale
  - monde infini



Voxel engine

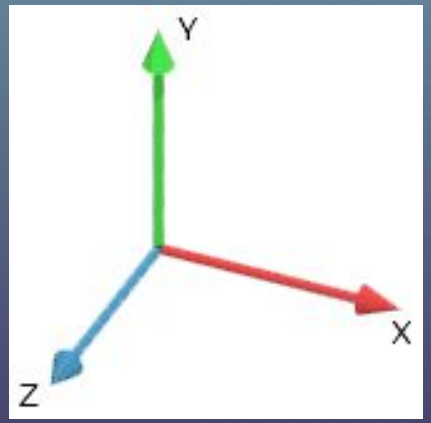
Problématiques

- Gestion de beaucoup de voxels
- Affichage de beaucoup de voxels
- Chargement d'un terrain infini
- Modification en temps réél

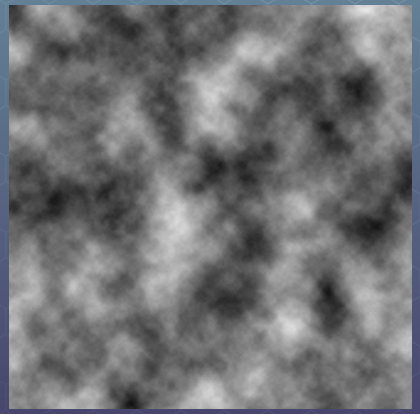
Optimisations

Division en chunk

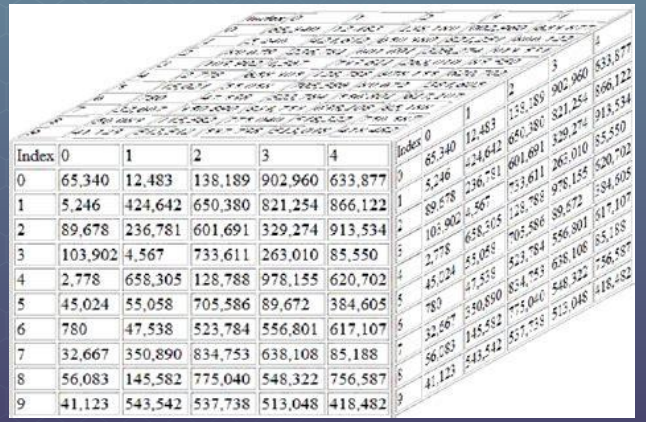
- Position
- Bruit de perlin
- Tableau Cubique



+



=

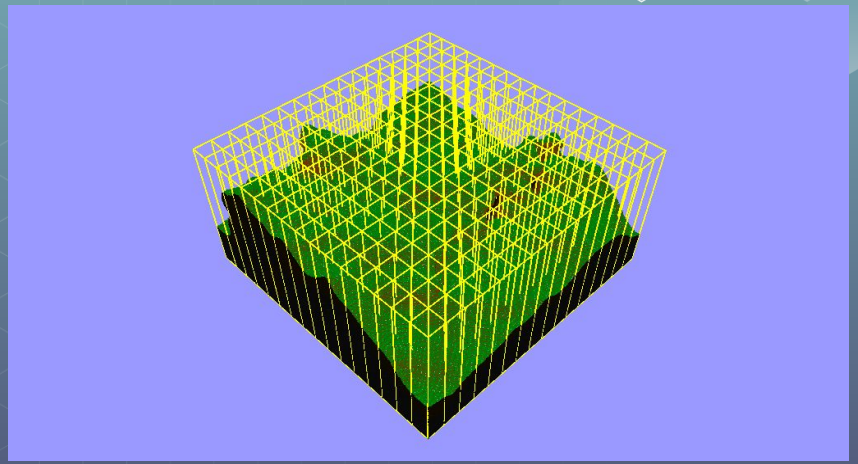


Index	0	1	2	3	4
0	65,340	12,483	138,189	902,960	633,877
1	5,246	424,642	650,380	821,254	866,122
2	89,678	236,781	601,691	329,274	913,534
3	103,902	4,567	733,611	263,010	85,550
4	2,778	658,305	128,788	978,155	620,702
5	45,024	55,058	705,586	89,672	384,605
6	780	47,538	523,784	556,801	617,107
7	32,667	350,890	834,753	638,108	85,188
8	56,083	145,582	775,040	548,322	756,587
9	41,123	543,542	537,738	513,048	418,482

