

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche  
Scientifique

**Université d'Ibn Khaldoun – Tiaret**

Faculté des Mathématiques et de l'Informatique

**Département d'Informatique**

Mémoire

Présenté pour obtenir le diplôme de

Master en Informatique

**Spécialité** : génie informatique

**Option** : Systèmes d'information

**Thème**

Mise en place d'un système de communication  
multi-agents

**Réalisé par :**

❖ M<sup>r</sup>ASNOUNE Mohamed El Amine

**Dirigé par :**

❖ M<sup>r</sup> ALEM Abdelkader

**Année Universitaire 2015-2016**

## *Dédicace*

*Je dédis ce modeste travail*

*A mes chers parents*

*Pour leur soutien, leur patience, leur sacrifice*

*et leur amour, vous méritez tout éloge,*

*vous qui avez fait de nous ce que je suis maintenant.*

*J' espère être l'image que vous êtes fait de*

*moi, que dieu vous garde et vous bénisse.*

*Je dédis aussi ce travail à mes sœurs*

*pour leur affection et leur encouragement qui ont toujours*

*été pour moi des plus précieux.*

*Que ce travail soit pour vous le gage de notre profond amour*

*A tout mes amis*

*A tous ceux qui m'a aidé.*

*A tous ceux que nous aimons, je dédis ce travail ...*

## *Remerciements*

*Au terme de ce travail, je tiens à exprimer mes profonde gratitude et mes sincères remerciements à mon encadreur monsieur ALGM Abd El Kader pour tout le temps qu'ils m'a consacré, leur directives précieuses, et pour la qualité de leur suivi durant toute la période de la réalisation de ce PFE.*

*Je remercie également les membres du jury pour avoir accepté d'évaluer ce travail.*

## *Résumé*

Le travail présenté dans ce mémoire consiste en la mise en place d'un système de communication multi agent. L'application porte sur un marché électronique, de ce fait, nous définissons un modèle en décrivant le comportement de deux types d'agents : les acheteurs et les vendeurs, aussi l'opérateur humain intervient pour la configuration de son agent et un administrateur gère l'ensemble du système.

Notre modèle est constitué d'un ensemble de propriétés incluant le nom d'agent, son produit, la stratégie de choix du vendeur par l'acheteur en cas de multiplicité de vendeurs disponibles et la stratégie de calcul du prix par le vendeur.

Nous utilisons la plate forme JADE pour la réalisation de notre système.

### **Mots clés**

Agents, systèmes multi agents, communication, commerce électronique, critères de choix, variation de prix.

# Sommaire

<b>Dédicace</b> .....	<b>2</b>
<b>Remerciements</b> .....	<b>3</b>
<b>Résumé</b> .....	<b>4</b>
<b>Introduction Générale</b> .....	<b>9</b>
<b>Chapitre I : Les systèmes Multi Agents</b> .....	<b>11</b>
<b>Introduction</b> .....	<b>12</b>
<b>1. L'intelligence Artificielle distribuée</b> .....	<b>12</b>
1.1. L'intelligence Artificielle (IA) .....	12
1.2. L'intelligence artificielle distribuée (IAD) .....	12
1.3. Axes fondamentaux dans la recherche en IAD .....	13
<b>2. Notion d'agent</b> .....	<b>13</b>
2.1. Définitions.....	13
2.2. Les caractéristiques d'un agent .....	14
2.3. La différence entre un objet et un agent.....	15
2.4. La classification d'agent .....	16
<b>3. Système Multi Agents (SMA)</b> .....	<b>18</b>
3.1. Définitions.....	18
3.2. L'environnement .....	20
3.3. Architecture du SMA.....	20
3.4. L'interaction entre les agents .....	21
3.5. Les caractéristiques d'un SMA.....	24
3.6. Les langages de communication .....	25
3.7. Les rôles de Système Multi Agent.....	25
3.8. Les avantages des SMA .....	26
3.9. Domaines d'applications des systèmes multi agents .....	27
<b>Conclusion</b> .....	<b>29</b>
<b>Chapitre II : La communication entre les agents</b> .....	<b>30</b>
<b>Introduction</b> .....	<b>32</b>
<b>1. Les Actes de Langage</b> .....	<b>32</b>

---

<b>2. Langages de Communication entre Agents</b> .....	<b>33</b>
<b>2.1. KQML</b> .....	<b>34</b>
2.1.1. Syntaxe.....	34
2.1.2. Sémantique.....	35
2.1.3. Implémentations.....	35
2.1.4. Problèmes de KQML.....	36
<b>2.2. FIPA ACL</b> .....	<b>36</b>
2.2.1. Syntaxe.....	37
2.2.2. Représentation XML.....	37
2.2.3. Sémantique.....	38
2.2.4. Implémentations.....	38
<b>2.3. Problèmes des ACLs</b> .....	<b>39</b>
<b>2.4. Langages sociaux et engagements</b> .....	<b>39</b>
<b>3. Contenu des messages ACL</b> .....	<b>39</b>
3.1. KQML & KIF.....	40
3.2. FIPA ACL & SL.....	40
3.3. Ontologies.....	41
<b>4. ACL &amp; Facilitateurs</b> .....	<b>41</b>
<b>5. Conversations entre Agents</b> .....	<b>42</b>
5.1. Règles de Conversation.....	42
5.2. Protocoles FIPA.....	42
5.3. Conduites de Conversation.....	43
<b>Conclusion</b> .....	<b>44</b>
<b>Chapitre III : La modélisation</b> .....	<b>45</b>
<b>E-Commerce</b> .....	<b>46</b>
<b>1. Introduction</b> .....	<b>46</b>
1. Définition du commerce électronique.....	46
2. Objet et formes du commerce électronique.....	47
3. Chiffres clés et tendances du commerce électronique.....	47
<b>La modélisation</b> .....	<b>49</b>
1. Diagramme de cas d'utilisation :.....	49
2. Diagrammes de séquences.....	50
a) L'ajout de client.....	50
b) Modification des produits.....	51

c)	Supprimer un produit .....	52
d)	La création de l'agent vendeur .....	53
e)	La création de l'agent acheteur .....	54
f)	L'appelle d'offre(CFP) .....	55
<b>3.</b>	<b>Diagramme d'activité.....</b>	<b>55</b>
a)	Ajout d'un produit .....	56
b)	Modifier un produit .....	57
c)	Supprimer un produit .....	58
d)	La création d'un agent acheteur .....	59
e)	La création d'un agent vendeur .....	60
	<b>Chapitre IV : L'implémentation.....</b>	<b>61</b>
	<b>JADE .....</b>	<b>62</b>
1.	Définition .....	62
1.1.	Conteneurs.....	63
1.2.	Outils utiles au débogage.....	64
1.3.	Architecture de la plateforme Jade.....	65
1.5.	Agents donnant accès aux services.....	66
	<b>Conclusion générale.....</b>	<b>75</b>
	<b>Référence .....</b>	<b>76</b>

## Liste des figures

<b>Figure 1 : Agent réactif</b> .....	17
<b>Figure 2 Agent cognitif</b> .....	18
<b>Figure 3 : Image globale de SMA</b> .....	19
<b>Figure 4 : Communication par envoi de messages</b> .....	23
<b>Figure 5 : Communication par partage d'informations</b> .....	24
<b>Figure 6 : Communication naturelle entre deux individus</b> .....	32
<b>Figure 7 : Echange d'informations entre agents utilisant KQML</b> .....	34
<b>Figure 8 : Syntaxe du performatif tell de KQML</b> .....	34
<b>Figure 9 : Echange d'informations entre agents utilisant FIPA ACL</b> .....	37
<b>Figure 10 : Spécification du protocole 'FIPA Request'</b> .....	43
<b>Figure 11 : Diagramme de CAS d'utilisation</b> .....	49
<b>Figure 12 : diagramme de séquence -ajout client</b> .....	50
<b>Figure 13 : diagramme de séquence -modifier client</b> .....	51
<b>Figure 14 : diagramme de séquence -suppression client</b> .....	52
<b>Figure 15 : diagramme de séquence - création agent vendeur</b> .....	53
<b>Figure 16 : diagramme de séquence - création agent acheteur</b> .....	54
<b>Figure 17: diagramme de séquence - appelle d'offre</b> .....	55
<b>Figure 18 : diagramme d'activité - ajout client</b> .....	56
<b>Figure 19: diagramme d'activité - modifier client</b> .....	57
<b>Figure 20 : diagramme d'activité -suppression client</b> .....	58
<b>Figure 21 : diagramme d'activité -création agent acheteur</b> .....	59
<b>Figure 22 : diagramme d'activité -création agent vendeur</b> .....	60
<b>Figure 23 : Conteneurs</b> .....	64
<b>Figure 24 : Architecture de la plateforme Jade</b> .....	65
<b>Figure 25 WampServer Logo</b> .....	67
<b>Figure 26 JDBC Connecter</b> .....	67
<b>Figure 27: fenêtre principale</b> .....	68
<b>Figure 28 : Fenêtre SNIFER</b> .....	68
<b>Figure 29 Fenêtre d'accueil</b> .....	68
<b>Figure 30 : fenêtre administrateur</b> .....	69
<b>Figure 31 : fenêtres création des agents</b> .....	70
<b>Figure 32: fenêtre d'affichage de type voiture</b> .....	71
<b>Figure 33:fenêtre d'affichage de type informatique</b> .....	71
<b>Figure 34 : fenêtre de actualisation des agents</b> .....	72
<b>Figure 35 : Fenêtre communication entre les agents</b> .....	72
<b>Figure 36: sniffer de confirmation</b> .....	73
<b>Figure 37: sniffer de refus</b> .....	74

## Introduction Générale

**L**es systèmes multi agents ont pris une place de plus en plus importante en informatique, que ce soit dans le domaine de l'intelligence artificielle, dans ceux des systèmes distribués, de la robotique ou même dans la vie artificielle, en introduisant la problématique de l'intelligence collective et de l'émergence de structures par interactions [1].

Une caractéristique intéressante des systèmes multi agents est la communication, de ce fait, nous essayons de mettre en place une application dont la communication est la forme d'interaction de base. Un exemple typique de ces applications est le commerce électronique.

Les agents intelligents peuvent être utilisés en commerce électronique de plusieurs façons : ils peuvent filtrer de l'information concernant les produits pour identifier les besoins de l'utilisateur, apprendre son modèle de raisonnement ou encore se comporter comme de vrais négociateurs et mener des discussions d'achat et de vente.

Dans notre système, nous avons deux types d'agents : acheteurs et vendeurs. L'acheteur peut communiquer avec le vendeur par le biais de deux critères qui sont le temps et le prix.

Le système est dynamique en ce sens que de nouveaux agents peuvent être lancés pendant l'exécution du SMA<sup>1</sup>.

L'opérateur humain intervient en configurant son agent et un administrateur gère la base de données des produits.

Un autre point important dans les systèmes multi agents est l'outil d'implémentation pour lequel plusieurs plates formes existent. Notre choix a porté sur la plateforme JADE vu qu'elle est largement utilisée dans les SMA.

---

<sup>1</sup> Système multi agents.

Ce document est organisé en quatre chapitres :

Le premier chapitre introduit l'intelligence artificielle distribuée ainsi que les notions d'agent et de systèmes multi agents.

Le deuxième chapitre met l'accent sur la communication, les actes de langage et les formes de messages sont détaillés dans ce chapitre.

Le troisième chapitre est consacré au commerce électronique et à la modélisation du système. Nous avons utilisé la méthode UML comme méthode de conception vue la similarité entre la notion d'objet et celle d'agent (objet autonome).

Dans le quatrième chapitre, nous présentons notre implémentation, nous introduisons les choix techniques à savoir la plateforme JADE et l'outil WAMP SERVER pour la gestion de la base de données. L'interface graphique ainsi que deux exemples de déroulement sont présentés dans ce chapitre.

Enfin, nous concluons ce mémoire par une récapitulation de notre travail et quelques perspectives complémentaires.

# Chapitre I<sub>Les</sub>

---

Systemes Multi Agents

## **Introduction**

Les systèmes multi-agents constituent aujourd'hui une nouvelle technologie pour la conception et le contrôle de systèmes complexes. Ils sont composés d'entités logicielles ou matérielles autonomes appelées agents, ces systèmes possèdent généralement plusieurs caractéristiques intéressantes, comme le parallélisme, la robustesse et l'extensibilité.

L'approche multi-agents repose sur plusieurs théories et concepts qui trouvent leurs origines dans plusieurs disciplines telles que la sociologie, la psychologie, les systèmes répartis et le génie logiciel.

Dans ce chapitre, nous verrons les concepts fondamentaux de l'approche agent. Après une petite introduction sur l'intelligence artificielle distribuée, nous mettons l'accent sur l'évolution de l'aspect individuel (le comportement d'un agent seul) vers l'aspect collectif (son comportement dans une société d'agents). Ensuite, nous donnons quelques plates formes de développement des systèmes multi-agents. Nous donnons un état d'avancement des travaux sur ce domaine avant de conclure sur les possibilités et les voies de recherche qu'offre ce domaine.

## **1. L'intelligence Artificielle distribuée**

### **1.1. L'intelligence Artificielle (IA)**

L'Intelligence Artificielle (IA) est reconnue comme étant une discipline informatique qui a pour objectif de modéliser et de simuler des comportements humains dits intelligents, tels que la perception, la prise de décision, l'apprentissage, etc.

Elle s'attache à la construction de problèmes informatiques, capables d'exécuter des tâches complexes, en s'appuyant sur une centralisation et sur concentration, de l'intelligence au sein d'un système unique.

Au fil du temps, l'Intelligence Artificielle (IA) classique a montré des limites à résoudre des problèmes complexes. Dans le but de combler ces limites, les chercheurs ont senti le besoin de passer du comportement individuel aux comportements collectifs et la nécessité de distribuer l'intelligence sur plusieurs entités.

### **1.2. L'intelligence artificielle distribuée (IAD)**

L'intelligence artificielle distribuée (IAD) est née, au début des années 80, de la volonté à remédier à l'insuffisance et d'enrichir l'approche classique d'IA en proposant la distribution de l'expertise sur un groupe d'agents, non soumis à un contrôle centralisé, devant être capables de travailler et d'agir dans un environnement commun et de résoudre les conflits éventuels. L'IAD peut alors être définie comme étant la branche de l'IA qui s'intéresse à la modélisation de comportements intelligents par la coopération entre un ensemble d'agents, d'où la réalisation

des systèmes dits « multi agents » dont les caractéristiques sont : La coopération, la coordination et la communication. [1]

### 1.3. Axes fondamentaux dans la recherche en IAD

Après avoir défini l'IAD, nous présentons les trois axes fondamentaux dans la recherche en IAD :

- Les systèmes multi-agents (SMA) : Il s'agit de faire coopérer un ensemble d'agents dotés d'un comportement intelligent et de coordonner leurs buts et leurs plans d'actions pour la résolution d'un problème.
- La résolution distribuée des problèmes (RDP) : Elle s'intéresse à la manière de diviser un problème particulier sur un ensemble d'entités distribuées et coopérantes. Elle s'intéresse aussi à la manière de partager la connaissance du problème et d'en obtenir la solution.
- L'intelligence Artificielle Parallèle (IAP) : Elle concerne le développement de langages et d'algorithmes parallèles qui visent l'amélioration des performances des systèmes d'intelligence artificielle sans toutefois s'intéresser à la nature du raisonnement ou au comportement intelligent d'un groupe d'agents. [49]

## 2. Notion d'agent

La notion d'agent est utilisée dans beaucoup de domaines : sociologie, biologie, psychologie cognitive, psychologie sociale, informatique, etc. Cependant, il n'y a pas une définition acceptée en unanimité. Nous citons dans ce qui suit quelques définitions sur la notion d'agent :

### 2.1. Définitions

On appelle agent un système mécanique, biologique ou logiciel qui interagit avec son environnement. [2]

Un agent est une entité physique ou virtuelle [3] :

- qui est capable d'agir dans un environnement,
- qui peut communiquer directement avec d'autres agents,
- qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser),
- qui possède des ressources propres,
- qui est capable de percevoir (mais de manière limitée) son environnement,
- qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune),
- qui possède des compétences et offre des services,

- qui peut éventuellement se «reproduire»,
- qui a un comportement qui tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.

