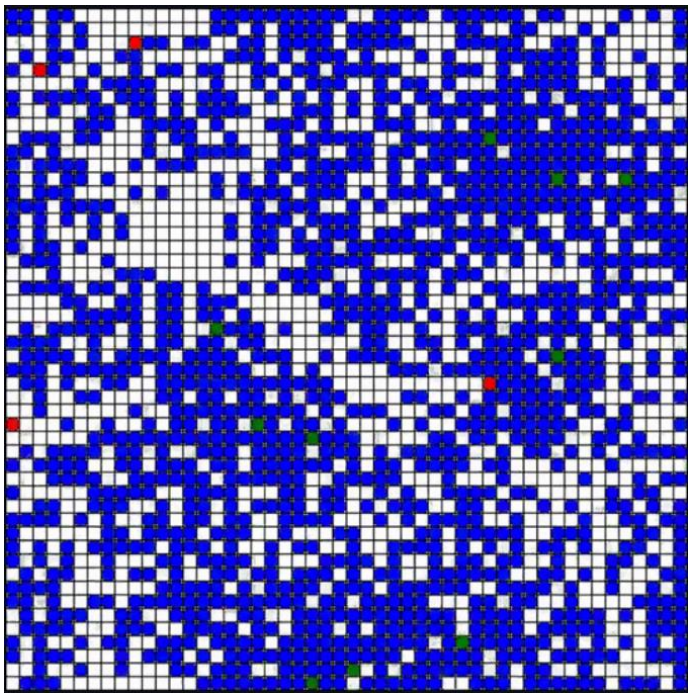
Master Informatique et Modélisation Des Systèmes Complexes (IMSC)

Département de Mathématique - Informatique

Compte rendu TP2 : S.M.A Travail sur les comportements



Réalisé par :

- HAYAR HAMZA

Responsable :

- Mr. GADI TAOUFIK

Année universitaire : 2021/2022

Table des matières

Tableau de figures

TP2

I. Introduction

L'objectif de ce TP était de réaliser un système multi-agents de type proies-prédateurs avec des requins et des poissons. Pour réaliser ce système nous avons principalement utilisé le même Framework que nous avons utilisé dans le TP1 JADE.

En plus de celles-ci, nous avons créé deux classes ainsi qu'un Starter qui charge au démarrage du projet :

- Une classe Fish définissant le comportement des poissons

- Une classe Shark définissant le comportement des requins

Comportement des poissons

Tout comme pour les particules, la méthode getNewPos() des poissons détermine leur destination et choisit donc une case libre aléatoire dans le voisinage du poisson. Une poisson peut aussi se reproduire grâce à la méthode breed() lorsque sa période de gestation est atteinte. Pour ce faire, s'il est en âge de procréer après un déplacement, il créera un autre poisson à son ancienne position. S'il n'est pas capable de se déplacer, il tentera alors de se reproduire en se déplaçant au tour suivant. Finalement, un poisson peut aussi mourir (s'il se fait manger par un requin) via la méthode die().

Comportement des requins

Le requin se reproduit d'une manière similaire au poisson. S'il ne trouve pas de poisson dans son voisinage direct, le requin décidera de son déplacement de la même manière que le poisson. Sinon, il se dirigera vers un poisson aléatoire dans son voisinage et se déplacera à sa place. La méthode eat() permet à un requin de manger. Ils possèdent un compteur qui diminue progressivement avant de tomber à zéro, auquel cas l'agent mourra de faim. Si un requin mange, il fait le plein de nourriture et le compteur reprend sa valeur initiale. La

méthode starve() diminue ce compteur si le requin n'a pas mangé et peut le tuer le cas échéant.

Ce sont les méthodes Action() dans un CyclicBehaviour() des poissons et des requins qui appellent leurs différentes actions

Dans ce TP on n'a pas besoin d'un agent broker paracerque il y'a pas des cas dans lequel on fait des traitements, c'est suffisant des travailles juste avec ces agent Poisson (Fish) et Requin (Shark).