Optimisations

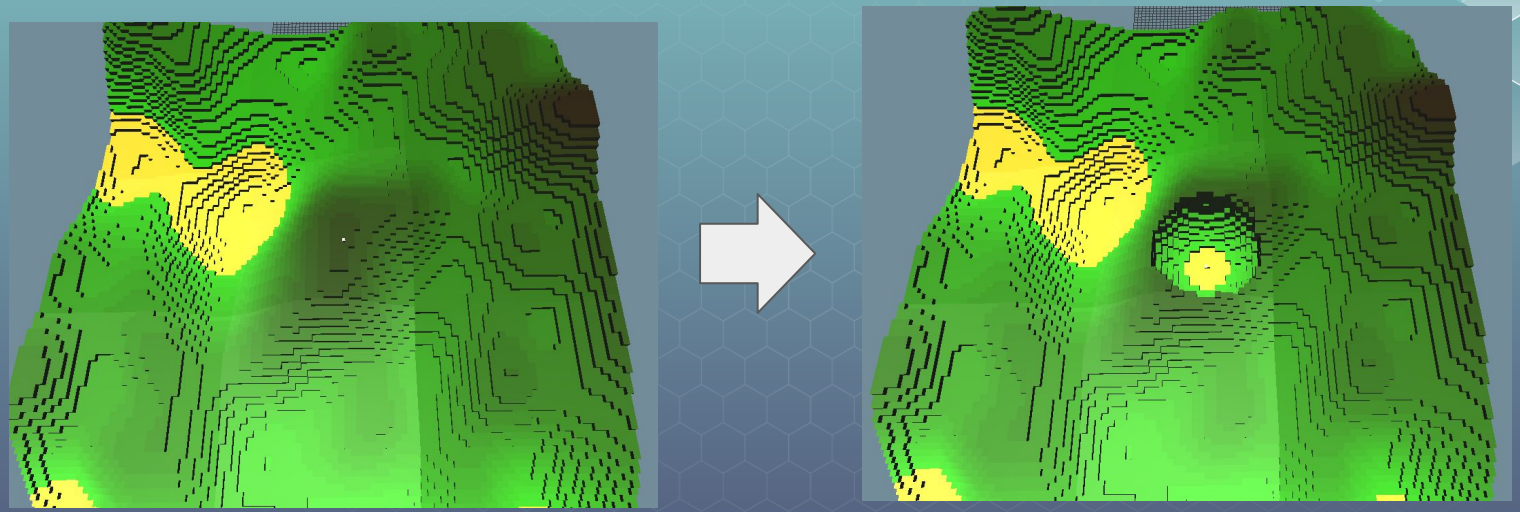
Division en chunk

Index	0	1	2	3	4
0	65,340	12,483	138,189	902,960	633,877
1	5,246	424,642	650,380	821,254	866,122
2	89,678	236,781	601,691	329,274	913,534
3	103,902	4,567	733,611	263,010	85,550
4	2,778	658,305	128,788	978,155	620,702
5	45,024	55,058	705,586	89,672	384,605
6	780	47,538	523,784	556,801	617,107
7	32,667	350,890	834,753	638,108	85,188
8	56,083	145,582	775,040	548,322	756,587
9	41,123	543,542	537,738	513,048	418,482