## 2.2. Les caractéristiques d'un agent

Nous pouvons distinguer les caractéristiques suivantes d'un agent [1]:

### • **Autonomie**

L'autonomie d'un agent peut s'exprimer de la manière suivante :

- Un agent a sa vie propre, indépendamment de l'existence des autres agents (mais cela n'empêche pas des liens de dépendances entre agents).
- Il est capable de survivre dans des environnements dynamiques sans l'aide d'un contrôle externe.
- Il prend des décisions motivées par son état interne sans intervention extérieure.

### • **Situé**

- Un agent est situé dans son environnement (physique ou virtuel).
- Il a une représentation de son environnement.

### • **Réactif**

Un agent est capable de percevoir son environnement et de réagir aux changements qui interviennent par ses actions.

- Un agent ne peut pas être dépourvu d'un environnement.
- L'environnement définit les conditions d'existence d'un agent.

### • **Social**

- Un agent est capable d'interagir et de communiquer avec les autres agents (par des langages de communication) pour accomplir ses buts.
- Il est capable de coopérer pour résoudre des problèmes ou effectuer des tâches,
- d'acquérir des ressources qu'il ne possède pas,
- de profiter des compétences des autres.

- **Proactif**

Il ne fait pas que réagir à son environnement, mais il est capable de lui-même de produire des actions motivées par des buts.

- **Intentionnalité**

Un agent intentionnel est un agent guidé par ses buts. Une intention est la déclaration explicite des buts et des moyens d'y parvenir. Elle exprime la volonté d'un agent d'atteindre un but ou d'effectuer une action.

- **Rationalité**

Un agent rationnel est un agent qui effectue les bonnes actions aux bons moments. Les agents rationnels disposent des critères d'évaluation de leurs actions, et sélectionnent selon ces critères les meilleures actions qui leur permettent d'atteindre le but. De tels agents sont capables de justifier leurs décisions.

- **Adaptabilité**

Un agent adaptatif est un agent de haut niveau de flexibilité, il est capable de contrôler ses aptitudes (communicationnelles, comportementales, etc.) selon l'agent avec qui il interagit. Il doit constamment modifier le plan qu'il poursuit pour atteindre un but, face à un environnement perpétuellement changeant.

- **Engagement**

La notion d'engagement est l'une des qualités essentielles des agents coopératifs. Un agent coopératif planifie ses actions par coordination et négociation avec les autres agents. En construisant un plan pour atteindre un but, l'agent se donne les moyens d'y parvenir et donc s'engage à accomplir les actions qui satisfont ce but : l'agent croit qu'il est en mesure d'exécuter tout le plan qu'il a élaboré, ce qui le conduit à agir en conséquence.

- **Intelligence**

En conclusion, on appelle un agent intelligent un agent cognitif, rationnel, adaptatif et intentionnel.

### **2.3. La différence entre un objet et un agent**

-Un agent a une autonomie de contrôle car il décide de son comportement en fonction de son état, croyances, connaissances, perceptions de l'environnement, requêtes des autres, par contre dans l'approche orientée objet on ne dispose pas d'autonomie car l'objet est invoqué par un appel de méthode qu'il ne peut refuser (pas de réactivité).

- Un agent est une entité sociale: composante importante dans des interactions et des organisations complexes, mais l'objet à un faible niveau de socialité: interaction simple et rigide (pas d'évolution dans le temps).
- Dans l'approche orientée agent, la notion d'environnement est importante et complexe mais dans l'approche orientée objet elle n'existe pas.
- Un agent a la possibilité de répondre par l'affirmative ou le refus à des requêtes provenant des autres agents. Il dispose donc d'une certaine liberté de manœuvre.
- Un objet est une entité passive (ou réactive). Si personne ne demande la valeur d'un attribut ou n'active une méthode de l'objet, alors il ne se passe rien, mais un agent possède, en plus des attributs et méthodes, des processus internes qui fonctionnent même en l'absence de sollicitations externes. Un agent peut donc agir même si personne ne lui demande rien.
- Les agents sont autonomes et responsables de leurs actions alors que les objets n'en sont toujours pas.
- On ne peut prévoir tout le comportement des agents dans les systèmes.
- L'approche orientée objet ne fournit pas un ensemble adéquat de concepts et de mécanismes pour modéliser les systèmes complexes dans lesquels les rapports évoluent dynamiquement.
- Certains chercheurs définissent un agent comme un objet actif ayant une autonomie et un objectif.

## **2.4. La classification d'agents**

### **a) Les agents réactifs**

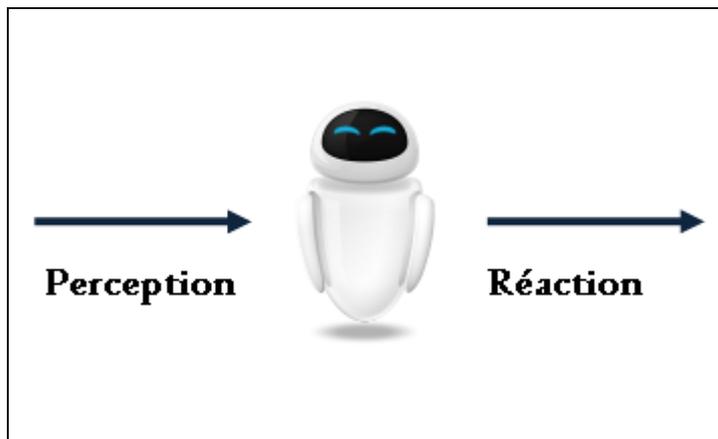
Les agents réactifs sont des entités très simples qui perçoivent l'environnement et sont capables d'agir sur celui-ci. Ils n'ont pas une représentation complète de leur environnement ou des connaissances et ils ne possèdent pas de capacité de raisonnement ni de mécanismes d'envoi de messages et ne sont pas capables de tenir compte de leurs actions passées.

Les agents réactifs sont issus de la modélisation physicaliste du monde : ils sont assimilés à des fonctions de transition.

Ils ont un comportement du type « stimulus – réponse » et utilisent des mécanismes de réaction aux événements.

Un SMA constitué uniquement d'agents réactifs possède généralement un grand nombre d'agents et présente un comportement global intelligent, mais il n'est pas nécessaire que les agents soient "intelligents" individuellement.

Le comportement intelligent devrait émerger de l'interaction entre ces agents réactifs et l'environnement qui les entoure.



*Figure 1 : Agent réactif*

Les architectures réactives ont l'avantage de la simplicité et de l'efficacité de calcul. Pourtant, elles présentent plusieurs limitations, ce qui fait que ces architectures ne peuvent pas être utilisées dans de nombreuses classes d'applications[1][3].

Les limites des agents réactifs viennent du fait qu'ils ont une vision de courte durée sur la résolution du problème, et ils ne peuvent pas toujours choisir la meilleure action à exécuter à un certain moment.

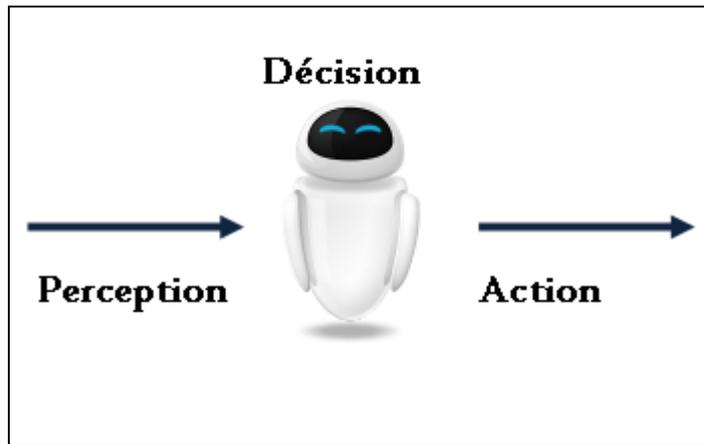
Comme les agents réactifs ne maintiennent pas une représentation de l'environnement, ils ne peuvent pas avoir des buts.

### **b) Les agents cognitifs**

Les agents cognitifs sont issus de la modélisation mentaliste du monde et ils sont assimilés à des systèmes experts.

Les agents cognitifs sont plus évolués. Ils ont des buts explicites, une représentation symbolique des connaissances, une représentation globale de leur environnement et des autres agents avec lesquels ils communiquent, Ils sont intelligents c'est-à-dire qu'ils effectuent un certain raisonnement pour choisir leurs actions, et sont capables de planifier leurs comportements, communiquer par envoi de messages, négocier, etc. Ils savent tenir compte de leur passé et s'organiser autour d'un mode social d'organisation.

Les systèmes multi agents constitués uniquement d'agents cognitifs sont constitués d'un nombre d'agents assez faible comprenant une base de connaissance (informations + savoirs faire). Ils réclament des ressources plus importantes que les systèmes d'agents réactifs.



*Figure 2 Agent cognitif*

Le module de décision de tels agents est réalisé via le raisonnement logique.

Les limites de cette approche sont dues à la complexité des algorithmes de manipulation symboliques[1][3].

### **c) Agents hybrides**

Chaque agent hybride est caractérisé par la notion de couches et chaque couche représente soit les agents cognitifs, soit les agents réactifs. Donc l'agent hybride combine entre les deux comportements (comportement réactif et comportement cognitif).

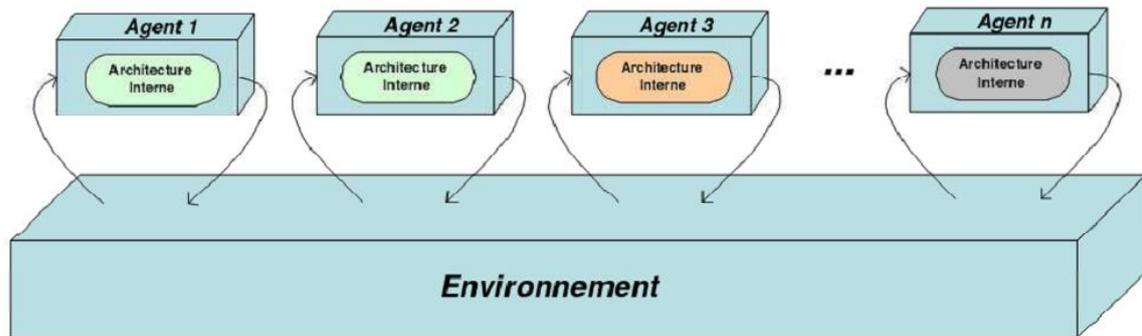
## **3. Système Multi Agents (SMA)**

### **3.1. Définitions**

Plusieurs définitions ont été données pour les SMA, nous citons les suivantes :

- a) Un système multi agents (SMA) comporte plusieurs agents qui interagissent entre eux dans un environnement commun. Certains des agents du SMA sont des agents logiciels, certains d'autres peuvent être des personnes ou même des machines mécaniques. [2]
- b) Un système multi agents est un système composé des éléments suivants [3] :
  - Un environnement E, disposant d'une métrique en général,
  - Un ensemble d'objets O, auxquels on peut associer une position dans E à un moment donné. Ces objets (hormis les agents) sont passifs : les agents peuvent les percevoir, les créer, les détruire et les modifier.
  - Un ensemble A d'agents, lesquels représentent les entités actives du système,

- Un ensemble de relations R qui unissent les objets (et agents) entre eux,
- Un ensemble d'opérateurs Op permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O. [4]



**Figure 3 : Image globale de SMA**

c) SMA = A+E+I+O

L'analyse, le design, et le déploiement d'un SMA comportent quatre aspects fondamentaux :

- Agents: architecture interne des agents.
  - Environnement : le milieu dans lequel évoluent les agents.
  - Interaction: les moyens par lesquels les agents interagissent.
  - Organisation : les moyens utilisés pour structurer les entités du système multi agents.
- Un SMA peut-être :
    - Ouvert : les agents y entrent et en sortent librement (exemple : un café).
    - Fermé : l'ensemble d'agents reste le même (exemple : un match de football).
    - Homogène : tous les agents sont construits sur le même modèle (exemple : une colonie de fourmis).
    - Hétérogène : des agents de modèles différents, de granularités différentes (exemple : l'organisation d'une entreprise). [5]

## 3.2. L'environnement

Dans un système multi agents, on appelle environnement l'espace commun aux agents du système.

L'environnement peut être considéré comme la représentation du monde dans lequel les agents se situent. Il est modifiable par les agents, soit de façon globale, soit en faisant la distinction entre objets passifs (soumis aux actions des agents) et entités actives (les agents) [3].

## 3.3. Architecture du SMA

L'architecture du SMA [4] sera liée à la façon dont seront implémentées les communications.

### a) Organisation centralisée

Dans une conception centralisée, un agent connaît tous les autres agents. Ainsi lorsque l'on a besoin d'une compétence particulière ou d'un agent particulier on s'en réfère à cet agent pour connaître les agents concernés.

L'avantage de la méthode est sa simplicité de mise en œuvre. Cependant, un seul objet gère toutes les adresses, il peut devenir un goulet d'étranglement et grever de façon significative les performances du système. De même, si l'objet gérant le système d'adressage plante, le système plante avec.

### b) Organisation libre (non centralisée)

Les systèmes non centralisés sont composés d'agents plus complexes que précédemment, dotés de capacités de raisonnement élaborées. Généralement, ces agents sont localisés sur des sites physiquement différents.

Cette méthode présente l'avantage d'être plus "distribuée" que la méthode centralisée. En cas de défaillance d'une partie du système, le reste peut continuer à fonctionner. Cependant, la recherche d'un agent donné ou d'une catégorie d'agent ayant des compétences particulières est moins directe. De même, l'attribution d'adresses uniques à chaque agent est moins directe que dans la méthode centralisée.

Aucun agent ne connaît tous les agents. Localement, un agent peut connaître les agents avec qui il est susceptible de traiter, mais personne n'a, à priori, de vision globale.

### 3.4. L'interaction entre les agents

Jacques Ferber donne la définition suivante de l'interaction : “ Une interaction est la mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques. Les interactions sont non seulement la conséquence d'actions effectuées par plusieurs agents en même temps, mais aussi l'élément nécessaire à la constitution d'organisations sociales. ” [3].