Architecture générale

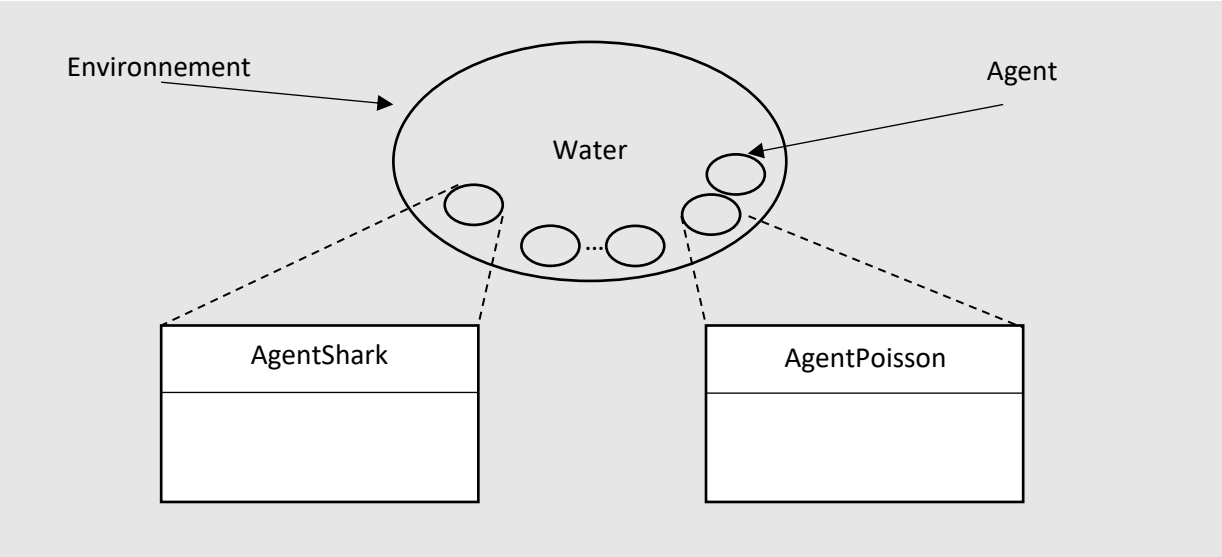


Figure 1 Architecture de système

II. Conception

Comme le premier TP, on va suivre la méthodologie MaSE pour faire la conception de ce système, d'après l'architecture on remarque que notre système se compose de deux différent agent, on pourra ajouter autre agent charge à l'affichage dans une interface graphique.

Cela pourquoi on fait la conception, alors il nous aide de savoir tous les agents qu'on a besoin et ses rôles aussi.

1 Phase 1 (Analyse)

➢ Les Objectifs de système :

- Un nombre des poissons et des requins qui se déplacent dans espace 2D
- Chaque poissant et requin possèdent un coordonnées (x,y) distinct
- Pour les poissons :
    - Dans chaque instant chaque poisson cherche un nouvel emplacement aléatoirement parmi ses voisinages pour déplacer
    - Un poisson peut aussi se reproduire lorsque sa période de gestation est atteinte
    - Un poisson peut aussi mourir (s'il se fait manger par un requin)
- Pour les requins :
    - Le requin se reproduit d'une manière similaire au poisson.
    - A chaque instant s'il ne trouve pas de poisson dans son voisinage direct, le requin décidera de son déplacement de la même manière que le poisson
    - Un requin mourra de faim si un compteur d'endurance tombera à zéro
- Afficher ce système dans une interface graphique

➢ Diagramme de Cas d'utilisation

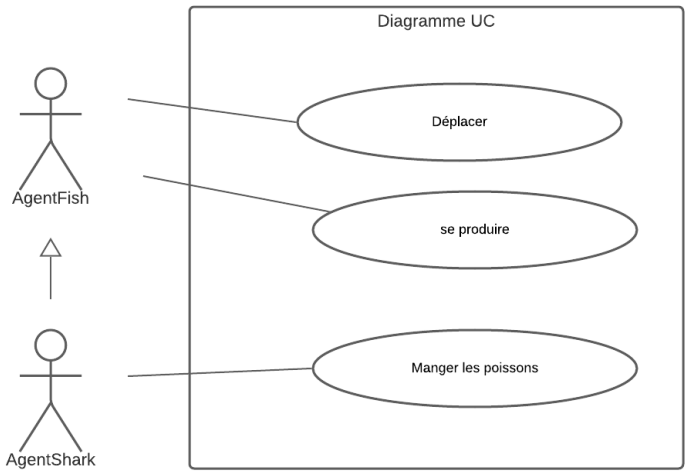


Figure 2 Diagramme de cas d'utilisation

On remarque que l'agent requin (Shark) même que poisson (Fish) il peut déplacer et aussi se produire alors on peut met un héritage entre eux.