Optimisations

Division en chunk

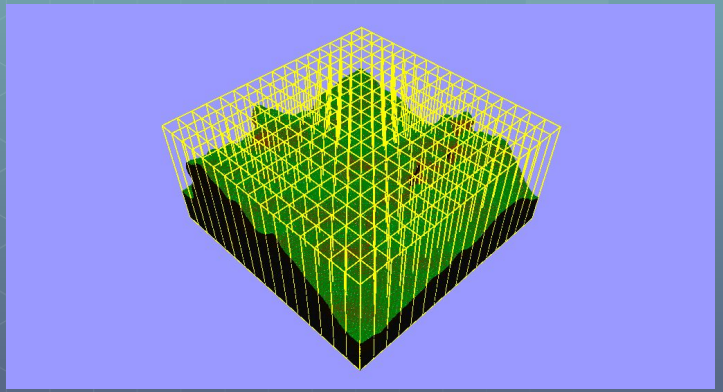


- Modification locales moins coûteuses
- Maillage régénéré à la volé quand nécessaire

Optimisations

Génération de la géométrie

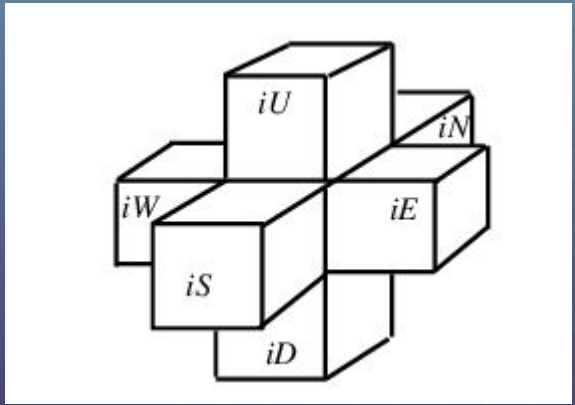
- Cube = 6 faces = 12 Triangles
- Chunk 32x32x32 = 393216 Triangles
- Communication GPU Couteuse

