

HMIN317 - Moteurs de jeux – TP3

Communication entre objets

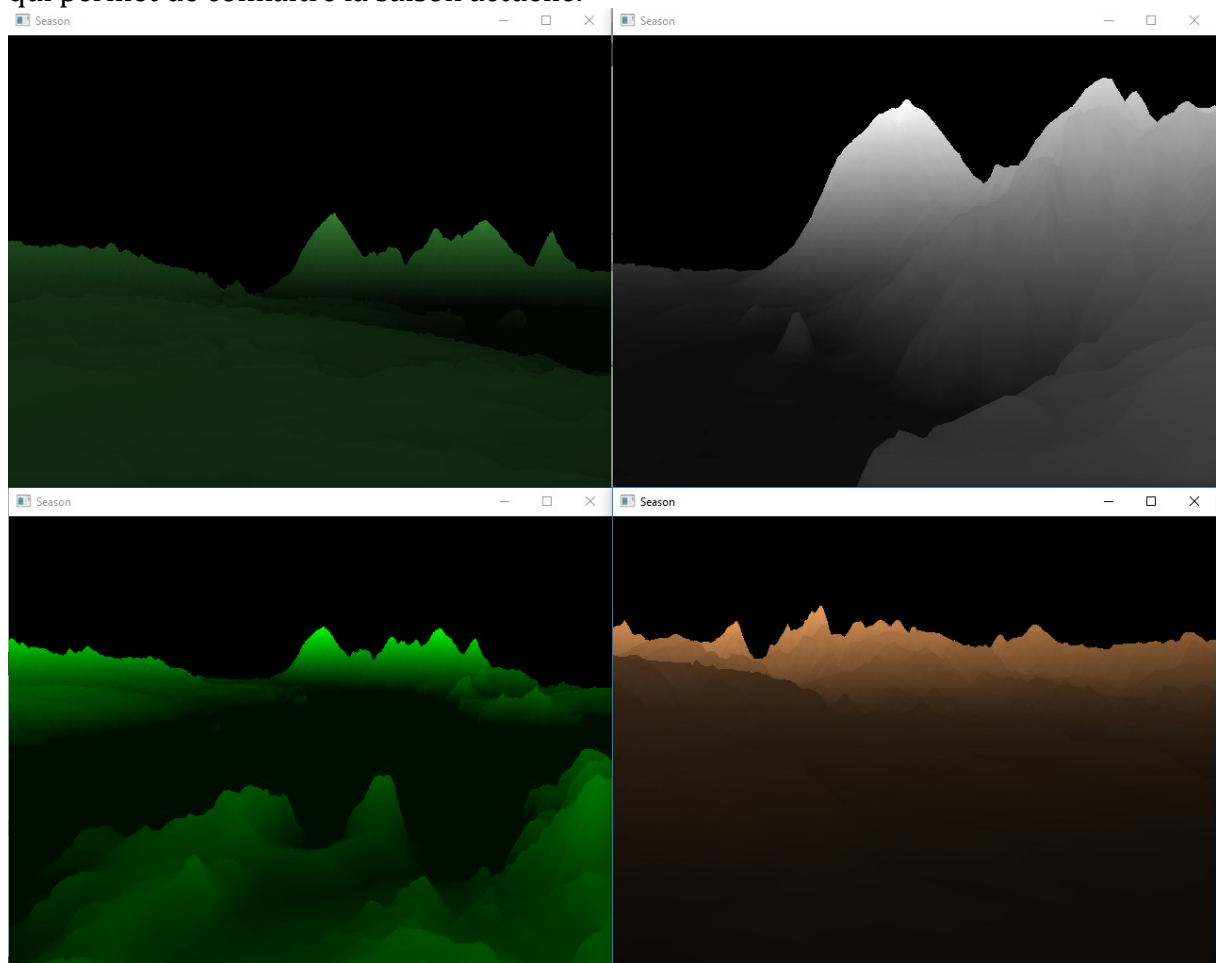
Simulation du passage des saisons

Synchroniser les fenêtres entre elles et changement de saison

Pour réaliser un changement de saison j'utilise un timer qui toutes les 30 secondes appellera une fonction *switchSeason()* qui va permettre a chacune des fenêtres à passer la saison suivante.

```
connect(seasonTimer,SIGNAL(timeout()),this,SLOT(switchSeason()));
```

La couleur du terrain est géré par le programme *shader* et une variable *uniform season* qui permet de connaître la saison actuelle.