➢ Diagramme de séquence

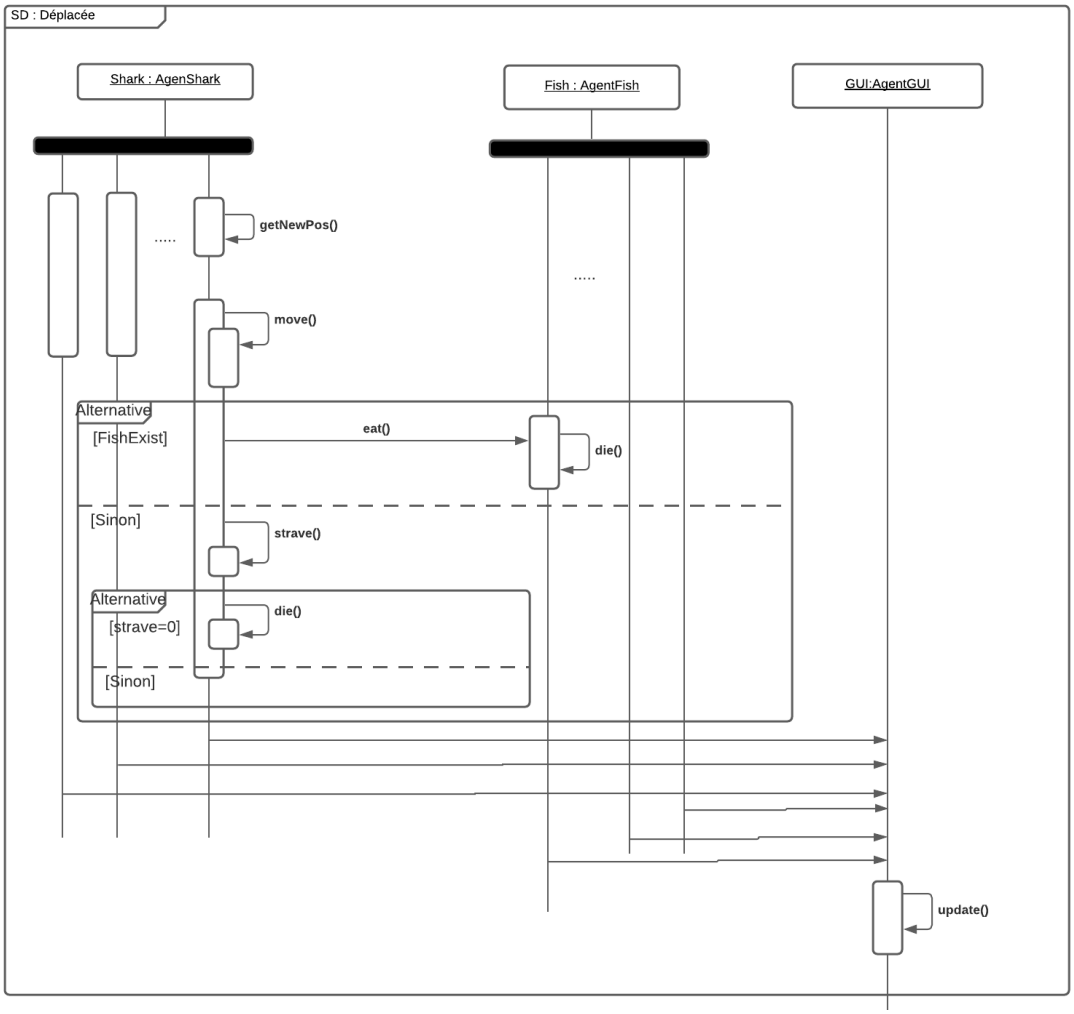


Figure 3 Diagramme de séquence - Déplacer

Puisqu'il y a plusieurs des poissons et des requins on à utiliser juste un seul line de vie d'un agent poisson et d'un requin.

