



UNIVERSITÉ  
DE MONTPELLIER

# Compte rendu du TP3

## Moteur de Jeux



## Synchroniser les fenêtres entre elles

Pour synchroniser les fenêtres et simuler le changement de saison j'ai utilisé un QTimer couplé à une énumération des saisons.

En effet je passe au constructeur de mes fenêtres la saison sur laquelle elle vont démarrer.

A l'intérieur de ma classe de fenêtre je connecte le signal timeout() d'un timer à une fonction updateSeason() qui va indiquer à ma fenêtre qu'elle doit passer à la saison suivante. Ainsi je sélectionne la prochaine saison de mon énumération et le timer continue.

Chaque fenêtre représente ainsi une saison différente et toutes les 5 secondes elles changent.

## Simuler des changements de saisons

Pour simuler des changements de saisons j'utilise deux procédés :

- le changement de texture : au préalable je charge 4 textures différentes qui représentent chacune des 4 saisons. Puis à chaque appel de updateSeason() j'applique la texture correspondante à la saison courante sur mon plan.
- les particules : pour simuler la pluie et la neige j'utilise des particules. En effet je génère 10000 particules (représentées par des points OpenGL) pendant l'hiver et l'automne. Puis à chaque update de ma scène je re-génère ces dernières mais un peu plus proches de mon plan pour donner l'illusion qu'elles tombent au sol. Une fois que les particules ont atteint le sol elles sont replacées à une position plus haute. Ainsi on voit apparaître de la pluie et de la neige pendant l'automne et l'hiver.

Pour renforcer cet effet de changement de saison j'ai tenté de jouer avec les lumières mais je n'ai pas réussi. Pour bien réaliser cela il faudrait, je pense, approfondir mon travail sur les shaders que je ne maîtrise pas encore.

J'ai également essayé de changer la couleur du background mais malheureusement une fois qu'elle est initialisée au début du programme je ne parviens pas à la faire varier.



## Capture d'écran du rendu final