Une des principales propriétés de l'agent dans un SMA est celle d'interagir avec les autres agents. Ces interactions sont généralement définies comme toute forme d'action exécutée au sein du système d'agents et qui a pour effet de modifier le comportement d'un autre agent. Elles permettent aux agents de participer à la satisfaction d'un but global. Cette participation permet au système d'évoluer vers un de ses objectifs et d'avoir un comportement intelligent indépendamment du degré de complexité des agents qui le composent.

En général, les interactions sont mises en œuvre par un transfert d'informations entre agents ou entre l'environnement et les agents, soit par perception, soit par communication. Par la perception, les agents ont connaissance d'un changement de comportement d'un tiers au travers du milieu. Par la communication, un agent fait un acte délibéré de transfert d'informations vers un ou plusieurs autres agents. L'interaction peut être décomposée en trois phases non nécessairement séquentielles :

- La réception d'informations ou la perception d'un changement,
  - Le raisonnement sur les autres agents à partir des informations acquises,
  - Une émission de message(s) ou plusieurs actions (plan d'actions) modifiant l'environnement.
- Cette phase est le résultat d'un raisonnement de l'agent sur son propre savoir-faire et celui des autres agents.

Le degré de complexité des connaissances nécessaires pour traiter les interactions dépend des capacités cognitives (de raisonnement) de l'agent et du fait que l'agent a connaissance ou non de l'objectif du système global. En effet, un agent qui poursuit un objectif individuel au sein du système, comme c'est le cas pour les agents dits réactifs, ne focalise pas son énergie pour interagir avec les autres même s'il y est amené. Par contre, un agent qui participe à la satisfaction du but global du système tout en poursuivant un objectif individuel, va passer une partie de son temps à coopérer ou à se coordonner avec les autres agents. Pour cela, il doit posséder des connaissances sociales qui modélisent ses croyances sur les autres agents. [6]

Selon les agents et le système que l'on cherche à modéliser, les interactions prennent diverses formes :

- **La coopération**

La coopération est une attitude adoptée par les agents qui décident de travailler ensemble. Elle permet à un agent de [3] :

- Mettre à jour les connaissances globales du système.
- Intégrer des informations venant d'autres agents.
- Interrompre son plan d'exécution pour aider les autres agents.
- Déléguer la tâche qu'il ne sait pas résoudre à un autre agent dont il connaît les compétences.

- **La coordination :**

La coordination permet aux agents, d'une part, de s'inspirer de certains résultats des autres agents, et d'autre part, d'éviter de refaire un travail qui a été déjà fait par un autre agent [3].

- **La communication :**

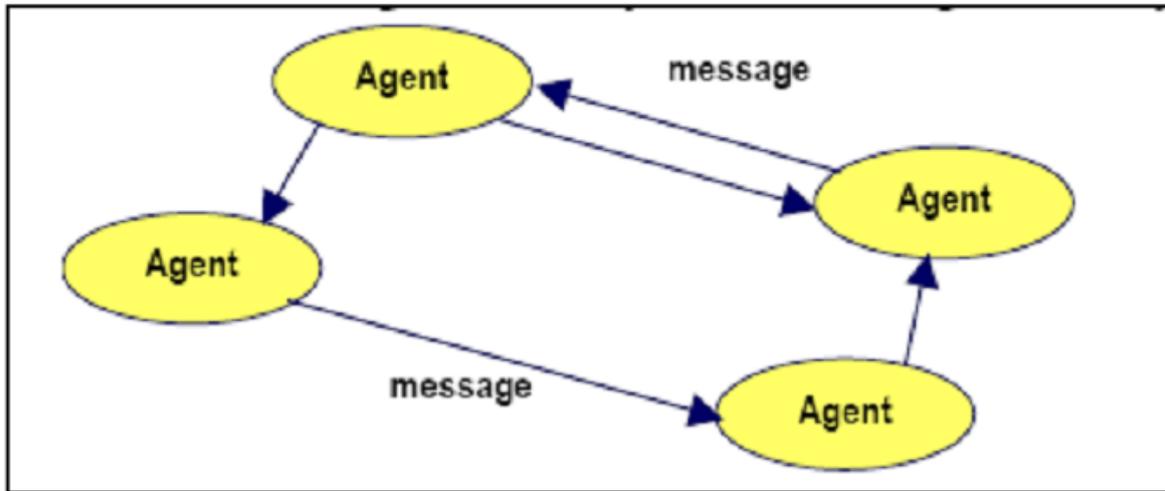
La communication est la base de la résolution coopérative des problèmes. Elle permet de synchroniser les actions des agents et résoudre les conflits de ressources et de buts par la négociation.

Les deux principaux modes de communications dans les systèmes multi agents sont :

- a) Communication par envoi de messages**

Les agents sont en liaison directe et envoient leurs messages directement et explicitement au destinataire. La seule contrainte est la connaissance de l'agent destinataire: « Si un agent A connaît l'agent B, alors il peut entrer en communication avec lui » [7].

- Mode point à point : l'agent émetteur du message connaît et précise l'adresse de ou des agent(s) destinataire(s). Ce type de communication est généralement le plus employé par les agents cognitif.
- Mode par diffusion : le message est envoyé à tous les agents du système. Ce type de transmission est très utilisé dans les systèmes dynamiques ainsi que les systèmes d'agents réactifs. En fait, ceci suppose en général une messagerie : un agent spécialisé gère autant de files d'attente que de destinataires, chaque agent peut traiter le premier message de sa file.



*Figure 4 : Communication par envoi de messages*

### **b) Communication par partage d'information :**

Les agents ne sont pas en liaison directe mais communiquent via une structure de données partagée, où on trouve les connaissances relatives à la résolution (état courant du problème) qui évoluent durant les processus d'exécution.

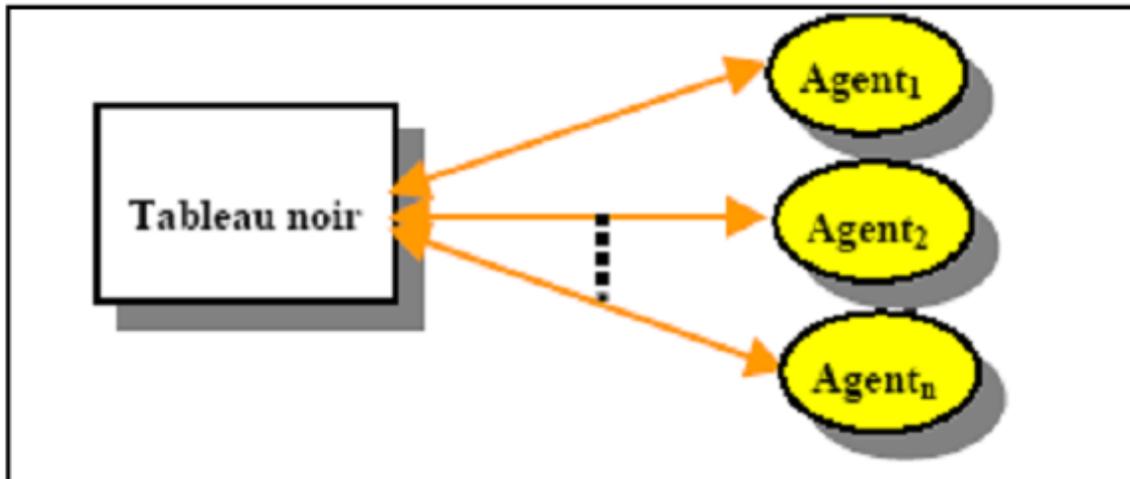
Cette manière de communiquer est l'une des plus utilisées dans la conception des systèmes multi-agents.

Le meilleur exemple qui utilise ce type de communication est les systèmes à tableau noir.

Ce mode de communication n'existe pas dans les SMA où l'on ne dispose que d'une vision partielle du système, car la communication par partage d'informations suppose l'existence d'une base partagée sur laquelle les agents viennent lire et écrire.

Ce type de communication est utilisé quand:

- Il y a recouvrement des domaines d'expertise de chaque agent.
- Il suppose également que les agents ne possèdent qu'une connaissance limitée sur les domaines d'activité des autres agents et pose des problèmes de synchronisation.



*Figure 5 : Communication par partage d'informations*

### 3.5. Les caractéristiques d'un SMA

Généralement, un SMA possède les caractéristiques suivantes :

- chaque agent a des informations ou des capacités de résolution de problèmes limitées, ainsi chaque agent a un point de vue partiel;
- Les agents agissent et travaillent indépendamment les uns des autres.
- Chaque agent travaille dans le but d'accomplir ses tâches.
- Chaque agent est capable de communiquer et d'interagir avec d'autres agents.
- Un agent est capable de coordonner ses activités avec les autres agents pour accéder à des ressources et à des services partagés dont il a besoin (pour réaliser ses buts).
- Il n'y a aucun contrôle global du système multi agent ;
- Les données sont décentralisées;
- Le calcul est asynchrone.
- Dans un SMA rien n'est complètement global: L'environnement étant vaste et ouvert, il n'est pas possible en un lieu donné (par exemple, dans un agent) de stocker toute la représentation du monde. Par contre, un agent peut se déplacer ou encore interagir avec d'autres agents qui sont dans son voisinage pour explorer l'environnement.
- Dans un SMA rien n'est complètement local: Pour un agent donné, toutes ses entités (informations, processus, buts, etc.) sont locales mais elles restent accessibles à l'introspection par d'autres agents [8].

### 3.6. Les langages de communication

Lorsque la notion de groupe d'agents est apparue, les recherches sur des mécanismes permettant aux agents de communiquer entre eux se sont amorcées.

Celles-ci ont mené à des langages de communication et de représentation de l'information qui sont devenus des « standards » en SMA : KQML (KnowledgeQuery Manipulation Language). KQML détermine un format pour les messages et un protocole pour la réception et l'envoi de ces derniers. KQML permet aux agents de partager des informations avec les autres agents du système afin de coopérer pour résoudre un problème [9].

Il y a d'autres langages : le langage (FIPA ACL), est un langage semblable à KQML auquel des protocoles d'interaction comme le contractnet et autres protocoles populaires ont été ajoutés.

### 3.7. Les rôles de Système Multi Agent

**Résoudre un problème de manière distribuée** : il s'agit en général d'agents rationnels, chaque agent sait traiter tout ou une partie du problème, il peut faire appel à d'autres agents pour certains sous problèmes. Plusieurs agents peuvent avoir des méthodes différentes pour les mêmes problèmes. Le résultat recherché est construit par un des agents, soit parce qu'il est prévu pour récupérer les actions des autres agents et les composer, soit parce qu'il arrive à un résultat du fait des actions des différents agents. Le système s'arrête quand le résultat est obtenu.

**Simuler un phénomène complexe** : la plupart des exemples utilisent des agents réactifs. Les agents simulent des actions physiques, biologiques ou sociales qui produisent des modifications du monde représenté. Il n'y a pas de résultat au sens propre, on s'intéresse à la dynamique qui émerge du comportement des agents, elle est observée par un agent extérieur sans que les agents participant au système ne la construisent explicitement. Ces systèmes de simulation ne s'arrêtent pas en général par eux-mêmes, mais du fait d'une action externe (soit le système est lancé pour un temps déterminé, soit l'utilisateur lance la commande Quitter). Ils sont couplés à des systèmes de traces (comptages, statistiques, calcul de fonctions globales) permettant d'analyser les phénomènes simulés et de comparer différents modèles, ou les effets des modifications de certains paramètres sur les modèles.

L'univers est complètement connu, il est déterminé par la représentation du monde qui a été donnée et par l'état du monde au moment du lancement de la simulation. Les résultats sont reproductibles si on initialise de la même manière l'état initial du monde et les fonctions de tirage aléatoire.

Applications : simulation des épidémies animales et de leur traitement social et sanitaire, simulation de la pêche dans le delta du Niger : écosystèmes (proies-prédateurs).

Créer et maintenir un environnement de travail : les actions physiques ou sociales réalisées par les agents sont des actions réelles, elles sont dans le temps et modifient le monde (exemple : les

robots qui sont allés ramasser des pierres sur la lune, les robots footballeurs, les agents représentant des humains pour des opérations de négociation comme «fixer un rendez-vous »).

L'univers n'est pas entièrement connu, chaque agent en a une représentation qui sera confrontée dans l'action avec les réactions du monde ou des autres agents.

Ces systèmes doivent donc évoluer avec le temps et leurs résultats ne sont pas reproductibles car les événements dans le monde changent continuellement [2].

### **3.8. Les avantages des SMA**

Les systèmes multi agents possèdent également les avantages traditionnels de la résolution distribuée et concurrente de problèmes [10] :

- L'autonomie et le contrôle décentralisé sont les caractéristiques les plus significatives du SMA qui sert à distinguer de tels systèmes des approches de calcul distribuées ou parallèles. L'autonomie implique que chaque agent a sa propre, indépendamment de l'existence des autres agents, il prend des décisions motivées par son état interne sans contrôle extérieur.
- Le Contrôle décentralisé supporte également l'extensibilité du fait qu'une fonctionnalité supplémentaire peut être ajoutée simplement en incluant d'autres agents.
- La modularité permet de rendre la programmation plus simple. Elle permet, de plus, aux systèmes multi-agents d'être facilement extensibles, parce qu'il est plus facile d'ajouter de nouveaux agents à un système multi-agent que d'ajouter de nouvelles capacités à un système monolithique.
- La robustesse, est une caractéristique du contrôle décentralisé, où le système global continue à fonctionner même si un certain nombre d'agents ont déconnecté.
- La vitesse est principalement due au parallélisme, car plusieurs agents peuvent travailler en même temps pour la résolution d'un problème.
- La fiabilité peut être également atteinte, dans la mesure où le contrôle et les responsabilités étant partagés entre les différents agents, le système peut tolérer la défaillance d'un ou de plusieurs agents. Si une seule entité contrôle tout, alors une seule défaillance de cette entité fera en sorte que tout le système tombera en panne.
- L'adaptabilité : Les systèmes multi-agents peuvent s'adapter à des modifications de structure ou d'environnement.
- Les systèmes multi agents permettent de modéliser les systèmes complexes et distribués.
- Conception génie logiciel en termes d'unités autonomes en interactions.
- Et d'autres avantages : parallèle, flexible, récursif et facile.

- Finalement, les systèmes multi-agents héritent aussi des bénéfices envisageables du domaine de l'intelligence artificielle comme le traitement symbolique (au niveau des connaissances), la facilité de maintenance, la réutilisation et la portabilité.

### **3.9. Domaines d'application des systèmes multi agents**

Les systèmes multi agents étant issus du domaine de l'intelligence artificielle distribuée, ils permettent de modéliser des applications où une modélisation classique est inadéquate, et où le système est souvent naturellement distribué. Si un problème n'est pas naturellement distribué, on peut choisir de le distribuer pour des raisons de simplification. C'est alors à la charge du concepteur de trouver la meilleure distribution possible afin qu'elle corresponde au problème qu'il souhaite résoudre. Les applications qui sont naturellement distribuées sont, par exemple, celles concernant la simulation d'écosystèmes où l'on associe à chaque individu un agent.