Dans le cas où l'agent requin mange un poisson, l'agent qui responsable a ce poisson détruit lui-même.

Dans ce diagramme tous les agents à la fin de la line ils ont envoyé leurs nouvelles coordonnées au agentGUI.

➢ Diagramme d'activité

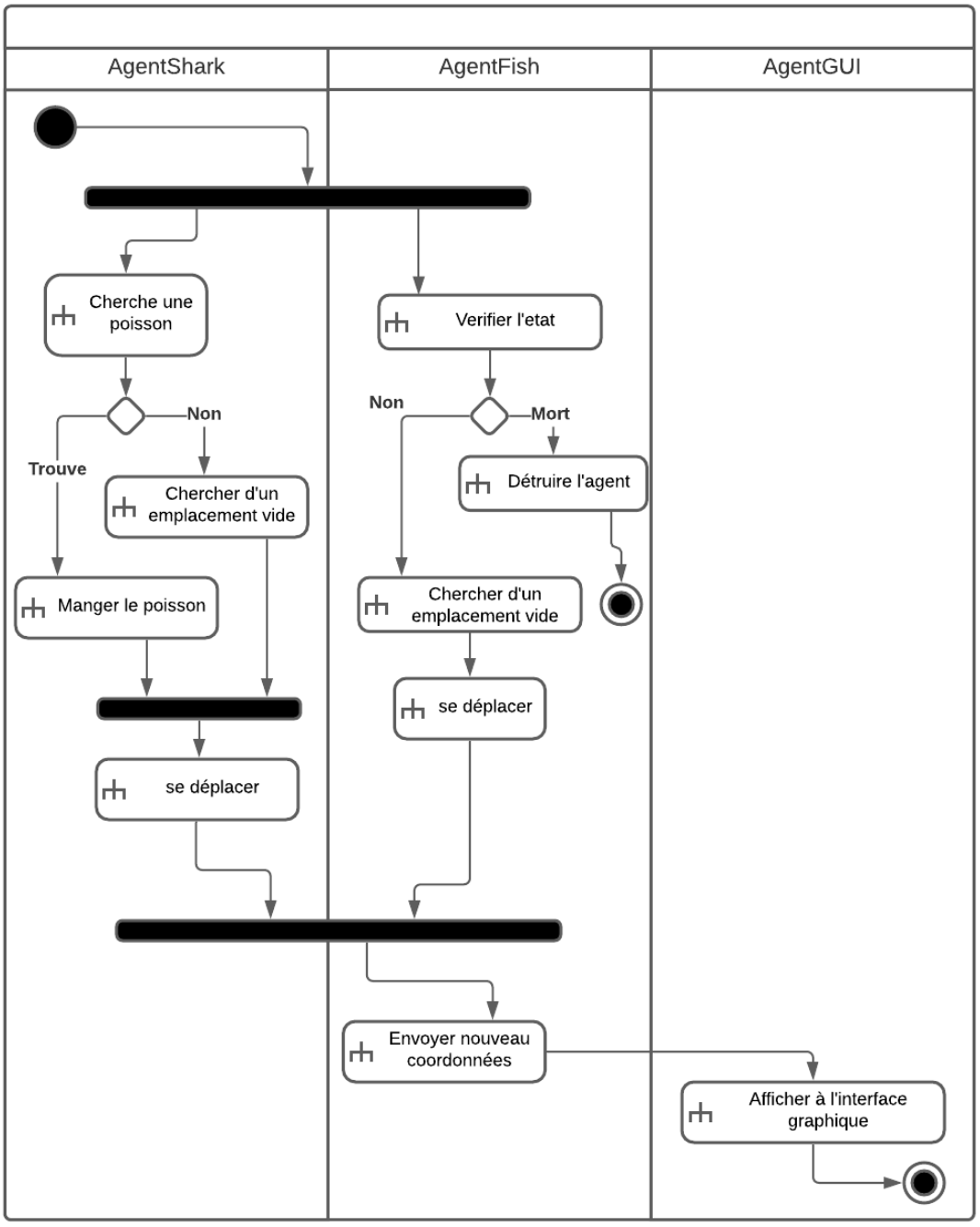


Figure 4 Diagramme d'activité - déplacer

Dans le diagramme d'activité de déplacement on remarque que l'agent poisson vérifier au début son état s'il est mort alors l'agent détruit lui-même, pour l'agent requin on remarque qu'il cherche premièrement un poisson au voisinage s'il ne trouvait pas alors il cherche un emplacement vide pour se déplacer.