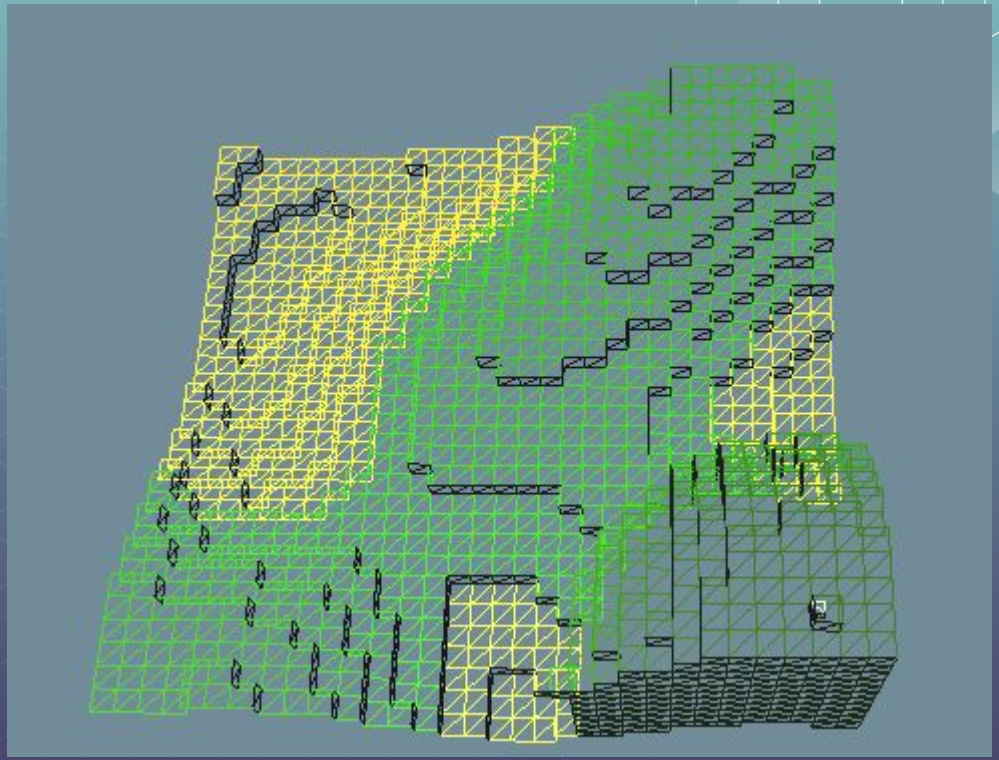


Optimisations

Génération de la géométrie

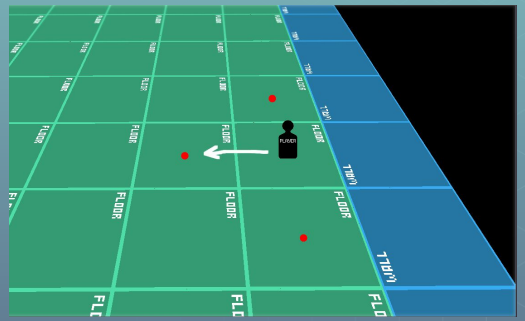
- Ignorer les voxels vides
- Ignorer les faces partagées



