

# Rapport rendu 2

## Organisation des robots :

Les robots de types S (spatial), N (neutralisateur) et R (réparateur) sont tous issus d'une superclasse Robot. Cette superclasse possède comme attributs l'attribut **circle** du robot, issu du module **shape**. Les seules méthodes propres à la superclasse sont des méthodes permettant d'obtenir et de modifier ses attributs mais elle possède également une méthode virtuelle permettant d'obtenir l'attribut **type** propre aux sous-classes.

Les sous-classes gèrent les attributs et fonctions spécifiques à chaque type :

- Le type S possédera comme attributs le nombre de robots de chaque type présent dans la simulation, ainsi qu'une fonction **draw** qui permet son affichage sur la fenêtre. Il possède également une fonction **update** qui est appelée par le module **simulation** afin d'incrémenter le compteur de la simulation et pour plus tard permettre d'autres actions. Ce type possède également des méthodes permettant d'obtenir et de modifier ses attributs.
- Le type N possède comme attributs, en plus de ceux hérités, son angle de direction de mouvement ainsi que des attributs permettant de gérer une éventuelle panne. Cette sous-classe possède comme méthode une fonction **draw** pour l'afficher. Ce type possède également des méthodes permettant d'obtenir et de modifier ses attributs.
- Le type R ne possède pas d'attributs spécifique à sa classe, son fonctionnement plus simple le permettant. Il possède comme méthode une fonction **draw** pour l'afficher. Ce type possède également des méthodes permettant d'obtenir et de modifier ses attributs.

## Structurations diverses du Modèle :

En plus des classes robots, une classe **particule** a été créée selon le même principe. Cette classe possède comme seul attribut le **square** issu du module **shape**. Les méthodes de cette classe sont donc de pouvoir accéder à cet attribut, de le modifier et de l'afficher avec sa méthode **draw**. La classe **particule** possède également une méthode **separate** dont l'appel permet de séparer une instance de cette classe en 4 nouvelles instances.

Pour permettre l'interaction entre ces différentes classes, le module **simulation** initialise toutes les instances de **robot** et de **particule** lors de la lecture d'un fichier texte et va garder la trace de ces objets grâce à deux vectors : l'un contient des **unique\_ptr** sur des instances de la superclasse **robot**, le deuxième contient des **unique\_ptr** sur des instances de **particule**.

Cela permet donc au module **simulation** d'afficher les robots sur la fenêtre dès qu'il en reçoit l'instruction du module **gui**.

## Types dans shape

Le module **shape** contient 3 structures : **s\_2d**, **square** et **circle**.

La première permet de représenter un vecteur de 2 dimensions, utilisé notamment pour les coordonnées de la simulation. Elle possède des surcharges d'opérateurs permettant d'additionner et soustraire 2 vecteurs entre eux.

La deuxième structure, **square**, possède un attribut **center** sous la forme **s\_2d** ainsi qu'un attribut **size** correspondant à la taille de son côté.

La troisième structure **circle** possède comme le square un attribut **center** sous la forme **s\_2d** ainsi qu'un attribut **radius** correspondant à son rayon.