#### **• Ecosystème et modèle individus centrés**

Les modèles individus centrés, c'est-à-dire orientés individus sont des simulations basées sur les conséquences globales d'interactions locales entre membres d'une application.

Ces individus peuvent représenter des plantes et des animaux dans un écosystème, des véhicules dans la circulation, des personnes dans une foule, ou des personnages autonomes dans une animation ou un jeu. Ces modèles consistent typiquement en un environnement ou un cadre globale dans lequel les interactions se passent et un certain nombre d'individus définit par leur comportement (règle comportementale) et des paramètres caractéristiques.

Dans un modèle orienté-individus, les caractéristiques de chaque individu peuvent être suivies de façon continue. Ceci contraste avec les techniques de modélisation où le modèle tente de simuler des changements dans les caractéristiques moyennées d'une population globale. Les modèles orientés individus sont aussi appelés « orientés entités » ou « orientés-agents », et on parle de simulation multi agents. Dans ces systèmes, un agent (individu) correspond toujours à une entité du monde réel (un animal, une personne, un objet, une bactérie, etc.). De nombreuses applications de simulation utilisent des SMA basés sur un modèle individu centré [12].

#### **• Systèmes complexes**

Les systèmes complexes sont des systèmes où les techniques de modélisation classique sont difficilement utilisables. En effet, dans ces systèmes, les paramètres sont beaucoup trop nombreux ou contradictoires pour pouvoir être pris en compte.

Parfois même, il n'est pas possible de connaître l'ensemble des paramètres qui interviennent pour la modélisation. L'approche multi agents permet alors d'avoir recours à une modélisation locale. Cette modélisation permet grâce aux principes d'émergence présents dans les SMA d'obtenir un système ayant les propriétés attendues. Il s'agit notamment de l'ensemble des systèmes permettant la simulation d'écosystèmes. Un bon exemple concerne la simulation d'une fourmilière [13]. En effet, lors de ses travaux de thèses, Alexis Drogoul a modélisé une fourmilière en utilisant des agents pour modéliser les fourmis. Il a alors réussi à démontrer que

l'on pouvait obtenir un but global qui était la survie de la fourmilière sans jamais avoir programmé cet élément dans le système mais uniquement à partir de l'interaction des agents par émergence organisationnelle.

- **Systèmes d'aide à la décision et SMA**

Les systèmes d'aide à la décision (SAD) sont présents dans de nombreux domaines et ont pour objectif d'aider le décideur dans sa tâche en lui fournissant tous les éléments pertinents pour la prise de décision. Cela consiste très souvent à extraire de l'information depuis de multiples sources et à la traiter. Des traitements de l'information pour une prise de décision multicritères sont alors nécessaires. Les systèmes multi agents apparaissent comme étant bien adaptés pour traiter de l'information qui peut revêtir diverses formes et provenir de diverses sources. En effet, il faut pouvoir faire la corrélation entre l'ensemble des éléments obtenus pour les présenter à l'utilisateur. L'approche à base d'agents permet d'effectuer cette corrélation en utilisant la négociation et la coopération entre agents. Il s'agit donc d'un domaine d'application pour les SMA [14].

- **Le domaine du commerce électronique et les agents du WEB**

Le domaine du commerce électronique est un domaine en plein essor qui permet de favoriser les transactions commerciales, Ce domaine utilise l'outil informatique et plus particulièrement les ressources mises à disposition par Internet pour rapprocher les acteurs commerciaux dans certains domaines. Ce domaine se rapproche de celui de l'aide à la décision et peut même être confondu avec celui-ci dans certaines circonstances car il est caractérisé par la mise en place de moyens permettant d'extraire de l'information sur les produits, les marchés, Les acteurs du marché [15].

L'utilisation de systèmes multi agents est une bonne solution ici aussi car elle permet comme pour le domaine de l'aide à la décision de faire de la fusion d'informations. La dénomination «agent du WEB» désigne les agents qui circulent sur le réseau Internet pour extraire de l'information. Ces agents qui sont en général mobiles se déplacent au travers des sites WEB. Certains systèmes ont pour objectif de fournir au consommateur une valeur ajoutée en lui fournissant des informations supplémentaires afin qu'il fasse un choix approprié selon les critères qu'il s'est fixé [16]. Il peut s'agir d'informations comparatives entre produits ou encore d'études qualitatives.

Prenons un exemple concret: vous souhaitez acheter une nouvelle voiture mais vous n'avez pas encore décidé du modèle exacte. Jusqu'à maintenant, vous deviez visiter un certain nombre de concessionnaires, étudier des documentations techniques (parfois ardues), acheter des magazines spécialisées qui effectuent eux-mêmes des tests et études comparatives. Tout ce travail est fastidieux et coûteux en temps. Le but des applications dans le domaine du commerce électronique va être d'extraire toute cette information qui est disponible à divers endroits du WEB et de n'en retenir que celle qui vous intéresse. Il va s'agir d'un travail d'extraction, de filtrage et d'analyse de l'information disponible. La multiplicité des sources d'informations disponibles sur le WEB et la complexité des traitements à effectuer rend particulièrement

appropriée l'utilisation du paradigme agent pour le développement des applications de commerce électronique [14].

### **Conclusion**

Le thème des SMA, s'il n'est pas récent, est actuellement un champ de recherche très actif. C'est une discipline qui s'intéresse aux comportements collectifs produits par les interactions de plusieurs entités appelées agents, ces interactions peuvent être mise en œuvre par des protocoles d'interactions.

Ce chapitre débute par la définition du concept d'agent comme étant un système informatique situé dans un environnement, autonome et flexible et en ajoutant que c'est la présence de ces trois caractéristiques qui donne la force au paradigme agent.

Nous avons aussi introduit, brièvement, les systèmes multi agents qu'ils pouvaient être très utiles dans certaines situations et qu'ils avaient plusieurs avantages intéressants comme la modularité, la vitesse et la fiabilité. La technologie agent semble prendre de plus en plus d'importance. En effet, elle permet de répondre aux besoins de nombreux domaines d'application jusque-là difficilement abordable par des moyens traditionnels. Il s'agit notamment des systèmes complexes, des systèmes d'aide à la décision, du commerce électronique, etc.

# Chapitre II<sub>La</sub>

---

communication entre les agents



## Introduction

La communication entre agents est primordiale à l'intérieur des systèmes multi agents. Les agents doivent communiquer pour se coordonner et pour échanger de l'information, ils peuvent donc, par l'intermédiaire de la communication, échanger de l'information et des connaissances pour déterminer leurs actions en fonction du comportement des autres agents du système.

En réalité virtuelle, on cherche à créer des agents ayant des comportements crédibles aux yeux de l'utilisateur. Ceci passe par un rendu de l'agent dans l'environnement (expressions faciales, gestuelles, etc.), mais surtout par des interactions avec l'utilisateur et/ou ses congénères, qui soient totalement compréhensibles.

On s'attend par exemple à des échanges de connaissances et/ou de services entre les entités qui peuplent ces mondes virtuels, et l'utilisateur aimerait également pouvoir communiquer avec les différents agents pour influencer leurs comportements.

### 1. Les Actes de Langage

Les systèmes de communication entre agents sont principalement basés sur le modèle de la théorie des actes de langage proposé par les linguistes pour modéliser les échanges d'informations entre humains.



*Figure 6 : Communication naturelle entre deux individus*

En 1962, Austin[30] pose les fondements du concept d'acte de langage dans « Quand dire, C'est faire ». Comme son titre l'indique, il considère la communication comme une action sur l'environnement. Une demande de service revient à la réalisation même de ce service (du moins en ce qui concerne l'état de l'environnement). Après un travail de fondation important réalisé par Austin, Searle et Vanderveken [32] ont formalisé cette théorie.

Au cours de ces travaux, ils extraient du langage naturel un groupe de verbes qu'ils baptisent performatifs<sup>2</sup> car le fait même de les prononcer a une influence sur l'environnement, ou plutôt sur les entités qui perçoivent le message. C'est ainsi que l'annonce d'une tragédie va déclencher des réactions fortes par le simple fait de son énonciation.

Ils distinguent également trois aspects dans un message : le locutoire (le message proprement dit), l'illocutoire (l'acte que représente le message) et le perlocutoire (les effets du message). Ainsi, le message « il neige ce matin » est le niveau locutoire, alors que la dimension illocutoire est la prise de conscience par le récepteur du fait qu'il neige (assertion dans sa base de connaissances), et la dimension perlocutoire est une invitation à se couvrir.

Ils séparent en fin les performatifs en cinq classes en fonction de leurs buts illocutoires :

- **les assertifs** : ajoutent des informations sur le monde (affirmer, etc.).
- **les directifs** : poussent à la transformation du monde (demander, interdire, etc.).
- **les engageants** : engagent l'émetteur (s'engager à, promettre, etc.).
- **les expressifs** : n'ont d'effets que sur la communication (remercier, saluer, etc.).
- **les déclaratifs** : affectent le monde et tout les interactants (déclarer la guerre, etc.).

C'est de cette modélisation du langage que sont issus les langages de communication entre agents (ACL) que nous allons présenter dans la section suivante.

## 2. Langages de communication entre agents

Les ACLs sont donc basés sur la linguistique et principalement la théorie des Actes de Langage. Il s'agit de permettre l'échange d'informations entre entités "intelligentes", mais également de faciliter l'interopérabilité entre les différentes plateformes multi-agents. Ces langages font donc abstraction du mode de transport et sont codés sous forme de chaînes de caractères ASCII.

Il existe principalement deux langages spécifiés à partir de ces théories : KQML<sup>3</sup> et FIPA ACL<sup>4</sup>.

---

<sup>2</sup>De l'anglais to perform : exécuter, faire une action.

<sup>3</sup> KQML : Knowledge Query Manipulation Language.

<sup>4</sup>FIPA : Foundation for Intelligent Physical Agents.

## 2.1. KQML

KQML est un ACL développé au début des années 90, par l'équipe External Interfaces Working Group du projet DARPA9-KSE10. Le but de KQML était principalement de fournir un langage de communication basé sur les actes de langage et indépendant des applications qui l'utilisent.

Ce langage de communication entre Agents a été spécifié par Labrou[42]. Il définit 35 performatifs réservés, et leurs sémantiques.

### 2.1.1.Syntaxe

La Syntaxe de KQML est inspirée de celle de Common Lisp. Les performatifs sont vus comme des fonctions et les différents champs comme des arguments.

Les messages sont composés sur trois niveaux distincts (voir figure 3) :

**-Communication** : permet le bon acheminement du message et sa mise en relation avec les autres messages d'une même conversation.

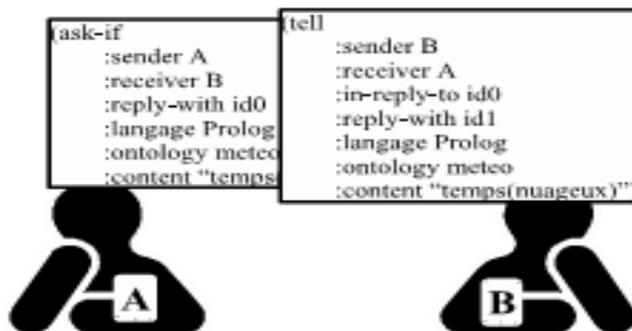


Figure 7 : Echange d'informations entre agents utilisant KQML

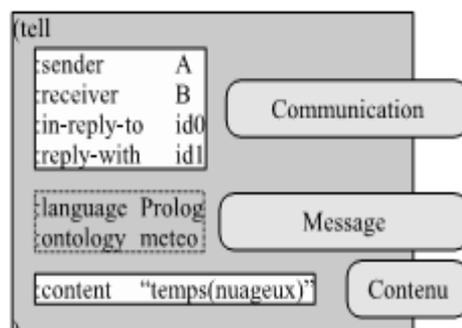


Figure 8 : Syntaxe du performatif tell de KQML

Ce sont les champs : **:sender**, **:receiver**, **:in-reply-to**, etc.

- **Message** : contient les informations nécessaires à la bonne compréhension de la sémantique du message. Il est composé notamment des champs **:language** et **:ontology**, mais aussi du performatif lui-même.

- **Contenu** : le message proprement dit (le champ **:content**).

Prenons l'exemple de la figure 3. Par ce message, l'agent A fait savoir à l'agent B que le temps est nuageux. Il lui donne également tous les éléments pour comprendre ce contenu. Le performatif **tell** est un assertif, donc si le message est accepté, son contenu devra être ajouté aux connaissances de l'agent B. Il s'agit d'un prédicat Prolog, et le vocabulaire est celui de la Météo. Le champ **:in-reply-to** met en relation ce message avec une possible question précédente, alors que **:reply-with** permet au récepteur de continuer la conversation par un acquittement ou un démenti par exemple.

L'émetteur, par l'envoi de ce message, exprime un certain état interne, fonction de ses croyances et ses désirs. Il relève sa vision du monde, mais aussi sa volonté (besoin d'informations ou d'actions). C'est le propos de la sémantique associée à tout message KQML.

### 2.1.2.Sémantique

La sémantique de KQML est définie à partir des états cognitifs des agents qui s'échangent le message. Les états internes des agents sont représentés par les opérateurs : **Bel**, **Know**, **Want** et **Intend** qui signifient :

**Bel(A, P)** : l'agent A croit que la proposition P est vraie.

**Know(A, P)** : l'agent A sait ce que vaut P, P est dans sa base de connaissances.

**Want(A, P)** : l'agent A aimerait que la proposition P devienne vraie.

**Intend(A, P)** : l'agent A a l'intention que P se produise, il a mis en place un plan d'actions pour y arriver. Labrou[42] définit la sémantique de chacun de ses performatifs selon le protocole suivant :

1. Une description intuitive en langage naturel.
2. Formalisation de cette description par une expression qui décrit l'acte illocutoire porté par le performatif.
3. Définition à l'aide des opérateurs vus précédemment des préconditions sur les états internes des interactants, préconditions nécessaires à l'émission du message.
4. L'effet du message est décrit par les post-conditions qui représentent l'état des protagonistes après le transfert d'informations.
5. La '**Completion**' représente l'état qui devrait être atteint après une conversation. On y retrouve l'intention de l'initiateur de l'échange.

### 2.1.3.Implémentations

KQML a donné lieu à un certain nombre d'implémentations, comme JavaAgent Template de l'université de Stanford ou Jackal de l'université du Maryland (UMBC). Il a également donné

lieu à une implémentation non diffusée sous forme de packages oRis[18], l'ancien environnement multi-agents développé au LI2.

L'agent Conversationnel Rea[29], développé par l'équipe de Bickmore et Cassel au MIT, est également implémenté avec KQML. Cette utilisation est interne à son architecture [29], les différents modules s'échangent des messages KQML générés à partir des paroles et des mouvements de l'utilisateur.

### **2.1.4.Problèmes de KQML**

Dans la première version de KQML, la sémantique n'était pas définie de manière formelle mais simplement en langage naturel. C'est ce qui lui a valu les plus grandes critiques [43]. En effet, d'après Ferber[3] cette description informelle de la signification des performatifs en faisait un langage ambigu et imprécis.