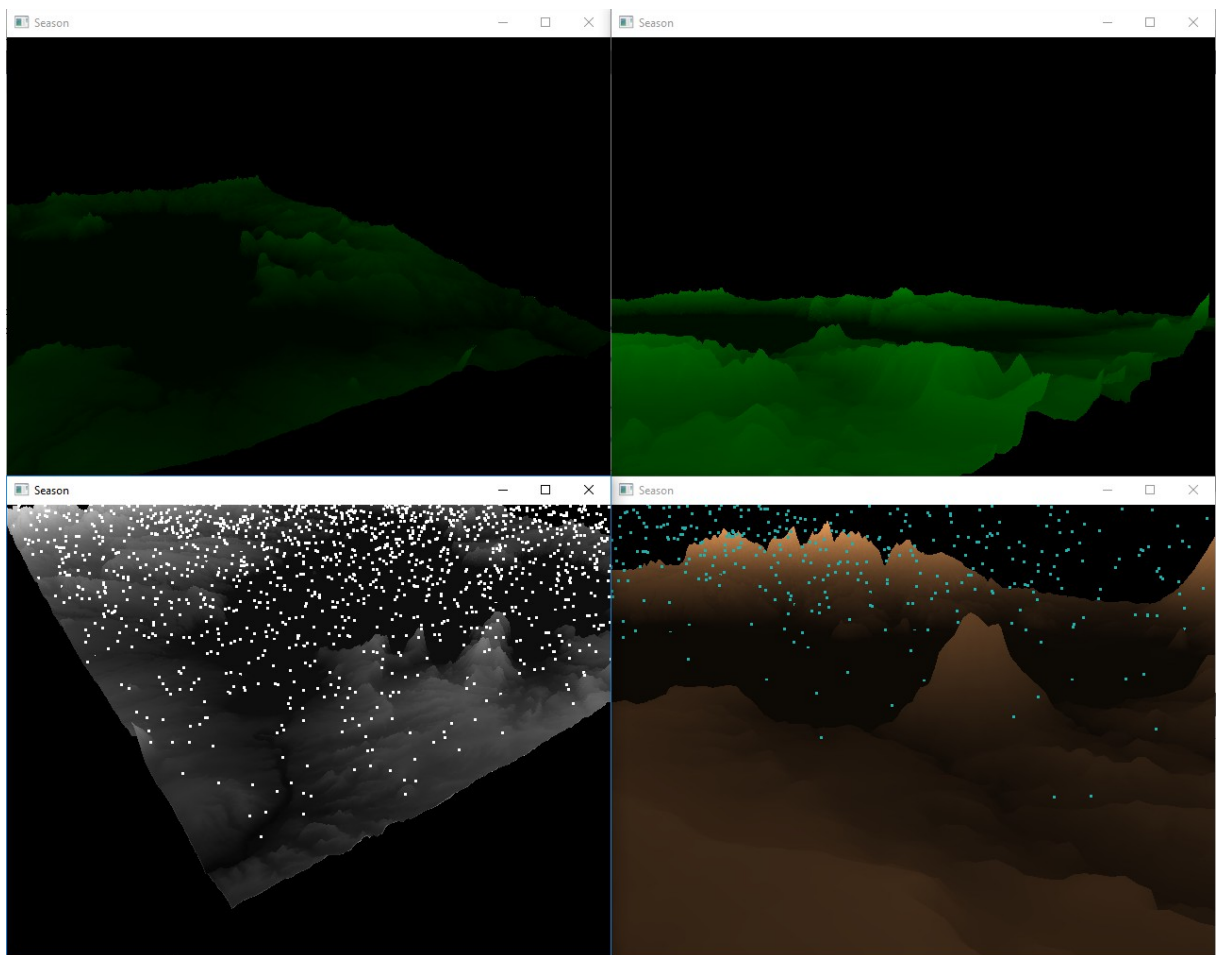


Les particules :

Pour simuler la pluie/neige j'ai créé un objet particule, celui-ci va me permettre de stocker toutes les informations relatives à une particule tel que la position, la vitesse, le temps de vie etc ...

Enfin celle-ci seront gérées par un *emitter* qui va créer les particules et les mettre à jour.

Pour la gestion de collision avec le terrain afin de faire disparaître les particules qui rentrent en contact avec le terrain je me suis servi de la *heightMap* utilisée pour générer le terrain, en effet avec la position de la particule je peux connaître sa position sur la *heightMap* je connais donc la hauteur du terrain à la position de la particule et si l'altitude de la particule est inférieure ou égale à l'altitude correspondante sur la *heightMap* je détruis la particule.



Fonctionnalités supplémentaire :

J'ai rajouté une classe `inputController` gérant les entrées souris/clavier de l'utilisateur et une caméra première personne pour un déplacement plus agréable.

Bonus accumulation

Pour accumuler la neige/pluie on pourrait utiliser une autre texture d'accumulation permettant en fonction de la chute d'eau/neige d'afficher l'eau/neige au sol. Du côté des particules lorsqu'elles atteignent le terrain on met à jour la hauteur d'eau/neige afin d'afficher correctement la texture d'accumulation.

Bonus parallélisation

Dans ce tp j'ai géré les particules à l'aide processeur et ce n'est pas parallélisé en effet si on augmente grandement le nombre de particule on va sentir une chute de performance. L'idéal serait de gérer la vie des particules avec la carte graphique (shader, VBO) qui à l'aide de ses nombreux thread serait beaucoup plus efficace que le processeur