➢ Rôles

Agent	Fish1..N	Shark1..N	GUI
Rôle	Fish	Shark	Interface_graphique

Tableau 1 Tableau des rôles

2 Phase 2 (Conception)

➢ Représentation d'agent Shark

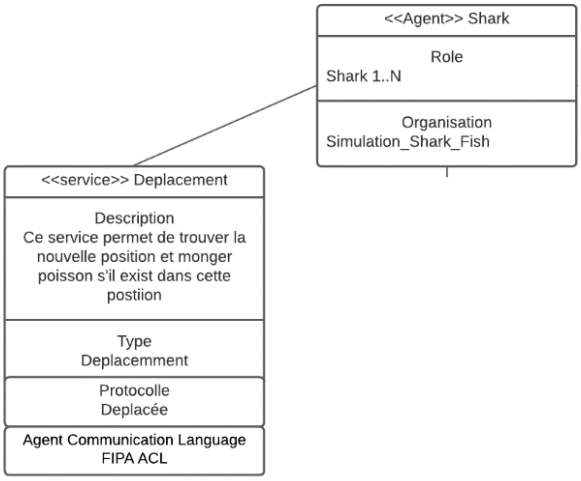


Figure 5 Représentation d'agent Shark

➢ Représentation d'agent Fish

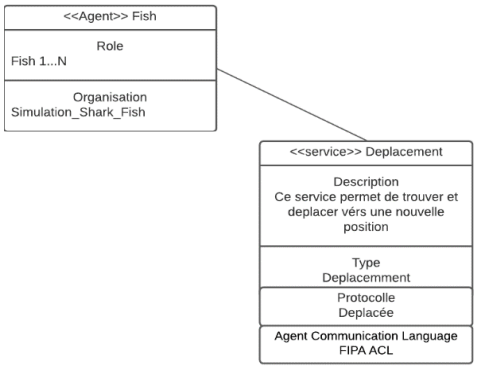


Figure 6 Représentation d'agent Fish

- Représentation d'agent GUI

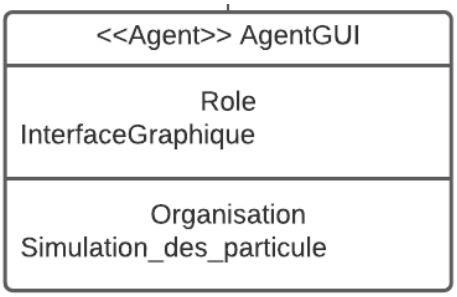


Figure 7 Représentation d'agent GUI

- Communication entre les Agents

Entre Fish et Shark

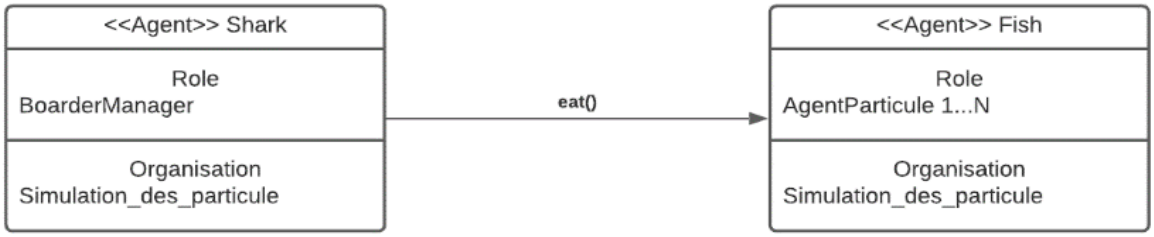


Figure 8 Communication entre les Agents Shark et Fish

La communication entre AgentFish et AgentShark est d'une seule sen, l'agent requin (Shark) indiquer au poisson qu'il l'a mangé, et par la suit agent poisson (Fish) vérifier son état et il détruit lui-même.