Comme nous l'avons vu précédemment, Labrou[42] lui a ajouté cette sémantique en 1996. Mais cette sémantique présuppose que les agents embarquent une base de connaissances [37], et la présentation des préconditions sous-entend également d'avoir des connaissances sur l'état interne de ses interlocuteurs, ce qui peut rapidement devenir une masse importante d'informations.

Les premières critiques sur KQML ont donné lieu à un certain nombre de recherches pour obtenir des langages à sémantique formelle, dont FIPA ACL est le principal représentant.

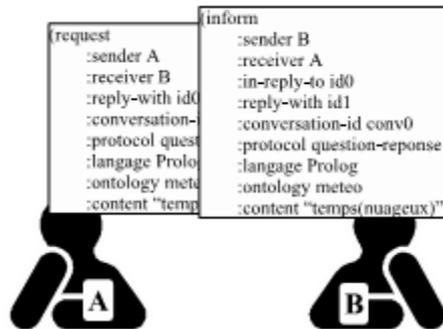
## **2.2. FIPA ACL**

FIPA ACL est, comme son nom l'indique, un projet de la FIPA pour fournir un ACL simplifiant l'interopérabilité entre les différentes plateformes multi-agents.

La FIPA est une association à but non lucratif, elle réunit de grandes sociétés de télécommunication aussi bien américaines qu'européennes, comme : British Telecom, France Telecom, Nortel, Siemens, etc. Les buts principaux de cette organisation sont la promotion des applications utilisant des agents et la mise en place de standards pour faciliter l'interopérabilité entre les différentes plateformes multi-agents. La différence majeure entre FIPA ACL et KQML est sa sémantique tirée d'ARCOL (un ACL développé chez France Télécom par David Sadek au cours de sa thèse publiée en 1991).

FIPA défini 4 performatifs<sup>5</sup> de base (inform, request, confirm et disconfirm) à partir desquels tout acte de langage doit pouvoir être exprimé par composition.

Dans sa version actuelle publiée en 2000, la librairie de performatifs FIPA en contient 22 (en comptant les 4 de base).



*Figure 9 : Echange d'informations entre agents utilisant FIPA ACL*

### 2.2.1. Syntaxe

La syntaxe de FIPA ACL[24] est très proche de celle de KQML. La liste de performatifs est totalement différente, mais les messages FIPA ACL reprennent les champs KQML auxquels sont ajoutés quelques autres comme par exemple :conversation-id et :protocol, qui sont utilisés pour mettre le message en relation avec son contexte (autres messages d'un même échange). Ils seront utilisés dans les protocoles d'échange détaillés.

### 2.2.2. Représentation XML

Pour simplifier le décodage des messages, des encodages plus génériques ont été mis en place.

L'équipe de Labrou propose ACML<sup>6</sup>[21] qui est une spécification de FIPA ACL en XML. La FIPA propose, elle, une simple spécification d'encodage en XML[24] de sa bibliothèque de performatifs. On retrouve les mêmes champs que dans l'encodage de base, mais avec une syntaxe décrite en XML.

Cet encodage présente l'avantage de standardiser le décodage des messages. Cela permet d'utiliser les parseurs XML standards qui sont devenus courants et qui sont mieux maintenus puisqu'ils sont utilisés dans un grand nombre d'applications.

---

<sup>5</sup>plutôt appelés Communicative Acts dans les spécifications FIPA.

<sup>6</sup> ACML : Agent Communication MarkupLanguage.

### 2.2.3. Sémantique

La sémantique de FIPA ACL[24] est définie en SL<sup>7</sup>[24]. Les propositions y sont exprimées sous la forme d'états mentaux et d'une logique de l'action.

Le modèle mental d'un agent est caractérisé par les trois opérateurs de jugement des propositions : **B(Believe)**, **U(Uncertain)** et **C(Choice)** indicés du nom d'un agent et suivis d'une proposition. Ces trois opérateurs signifient respectivement : que l'agent croit la proposition vraie, qu'il n'en est pas sûr (mais que pour lui elle est plutôt vraie) et qu'il aimerait qu'elle devienne vraie. SL contient également une logique de l'action permettant d'exprimer les capacités d'agir, par exemple, et un certain nombre de propriétés permettant de raisonner autour d'un ensemble de propositions. Quelques abréviations permettent d'éviter l'explosion des formules, par exemple BifiX signifie que l'agent A connaît la valeur de X (BifiX équivaut à  $Bi(X) \_ Bi(:X)$ ).

A chaque performatif est associé des préconditions de faisabilité (FP : FeasibilityPreconditions) et un effet rationnel (RE : Rational Effect) qui correspond à l'effet perlocutoire de la théorie des Actes de Langage.

### 2.2.4. Implémentations

Actuellement, de nombreuses plateformes multi-agents sont FIPA Compliant, c'est à dire qu'elles sont capables de parser un message FIPA ACL et d'y répondre par un 'not-understood' si elles ne sont pas capables de l'interpréter (à cause d'un langage de contenu inconnu par exemple). La plupart des industriels faisant partie de la FIPA ont développé un environnement issu de ces spécifications. Parmi ces implémentations on peut citer : FIPA OS[25], JADE[27], etc.

Mais, par défaut, ces implémentations ne proposent pas de vérifications explicites de la sémantique des messages. C'est à dire que ces plateformes ne vérifient pas les états mentaux des protagonistes, les agents fournis n'ont d'ailleurs aucune base de connaissances prévue à cet effet. Dans le cas de JADE, on peut utiliser JESS[28] pour obtenir une vérification d'une partie de cette sémantique. Sinon, la tâche de vérification de la sémantique est donc laissée aux développeurs d'agents "intelligents", qui auraient besoin de raisonner autour des croyances et intentions exprimées par les messages. Les développeurs d'agents moins évolués doivent

---

<sup>7</sup> SL : "Semantic Langage", le langage de contenu spécifié par FIPA.

s'efforcer d'utiliser un enchaînement de messages cohérents avec la sémantique spécifiée par FIPA pour obtenir l'interopérabilité avec les autres agents et les autres plateformes.

### 2.3. Problèmes des ACLs

Bien que ce réclamant de la théorie des actes de langage, les ACLs n'en ont retenu que certains aspects formels et terminologiques [26]. Certaines forces illocutoires sont mal représentées : 4 performatifs de base en FIPA ACL qui ne représentent que les assertifs et les directifs, les **commissifs** pouvant être simulés [37].

Ces langages imposent de par leur sémantique d'utiliser des agents ayant des capacités de réflexion autour de leurs croyances et leurs intentions et exigent une sincérité des agents participants à la conversation [20]. Les différents agents n'ont pas la possibilité de vérifier que les effets d'un message (en termes d'états mentaux) ont été effectivement pris en compte. Les préconditions sont également trop restrictives : impossibilité de confirmer ou de mentir.

De plus, que ce soit KQML ou FIPA ACL, les Langages de Communication entre Agents restent des langages pour faire communiquer des entités numériques entre elles. Le dialogue entre êtres humains reste beaucoup moins rigoureux et beaucoup plus complexe que les échanges entre agents informatiques.

Enfin, le langage naturel offre une plus grande variété de performatifs que ces langages informatiques. Ainsi 'request' peut aussi bien vouloir dire exiger que supplier.

### 2.4. Langages sociaux et engagements

Pour répondre à un certain nombre de problèmes des ACLs, Singh[37] propose l'utilisation de langages se focalisant sur la composante sociale de la communication plutôt que sur son **e et mental**. La couche sociale étant publique, elle est facilement vérifiable et la communication n'a plus d'exigence en ce qui concerne l'architecture des agents.

L'équipe de Chaib-Draa[40] au laboratoire DAMAS a, quant à elle, réalisé un langage à base d'engagements sociaux. Les engagements sont pris publiquement et s'ils ne sont pas respectés, les sanctions qui leurs sont associées s'appliquent.

Ils définissent également des jeux de dialogue qui permettent aux agents de négocier les engagements qu'ils prennent les uns envers les autres.

## 3. Contenu des messages ACL

Les ACLs prennent en charge l'acheminement d'un message et la sémantique à lui associer, mais ils ne se préoccupent pas du sens du contenu qu'ils encapsulent. Ils n'ont d'ailleurs pas de préférence quant au langage choisi pour donner du sens à ce contenu. Ainsi, on peut tout à fait choisir un langage existant comme par exemple Prolog (voir les exemples de la section précédente) ou SQL pour jouer ce rôle.

Les équipes qui ont développé les ACLs ont malgré tout défini des langages pour donner du sens au contenu des messages. Les deux principaux langages de contenu développés autour des ACLs sont : KIF et SL.

### 3.1. KQML & KIF

KIF<sup>8</sup> est un langage issu du projet DARPA-KSE (comme KQML) et la FIPA[24] en a intégré une version à ses spécifications.

L'étude de KIF avait deux buts :

1. créer un langage pour permettre à des applications intelligentes de s'échanger des connaissances en minimisant les problèmes d'interopérabilité.
2. créer un langage commun de traduction pour faciliter la communication entre des applications n'utilisant pas le même langage de contenu.

Ce langage permet d'exprimer des connaissances mais aussi des informations sur ces connaissances, des liens logiques entre les concepts. Il permet donc de raisonner sur l'environnement, mais aussi sur la façon de le représenter.

### 3.2. FIPA ACL & SL

Comme nous l'avons vu précédemment, SL est le langage formel de description de la sémantique de FIPA ACL. Ce langage peut également être utilisé en temps que langage de contenu, pour communiquer autour d'états internes par exemple. Il permet également de communiquer autour d'actions, la figure 5 montre comment on exprime en SL le fait qu'un agent de nom A fait l'action 'avancer' sur une longueur de '10' centimètres. Le performatif dans lequel cette expression est incluse, détermine s'il s'agit d'une information, d'une requête, etc.

*(action (agent-identifier name A) (avancer 10))*

Fig. 5 Expression d'une action de déplacement en SL

Pour en simplifier l'usage, la FIPA a défini trois sous-ensembles de SL (SL0, SL1 et SL2). Ces sous-ensembles expriment moins de choses que SL, mais ils sont plus simples à interpréter.

Pour utiliser ces langages, il suffit de donner la valeur fipa-slX (ou X est vide pour SL, 1 pour SL1, etc.) au champ :language du message ACL.

---

<sup>8</sup>KIF : KnowledgeInterchange Format

### 3.3. Ontologies

En fonction du langage de contenu qu'ils utiliseront, les agents pourront s'échanger des connaissances de différents types. Mais pour les comprendre, il leur faut aussi utiliser un vocabulaire commun représenté par les ontologies.

Les ontologies représentent le domaine **du** discours. Deux agents pour pouvoir se comprendre doivent communiquer en utilisant le même vocabulaire, ou tout au moins des vocabulaires que chacun connaît. Dans le message ACL, la valeur du champ `:ontology` permet d'identifier le domaine de discours.

Pour la communication entre agents simples, cette ontologie peut tout à fait être implicite. Le sens à associer aux informations reçues est alors embarqué dans l'agent, il n'utilise le champ `:ontology` que pour s'assurer qu'il comprendra bien le contenu. C'était le cas dans l'exemple de la partie précédente, lorsque les agents discutaient de la météo.

En revanche, des agents plus complexes peuvent utiliser des vocabulaires plus élaborés, dans lesquels les différents mots ont des relations d'héritage par exemple (la truite est un poisson). Il devient alors intéressant d'extraire de l'agent cette représentation du domaine de discours pour en faire un service et faciliter les traductions [24, 39].

## 4. ACL & Facilitateurs

Un échange de connaissances entre deux agents peut se faire sous différentes formes, mais il ne se fait pas sans raison. L'agent qui commence une conversation attend quelque chose de l'interaction qu'il met en place : il a besoin d'un service (d'une information, d'une action, etc.). Avant de contacter un agent, il est préférable qu'il s'assure que son interlocuteur pourra lui fournir ce qu'il recherche. Bien souvent d'ailleurs, il ne saura pas à qui s'adresser pour obtenir ce dont il a besoin. Il s'adresse donc à un agent facilitateur, qui lui fournira ces informations.

Les facilitateurs sont des agents de l'environnement qui jouent le rôle d'annuaires. Ils peuvent stocker les manières d'accéder aux autres agents et alors représenter les pages blanches d'un annuaire (localisation, protocole de communication bas niveau, etc.) et/ou maintenir le jeu de services que sont capables de fournir les différents agents (équivalent des pages jaunes).

Contrairement à FIPA-ACL, KQML définit un certain nombre de performatifs réservés à la communication avec ce type d'agent. Ces derniers, contrairement aux autres performatifs KQML, n'ont pas de véritable définition formelle de leur sémantique. Ils permettent de déléguer à un agent facilitateur le traitement d'un performatif embarqué dans le champ `:content` du message.

En revanche, la FIPA considère les agents facilitateurs comme des agents normaux qui délivrent un service particulier. Le 'Directory Facilitator' (DF) enregistre chacun des agents de l'environnement et conserve un moyen de les contacter (pages blanches) alors que le « Agent Management System » (AMS) rend publique la liste des services que chacun des agents est prêt

à fournir (pages jaunes). Ces deux agents sont considérés comme indispensables à toute plateforme FIPA **Compliante**, elles doivent donc donner accès à au moins un représentant de chacun d'eux.

## 5. Conversations entre Agents

Un échange de connaissances ne se contente que rarement d'un seul message, en général, il est nécessaire de s'en échanger plusieurs : une question suivie d'une réponse par exemple.

Une conversation entre deux agents correspond à une suite de messages qu'ils s'échangent pour atteindre le but qui a initié le dialogue. Dans le cas d'agents ayant des avis et intentions sur le monde qui les entourent, on pourrait laisser ces intentions guider le choix du message à envoyer. Mais sans autres données que la sémantique des performatifs, ce choix peut s'avérer complexe.

### 5.1. Règles de Conversation

Pour simplifier ce choix, Labrou spécifie 'The Conversation Policies for KQML performative'[42] pour limiter le nombre de réponses possibles à un performatif KQML. Il définit les enchaînements qui ont du sens. Par exemple, une question doit être suivie d'une réponse ou d'une excuse, mais pas d'une autre question. Ces règles ne sont pas obligatoires, elles apportent simplement un cadre pour réaliser des agents moins sophistiqués communiquant par l'intermédiaire de messages KQML. Les agents ayant la capacité de mettre en place des interactions plus complexes peuvent s'en affranchir.

Mais si on souhaite réaliser des agents plus simples (sans avis ni intentions), ces règles ne suffisent pas, il faut définir des protocoles plus strictes spécifiant l'ensemble des conversations possibles pour atteindre un but précis.

### 5.2. Protocoles FIPA

La FIPA[24] propose une bibliothèque de protocoles, destinée à encadrer les échanges dans certaines situations précises. Ces protocoles spécifient ce que doivent s'échanger les interactants et dans quel ordre. A chaque étape du protocole, chacun des interlocuteurs sait : si c'est à lui de s'exprimer ou pas, et ce qu'il a la possibilité d'envoyer ou ce qu'il doit s'attendre à recevoir.

