

# Rapport TP1 : SMA - Billes

François LEPAN - Alexis LINKE

17 septembre 2013

## 1 Choix

Nous sommes parti sur un système de déplacement sur une grille case par case. Deux agents ne peuvent se retrouver sur une même case.

Nous avons fait ce choix car le système de collision ainsi que la représentation sur un panel est plus facile à gérer qu'un système de déplacement libre.

Les couleurs des agents sont fixés lors de leurs création.

## 2 UML + explication

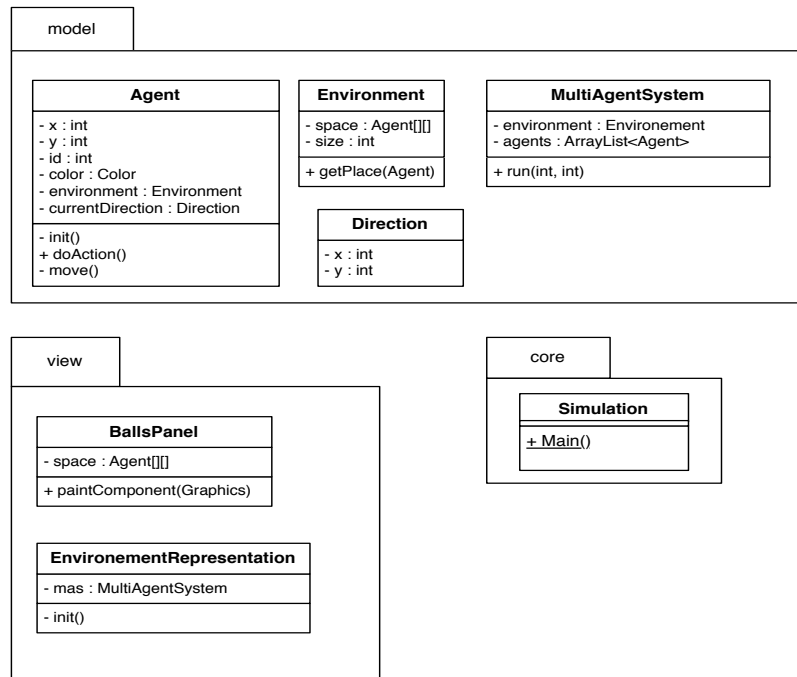


FIGURE 1 – UML

Le design pattern utilisé pour réaliser ce programme est le MVC. Il y a 3 packages : *core*, *view* et *model*.

Le package *core* contient la classe principale **Simulation** qui exécute le programme.

Ensuite le package *view* contient les classes **BallsPanel** qui représente le panel sur lequel on dessine les billes et **EnvironmentRepresentation** qui est la *vue* du MVC ainsi que la fenêtre dans laquelle on ajoute le panel **BallsPanel**.

Et enfin on a le package *model* qui lui contient toutes les classes nécessaire aux calculs de collisions ainsi que de position tout au long de l'exécution du programme. La classe **Agent** contient les données nécessaires pour situer et identifier un agent au sein de l'environnement. La classe **Environment** contient une grille d'**Agent** et possède une méthode **getPlace(Agent)** qui permet d'allouer une place sur cette grille qui est utilisé lors de l'ajout d'un agent dans la classe **MultiAgentSystem**. Cette classe est le *model*. Elle contient une liste d'**Agent** ainsi que l'**Environment** dans lequel les agents se meuvent et la méthode **run()** qui à chaque tour donne la parole à chaque agents de façon équitable.

### 3 Compilation + fonctionnement

#### Compilation

Se mettre dans le dossier *src* → `javac core/Simulation.java`

#### Execution

Ne pas oublier de compiler le dossier *src*

`java core.Simulation <taille> <nb agent> <nb tour> <delai entre chaque tour>`

Si on rentre un nombre de tour = -1 alors c'est infini

#### Exemples

`java core.Simulation 100 50 -1 5`

`java core.Simulation 10 5 100 5`

`java core.Simulation 50 40 -1 5`