Entre Fish ou Shark et GUI

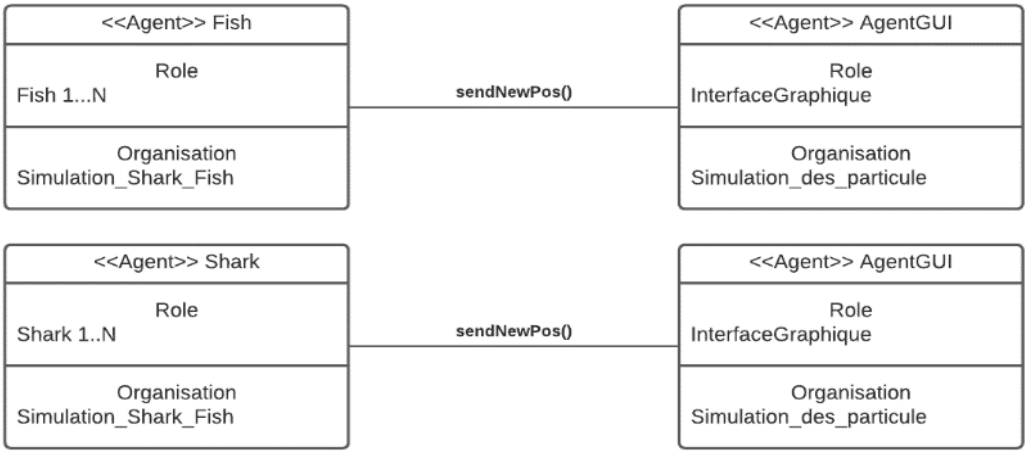


Figure 9 Communication entre les Agents Shark/Fish et GUI

La communication entre les deux agent Shark/Fish et GUI aussi est simple, il sert juste pour envoyer les nouvelles coordonnées (X, Y) des agent (Fish et Shark) à chaque instant (frame), et GUI reste toujours dans l'attente des messages des agent Fish et Shark