La figure 6 montre la représentation du protocole 'FIPA Request' dans un formalisme inspiré du diagramme de séquence d'UML [24]. On voit que l'agent Initiator met en place le protocole en envoyant un performatif request à l'agent Participant. En fonction de ses capacités à répondre à la demande qui lui est faite, ce dernier répond par une acceptation ou un refus. S'il a accepté de répondre à la requête, l'agent Participant clôture la discussion quand son traitement est terminé en envoyant le résultat de l'action ou la raison de l'échec si elle n'a pas abouti.

Pour utiliser ce protocole, il suffit de donner la valeur fipa-request au champ : protocole du performatif request qui initie la conversation. L'ensemble des messages qui s'échangeront pendant ce dialogue devront également contenir la même valeur non vide du champ :

conversation-id pour permettre aux participants de mettre en relation les différents messages entre eux.

Ces protocoles rendent possible la communication avec des agents peu évolués qui n'ont pas la capacité de réfléchir sur leur façon de communiquer.

Ils permettent également de minimiser les échanges de messages pour les interactions simples, mais leur rigueur ne leur permet pas de s'adapter à toutes les situations.

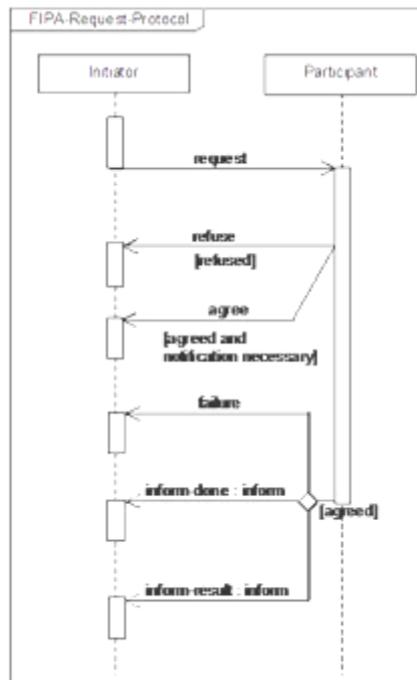


Figure 10 : Spécification du protocole 'FIPA Request'

### 5.3. Conduites de Conversation

Les conduites de conversation se positionnent entre la liberté totale d'expression et les protocoles stricts que nous avons vus précédemment. On spécifie des directions globales à tenir (avec des points de passage par exemple), ce qui permet de limiter les risques d'interblocage. On autorise des variations dans le protocole pour simplifier les interactions entre agents ayant des conceptions différentes.

Mais, la jeunesse du domaine fait qu'il n'existe pas de modèle éprouvé, ni même reconnu ou étudié à grande échelle [26].

## Conclusion

Nous avons donc vu comment on peut faire communiquer des agents à l'aide d'actes de langages. Ces échanges peuvent se faire entre des entités hétérogènes interoperables. Ces entités doivent néanmoins partager un minimum de concepts, pour que l'acte de communication réussisse.

En effet, elles doivent comprendre le sens associé à chaque niveau du message (performatif, langage et ontologie). Il n'est d'ailleurs pas nécessaire que les agents communicants aient les mêmes mécanismes d'interprétation de message, cependant ils doivent en tirer des conclusions équivalentes (au sens échange de connaissances) dans leur représentation respective de l'environnement.

Il faudra donc mettre en œuvre tous ces concepts pour permettre une bonne communication à base d'actes de langage entre agents. Ceci doit tenir compte de l'enchaînement des actes de langage au cours d'une conversation. L'aspect social de la communication devra également être abordé pour étudier son impact sur les échanges entre agents dans l'environnement.

Dans un environnement de Réalité Virtuelle, le passage de la communication Agent/Agent à la communication Utilisateur/Agent devra poser le problème sous l'angle des agents conversationnels ou de la communication Humain/Avatar.

Dans le premier cas l'approche se focalise sur le réalisme d'un agent d'un point de vue communication verbale et non-verbale, alors que dans l'approche où l'humain doit être un modèle parmi les modèles [33], il est nécessaire de se baser sur la prise de contrôle d'un avatar, ce dernier communique avec les autres agents de l'environnement par actes de langages tout comme le feraient deux agents.

# Chapitre III<sub>La</sub>

---

modélisation

## **E-Commerce**

### **1. Introduction**

Depuis l'ouverture au grand public du réseau au début des années 1990, Internet s'est progressivement transformé en un canal de distribution électronique au sein duquel les entreprises et les consommateurs échangent, commercialisent des biens et des services. Conçu à des fins militaires puis universitaires, le réseau Internet n'avait pas pour vocation initiale de permettre la réalisation de transactions commerciales, encore moins de se substituer à d'autres canaux de distribution. Cette transformation d'Internet en un espace économique a cependant été très rapide, et il constitue désormais un espace marchand incontournable. Toutefois, la place d'Internet est très variable selon les secteurs économiques. Après une phase de démarrage euphorique au début des années 2000 (la « bulle Internet »), Internet revient à des réalités plus complexes que le simple basculement de transactions physiques vers des transactions électroniques. En effet, il ne se limite désormais plus au Web, mais est accessible depuis de nombreux autres réseaux et terminaux comme les téléphones mobiles, les tablettes, les télévisions connectées, mais aussi bien d'autres objets encore, comme les voitures équipées de tablettes multimédias intégrées où les transactions marchandes se développent également. Après avoir présenté la réalité actuelle du commerce électronique, les tendances et les facteurs qui influencent son développement, il est nécessaire de se pencher sur la définition du commerce électronique, tant les utilisations d'Internet à des fins marchandes sont multiples. Il convient également de comprendre la variété des différents modèles d'affaires envisageables sur cet espace numérique. Enfin, les raisons de l'utilisation d'un tel canal pour les consommateurs et les entreprises sont examinées, afin de comprendre l'intérêt de cette forme de commerce [44].

### **1. Définition du commerce électronique**

Le commerce électronique peut être défini comme l'ensemble des échanges électroniques liés aux activités commerciales. Il recouvre toute opération de vente de biens et de services via un canal électronique. Internet n'est donc qu'un support parmi d'autres du e-commerce avec, entre autres, l'EDI (échanges des données informatisées), le Minitel (en France) voire même le téléphone (Audiotel) ou la télévision (PayPer-View). Si le commerce électronique n'est pas un phénomène nouveau, le développement très rapide d'Internet et des NTIC lui apporte une nouvelle dimension, puisqu'il permet en théorie de toucher le marché mondial à moindre coût. Cette révolution s'accompagne d'un élargissement de l'offre des biens traditionnels, de l'apparition de nouveaux types de biens (comme les biens informationnels digitaux : logiciels, publications électroniques, etc.) et de nouveaux acteurs (Fournisseurs de services Internet, moteurs de recherche, portails, etc.). A ce titre, la notion de commerce électronique est de plus en plus confondue et réduite au processus de vente de biens et services sur Internet. Elle s'intègre au cadre plus général du business, que l'Electronic Business Group définit comme « l'utilisation de tout ou partie des technologies d'Internet pour transformer le fonctionnement des activités principales de la chaîne de valeur de l'entreprise en vue d'en dégager une valeur, économique supérieure directe ou indirecte ».

Trop souvent assimilé à la seule transaction en ligne (l'acte de vente en lui-même), le e-commerce englobe en fait toutes les étapes traditionnelles de la vente, en amont (recherche du client, diffusion d'informations commerciales, gestion des commandes, livraison) comme en aval (fidélisation, service après vente, relance). Il nécessite la mise en place d'outils spécifiques et l'adaptation des structures déjà existantes au sein de l'entreprise. Le site commercial, vitrine de l'entreprise sur Internet et lieu de vente, n'est donc qu'un élément du processus de vente en ligne, au même titre que l'emploi de la messagerie électronique pour optimiser la relation client ou la création d'un extranet pour faciliter les échanges commerciaux avec les différents partenaires et les fournisseurs. Le commerce électronique offre aux entreprises des perspectives de croissance considérables. Mais il représente aussi un enjeu majeur, nécessitant des changements profonds en termes d'organisation interne, de fonctionnement et de stratégie.

## **2. Objet et formes du commerce électronique**

La mise en place d'une stratégie d'e-business au sein de l'entreprise offre des avantages dans différents domaines : En interne, une optimisation du processus de production (objectifs « juste à temps » et « zéro stock ») et un meilleur partage des connaissances (knowledge management), En externe, l'échange rapide d'informations avec les partenaires de l'entreprise (fournisseurs, clients) via des extranets spécialisés, la gestion de la relation client dans une optique one to one et la vente en ligne proprement dite, qui permet de proposer à un public mondial une offre plus importante, mieux présentée, avec des coûts de distribution moindre (notamment grâce au phénomène de désintermédiation). Sur un plan macroéconomique, le phénomène e-business s'accompagne du développement de trois grands types d'entreprises dont les activités sont liées à Internet : Les générateurs de trafic (portails, fournisseurs d'accès), points d'accès généralistes à Internet, qui concentrent la plus grosse part de l'audience sur le Web, Les « facilitateurs » (logiciels, service, fonds de capital risque), Les sites de commerce électronique, qui se subdivisent en différents domaines en fonction des acteurs impliqués et des produits et services vendus ou échangés[45].

## **3. Chiffres clés et tendances du commerce électronique**

Malgré de très belles réussites d'entreprises désormais mondialement connues, comme Amazon (chiffre d'affaires en 2012 de 61,1 milliards de dollars) ou eBay (chiffre d'affaires en 2012 de 14,1 milliards de dollars), le commerce électronique reste une réalité complexe à appréhender de par sa portée mondiale et sa structure en réseau.

Évaluer l'importance actuelle du commerce électronique est un réel défi pour les entreprises, les investisseurs et les régulateurs. Cette évaluation précise se révèle pourtant nécessaire, car malgré une forte croissance au cours de la dernière décennie, il est important de comprendre que le commerce électronique reste encore une forme de commerce relativement marginale à l'échelle des échanges commerciaux. Par ailleurs, le commerce électronique ne revêt pas la même importance selon les secteurs économiques. La croissance reste très disparate selon les pays, et des évolutions contrastées s'esquissent d'ores et déjà. En outre, il importe de comprendre que le développement futur du commerce électronique repose sur un certain nombre de facteurs indispensables à la consolidation de ce canal de distribution. Ces facteurs

expliquent le niveau de pénétration du commerce électronique dans les différentes zones économiques [44].

#### **4. Le commerce électronique : quelles réalités ?**

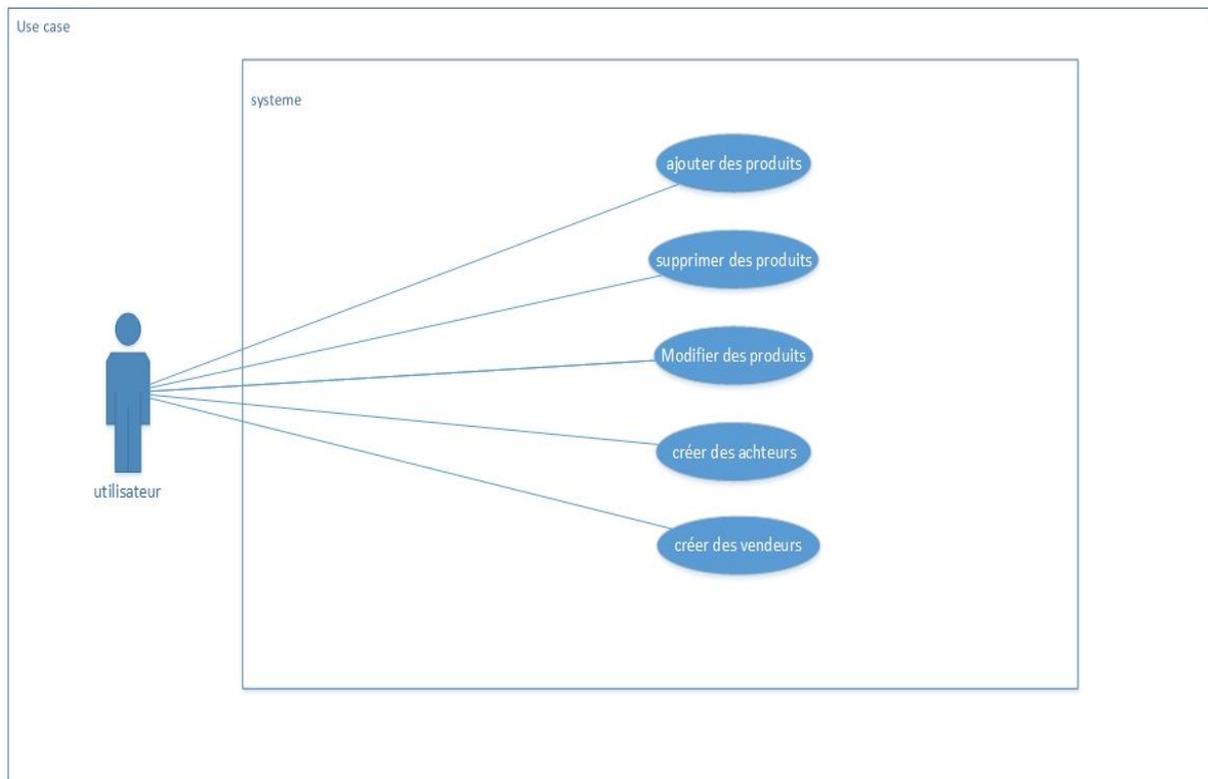
De nombreux experts datent l'émergence du commerce électronique à l'arrivée d'Internet et à la création des sites de vente en ligne aux États-Unis. **C'est oublier** qu'Internet n'est pas le seul réseau électronique permettant de réaliser des transactions commerciales **supprimer**. Bien avant l'existence d'Internet, la France a bénéficié, grâce au Minitel, d'une infrastructure de commerce électronique très aboutie pour l'époque<sup>1</sup>. Aussi, les contours du commerce électronique ne sont pas si simples à dessiner. Des transactions s'effectuent ainsi désormais aussi bien depuis un téléphone mobile ou une tablette, que depuis d'autres objets connectés à Internet comme par exemple la cafetière Nespresso dans sa version B2B ou encore la Renault Zoé. Le poids du commerce électronique dans l'économie reste délicat à évaluer. Les statistiques officielles des États l'appréhendent difficilement. La réalité du commerce électronique est également très différente selon les régions du monde, tout comme elle l'est selon les secteurs dans lesquels il se déploie. En outre, pour certaines industries, ce nouvel espace économique n'est pas toujours appréhendé comme une opportunité de développement [44].

## La modélisation

### 1. Diagramme de cas d'utilisation :

Dans notre système nous distinguons cinq cas d'utilisation :

- ajouter des produits : accorder à l'utilisateur la possibilité d'ajouter de nouveaux produits.
- Modifier des produits : permet la modification des produits déjà existants.
- Supprimer produit : permet la suppression des produits existants.
- Créer des acheteurs : permet l'ajout de nouveaux acheteurs.
- Créer des vendeurs : permet l'attribution de nouveaux vendeurs.