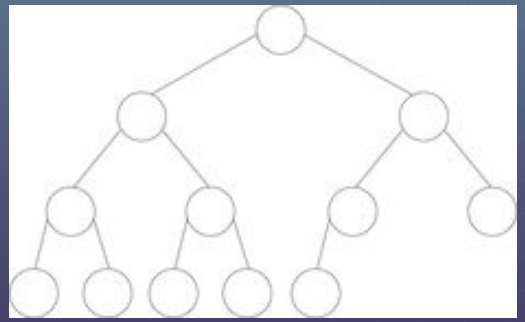


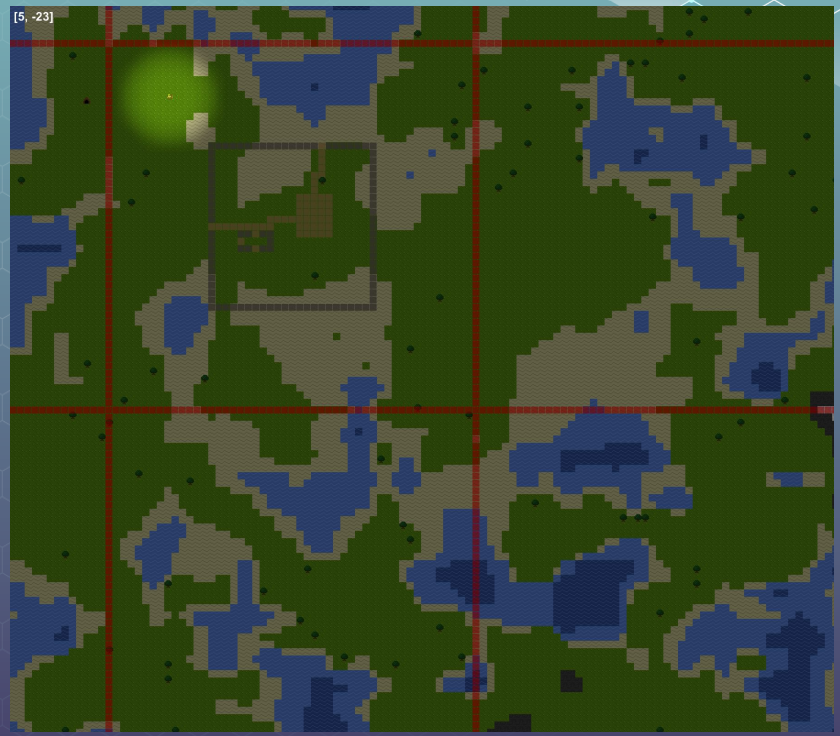
Optimisations

Chargement des chunks



+

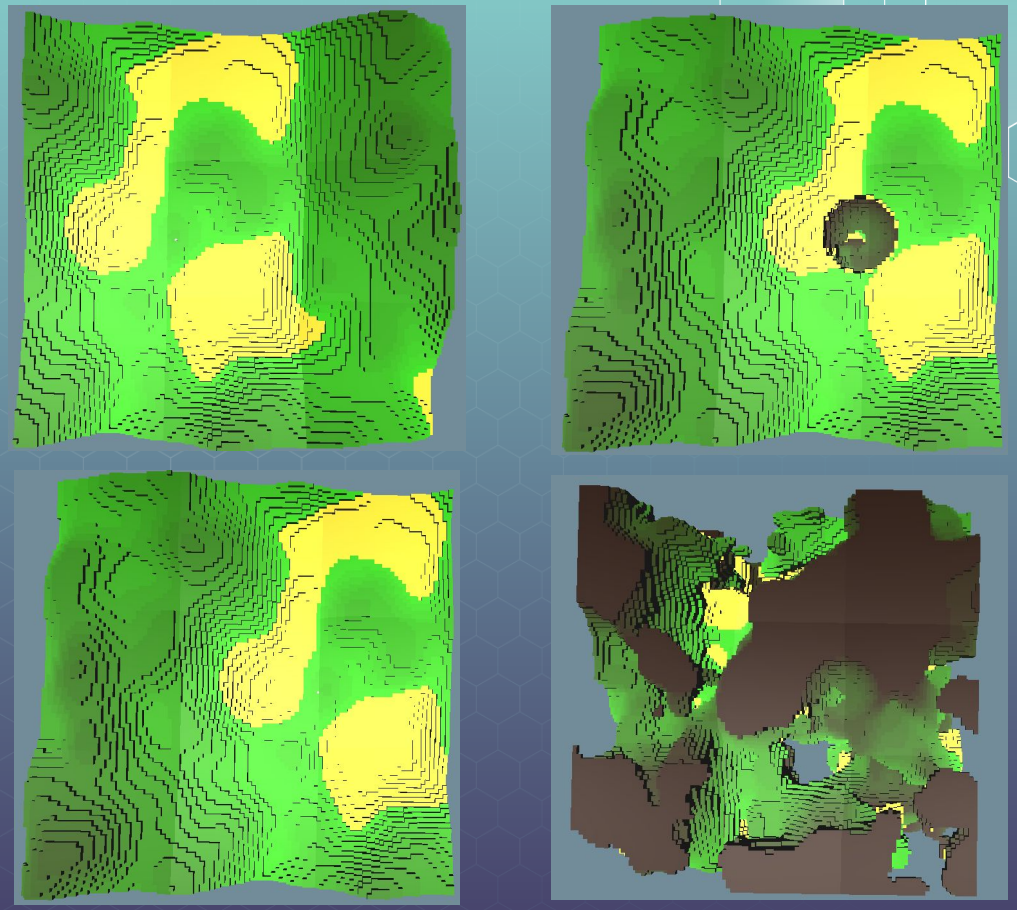




Optimisations

Chargement des chunks

- Déplacement possible sur tous les axes
- Génération uniquement des chunks

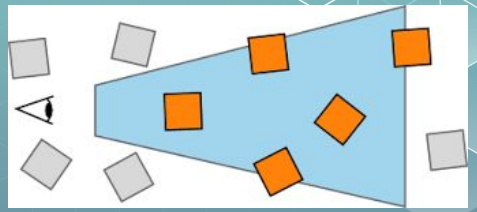


Gameplay

- Mouvement de camera
- Mouvement du personnage
- Explosions

Démonstration

Améliorations



- génération procédurale plus riche
- implémentation “frustum culling”
- collisions entre objets
- meilleur gestion de la caméra
- ajout d’ennemis se baladant dans le monde
- création d’un jeu type tower defense

MERCI POUR VOTRE ATTENTION