- Diagramme de class

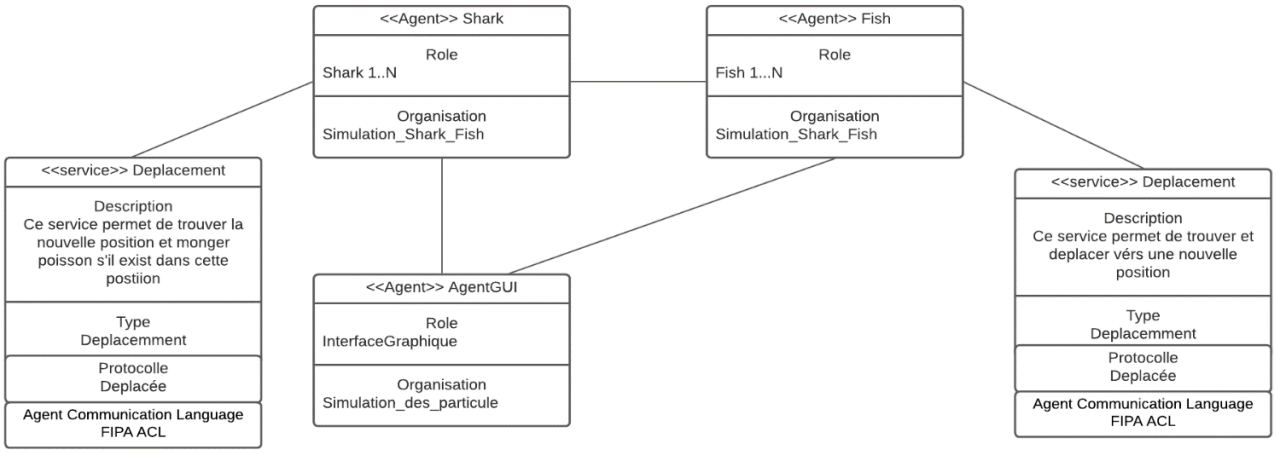


Figure 10 Diagramme de Class

III. Implémentation

1 GUI

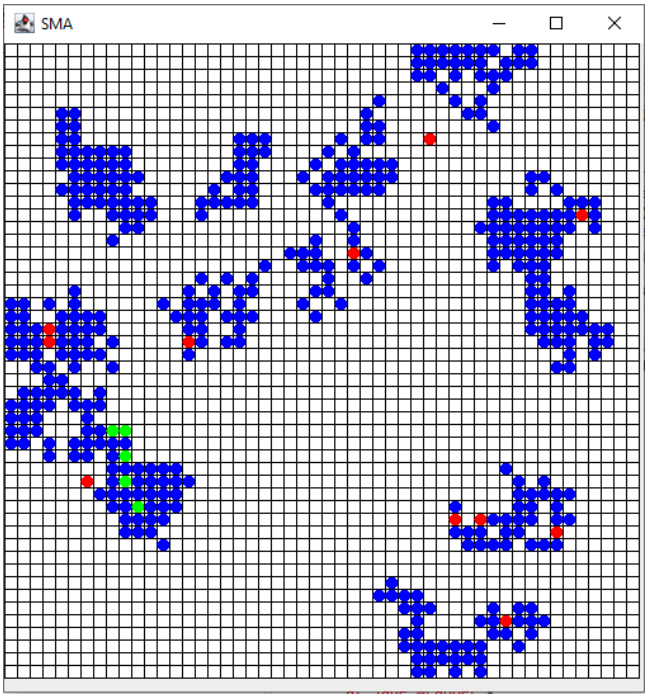


Figure 11 GUI - progresse 1

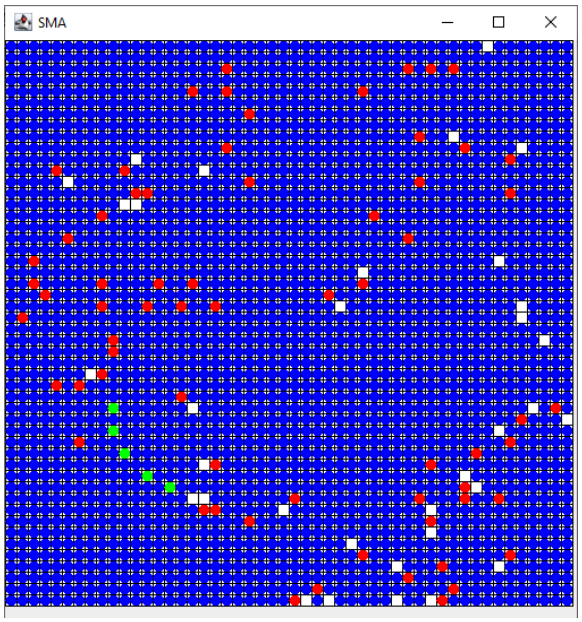


Figure 12 GUI - progresse 2

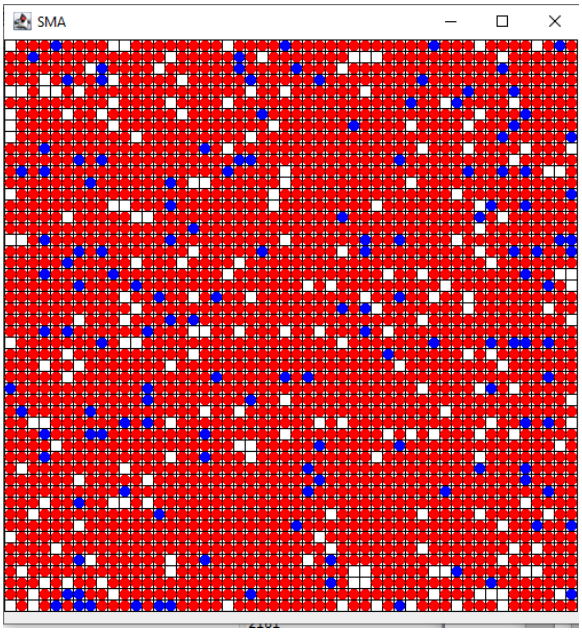


Figure 13 GUI - progresse 3

Les poissons sont présentés par le couleur bleu et les requins avec couleur rouge