*Figure 11 : Diagramme de CAS d'utilisation*

## 2. Diagrammes de séquences

### a) L'ajout de client

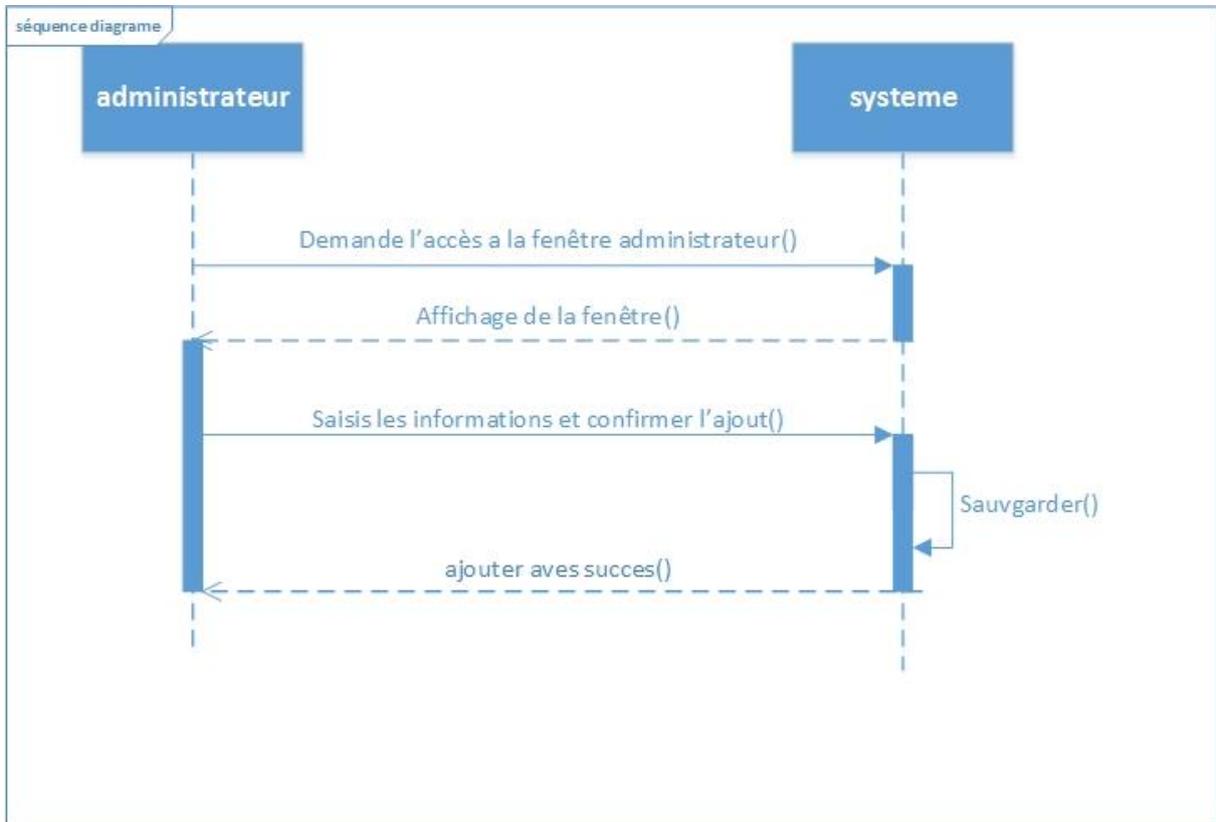
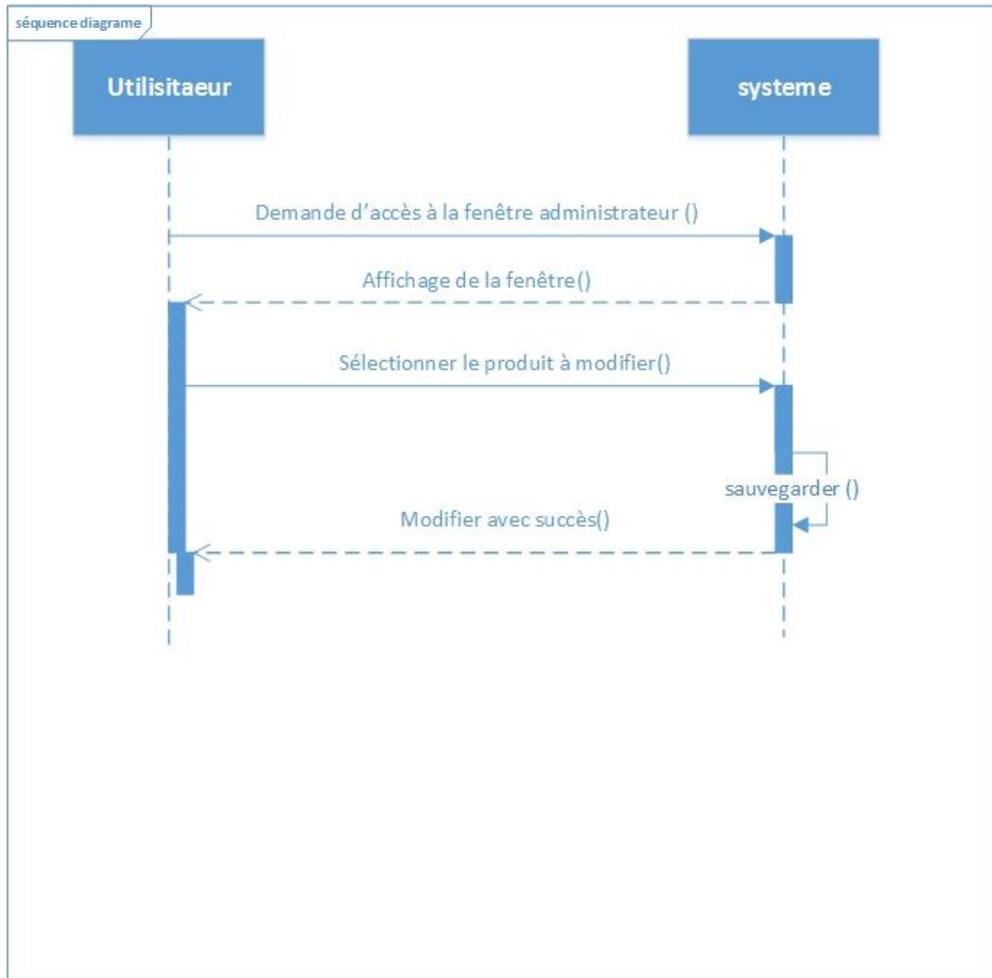


Figure 12 : diagramme de séquence -ajout client

Le diagramme précédent montre les étapes à suivre pour la fonction Ajout de client :

1. l'utilisateur demande un accès aux fonctionnalités de l'administrateur.
2. le système affiche à l'utilisateur la fenêtre de la fonction ajout client.
3. l'utilisateur remplis les champs nécessaires d'un client.
4. les informations saisies seront enregistrées dans la base de données.
5. un pop-up apparaît pour informer que l'enregistrement des données est effectué avec succès.

## b) Modification des produits



**Figure 13 : diagramme de séquence -modifier client**

Ce diagramme montre les étapes à suivre pour la fonction modifier un produit, cette fonction n'est effectuée que par l'administrateur:

1. l'administrateur demande un accès.
2. le système affiche à l'administrateur la fenêtre administrateur.
3. l'administrateur remplit les champs à modifier.
4. les informations saisies seront enregistrées dans la base de données.
5. un pop-up apparaît pour informer que la modification des données est effectuée avec succès.

### c) Supprimer un produit

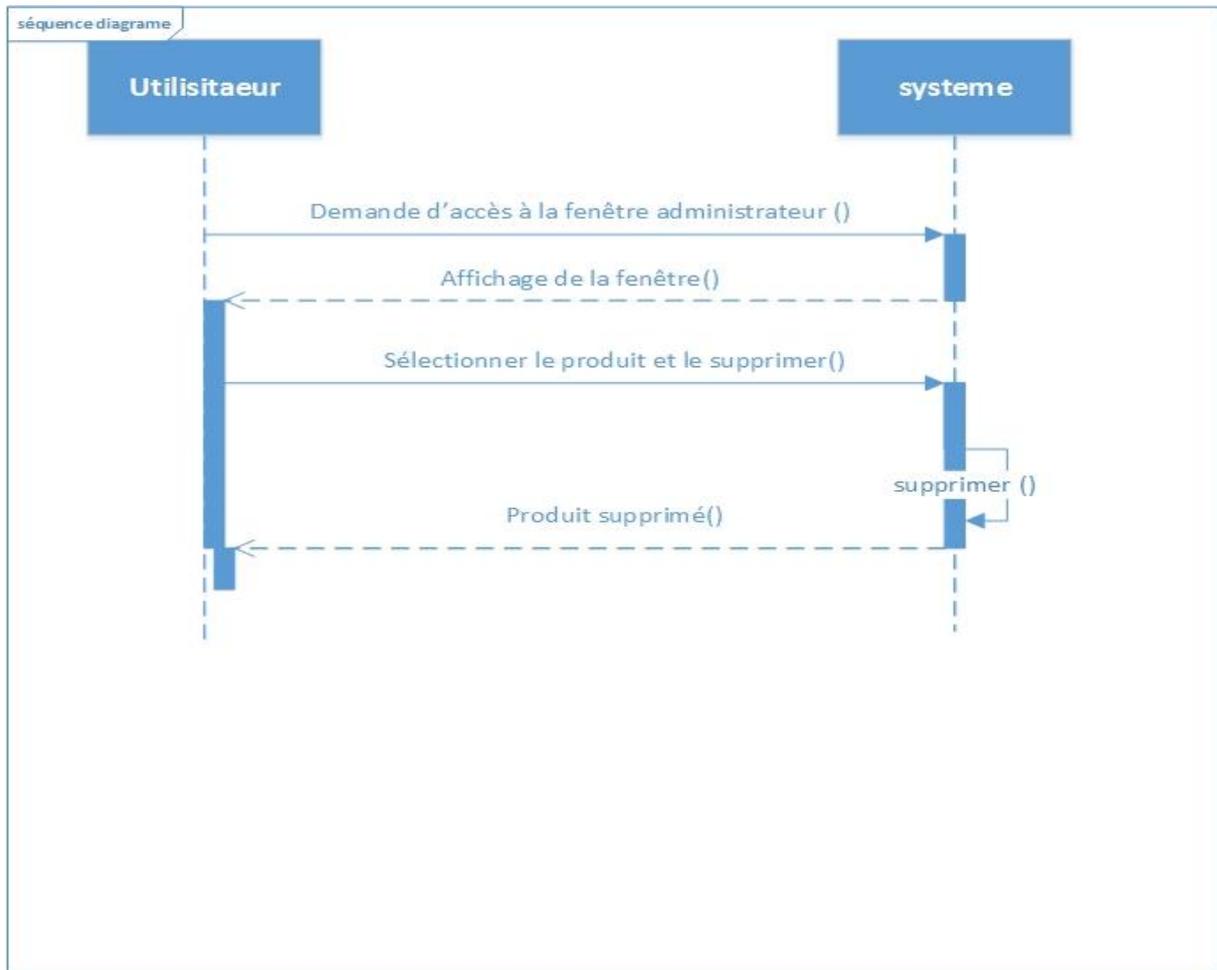
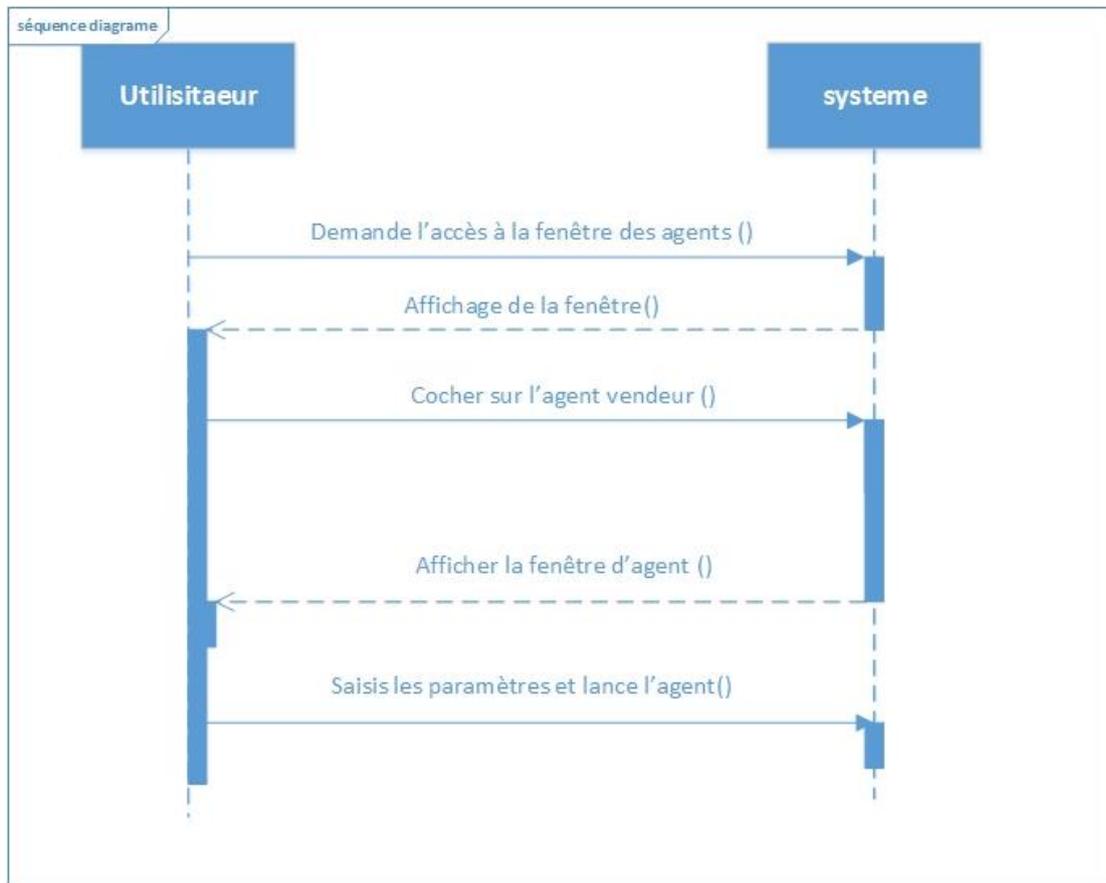


Figure 14 : diagramme de séquence -suppression client

Ce diagramme montre les étapes à suivre pour la fonction supprimer un produit, cette fonction ne peut être effectuée que par l'administrateur :

1. l'administrateur demande un accès.
2. le système affiche à l'administrateur la fenêtre administrateur.
3. l'administrateur sélectionne le produit à supprimer.
4. un pop-up apparaît pour informer que la suppression est effectuée avec succès.