2 Les Messages enter les agents (outil sniffer)

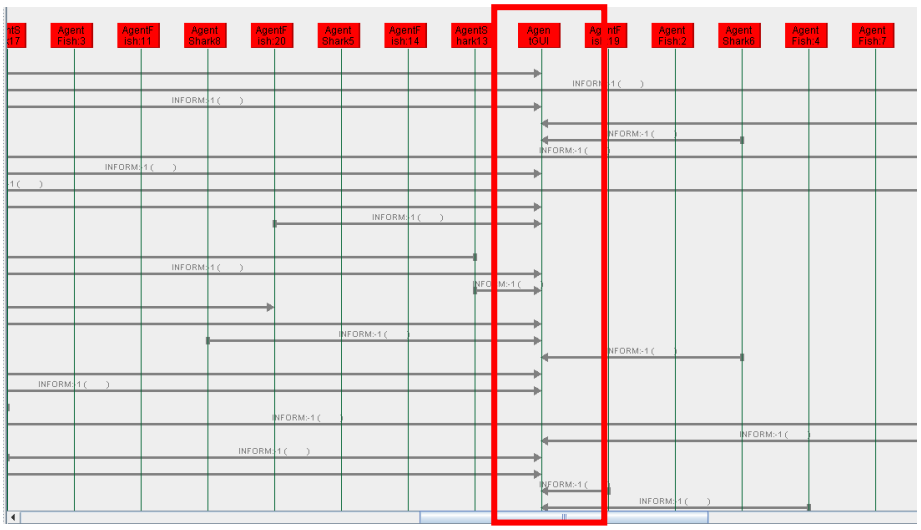


Figure 14 Sniffer 1

Dans cette figure on peut remarquer bien le message envoyé soit par les agents Shark ou les agents Fish au AgnetGUI, pour mettre à jour les coordonnées, et par la suite afficher à l'interface graphique

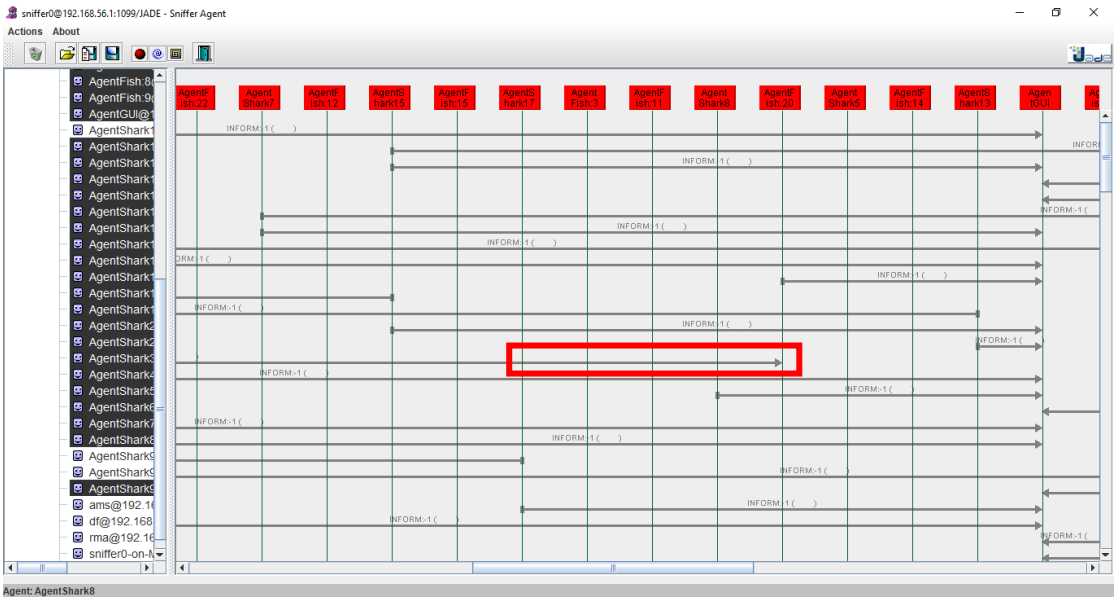


Figure 15 Sniffer 2

Dans cette figure on voit que AgentFish20 a reçu un message d'un requin, évidement ce message informe que ce requin a mangé ce poisson, alors l'agent va détruit