### d) La création de l'agent vendeur

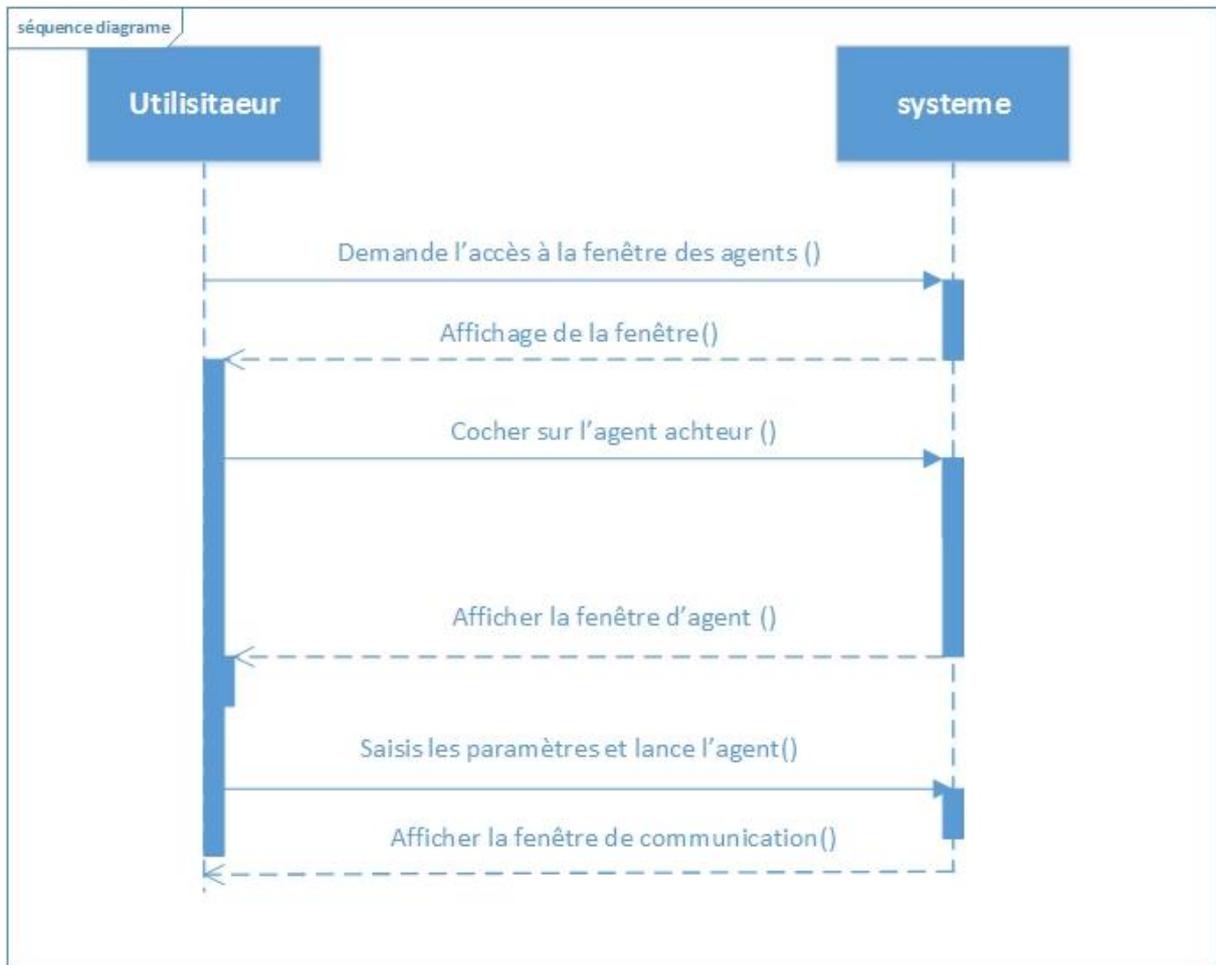


**Figure 15 : diagramme de séquence - création agent vendeur**

Le diagramme en haut montre les étapes à suivre pour la fonction création d'agent vendeur:

1. l'utilisateur demande un accès aux fonctionnalités de l'administrateur.
2. le système affiche à l'utilisateur la fenêtre de la fonction création d'agent vendeur.
3. l'utilisateur coche le type vendeur.
4. le système affiche à l'utilisateur la fenêtre de l'agent sélectionné.
5. l'utilisateur doit saisir les paramètres de l'agent à créer.

### e) La création de l'agent acheteur



**Figure 16 : diagramme de séquence - création agent acheteur**

1. l'utilisateur demande un accès aux fonctionnalités de l'administrateur.
2. le système affiche à l'utilisateur la fenêtre de la fonction création d'agent acheteur.
3. l'utilisateur coche le type acheteur.
4. le système affiche à l'utilisateur la fenêtre de l'agent sélectionné.
5. l'utilisateur doit saisir les paramètres de l'agent à créer.
6. le système affiche une fenêtre de communication pour voir les offres.

## f) L'appel d'offre (CFP)

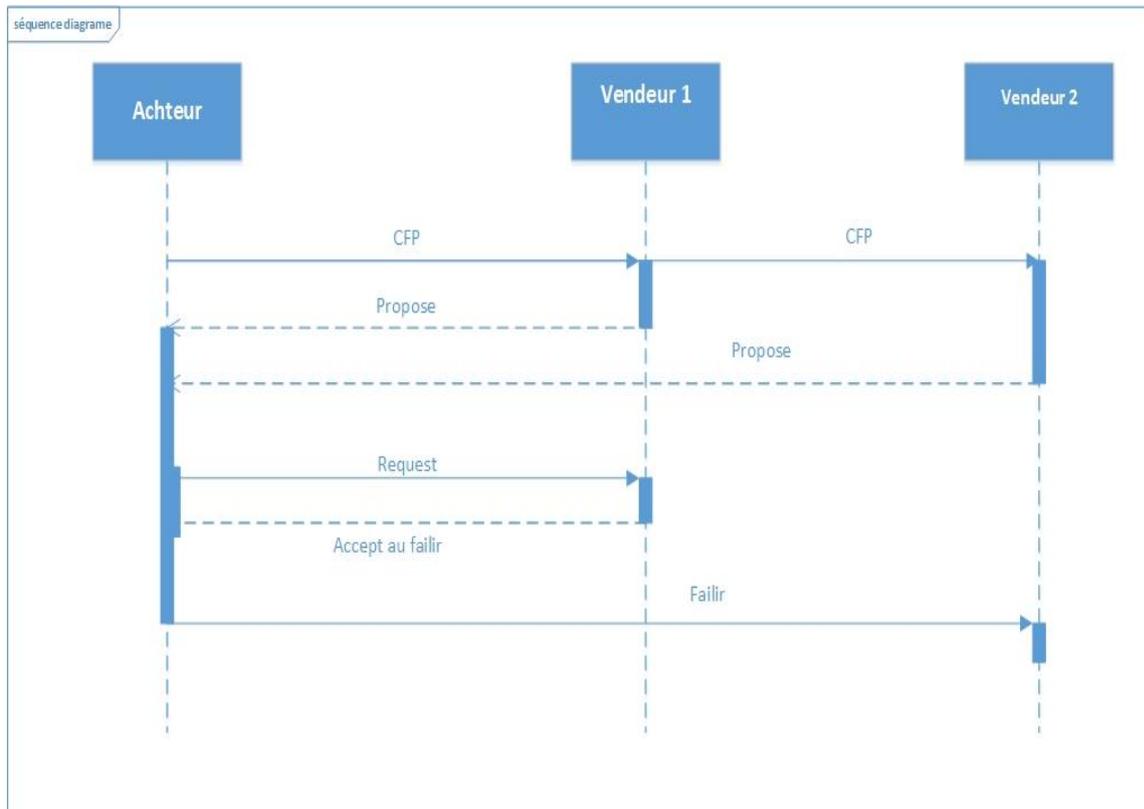


Figure 17: diagramme de séquence - appel d'offre

Nous expliquons le processus de l'appel d'offre effectué par un acheteur pour vendeur1 et vendeur2 :

**1 et 2** : l'acheteur envoie un CFP aux deux vendeurs.

**3 et 4** : les deux vendeurs vendeur1 et vendeur2 répondent au CFP envoyé par l'acheteur par des propositions (des messages de type propose), prenons le cas où le vendeur1 a mis la meilleure offre.

**5** l'acheteur envoie un message de type Request pour l'agent (vendeur1) avec la meilleure offre.

**6** le vendeur1 répond par un message de type :

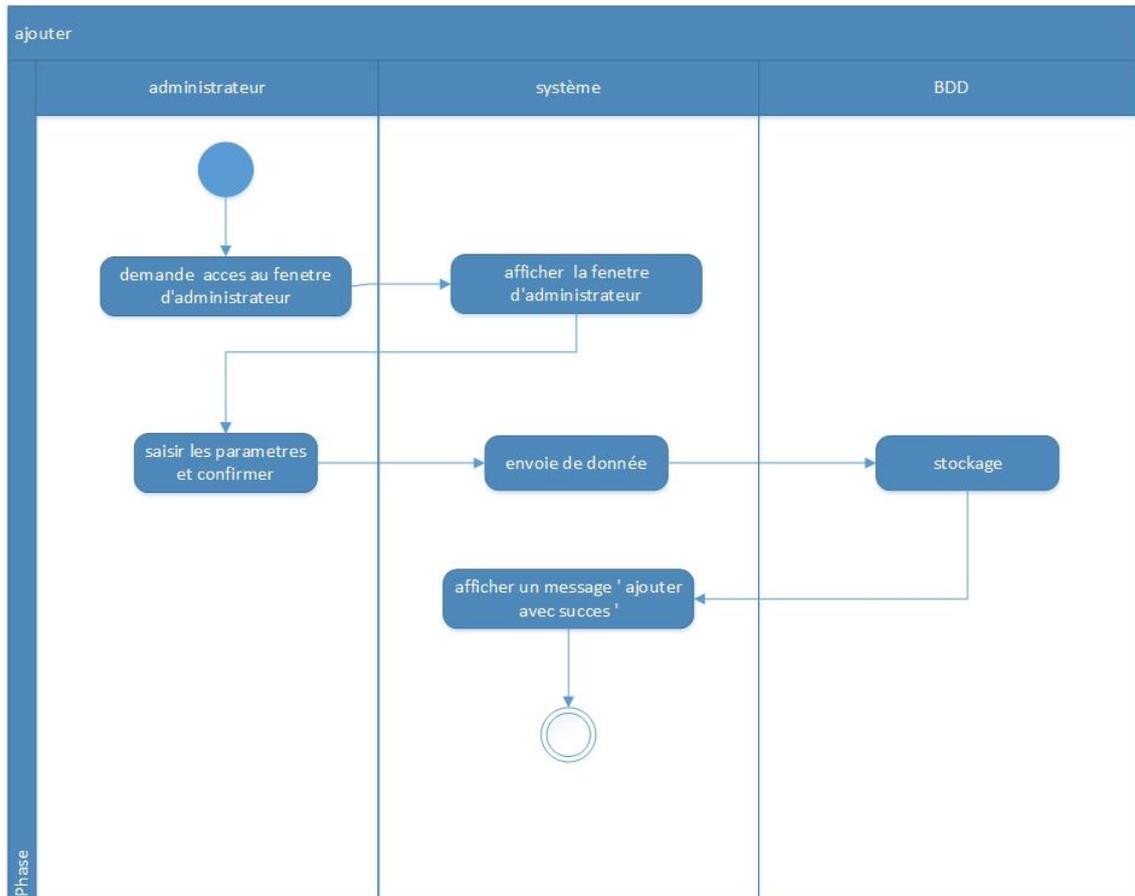
- \* accept dans le cas normal.

- \* faillure : dans le cas de stock insuffisant.

**-7** l'acheteur envoie un message de type failure au vendeur2.

### 3. Diagramme d'activité

### a) Ajout d'un produit



**Figure 18 : diagramme d'activité - ajout client**

Le diagramme d'activité d'authentification nous permet de voir les comportements internes du système, il nous permet de voir qui est responsable de chaque activité.

Dans ce cas qui est « l'ajout d'un produit », l'administrateur demande l'accès au système, ce dernier lui affiche la fenêtre correspondante, il remplit les informations et confirme l'ajout. Ces informations seront enregistrées au niveau de la BDD et un message de succès de l'opération est affiché.

## b) Modifier un produit

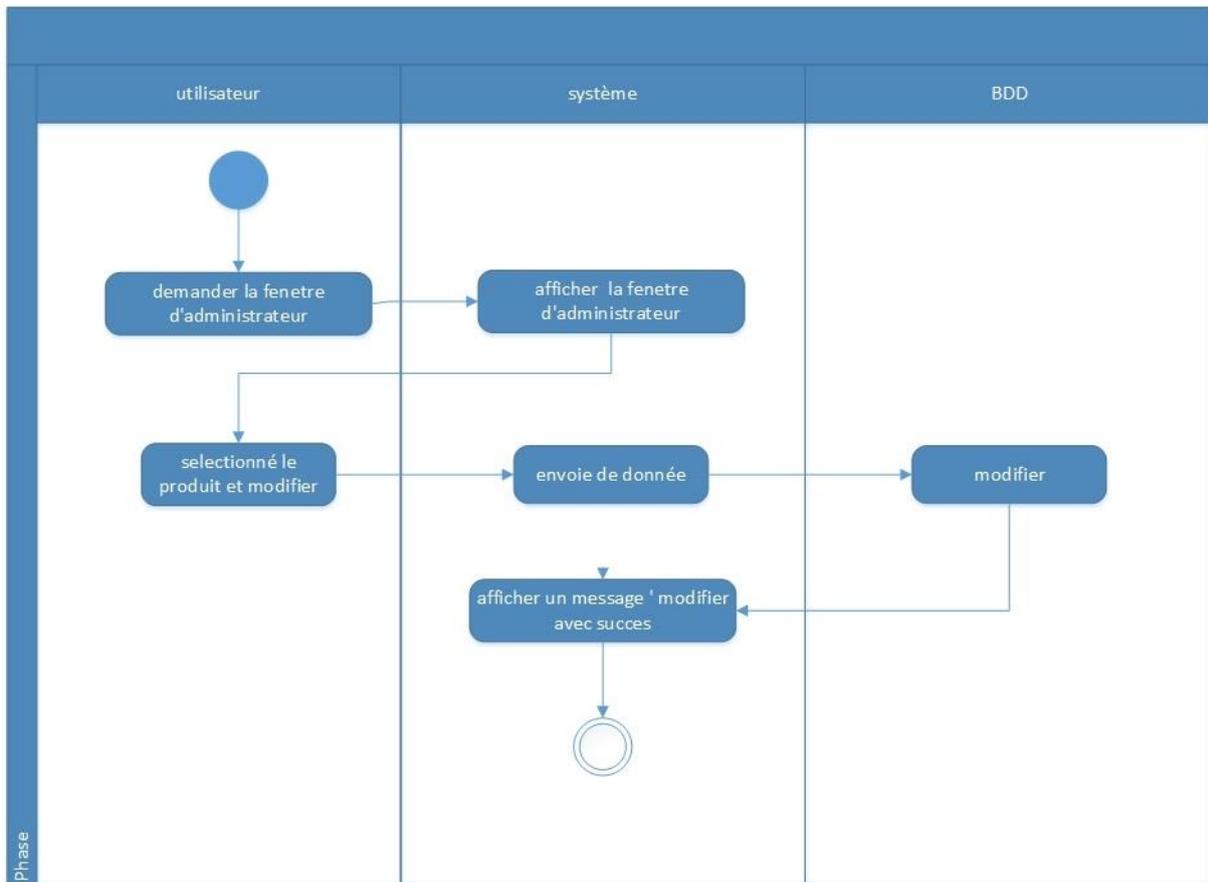


Figure 19: diagramme d'activité - modifier client

- Nous suivons les mêmes procédures que l'ajout.

### c) Supprimer un produit

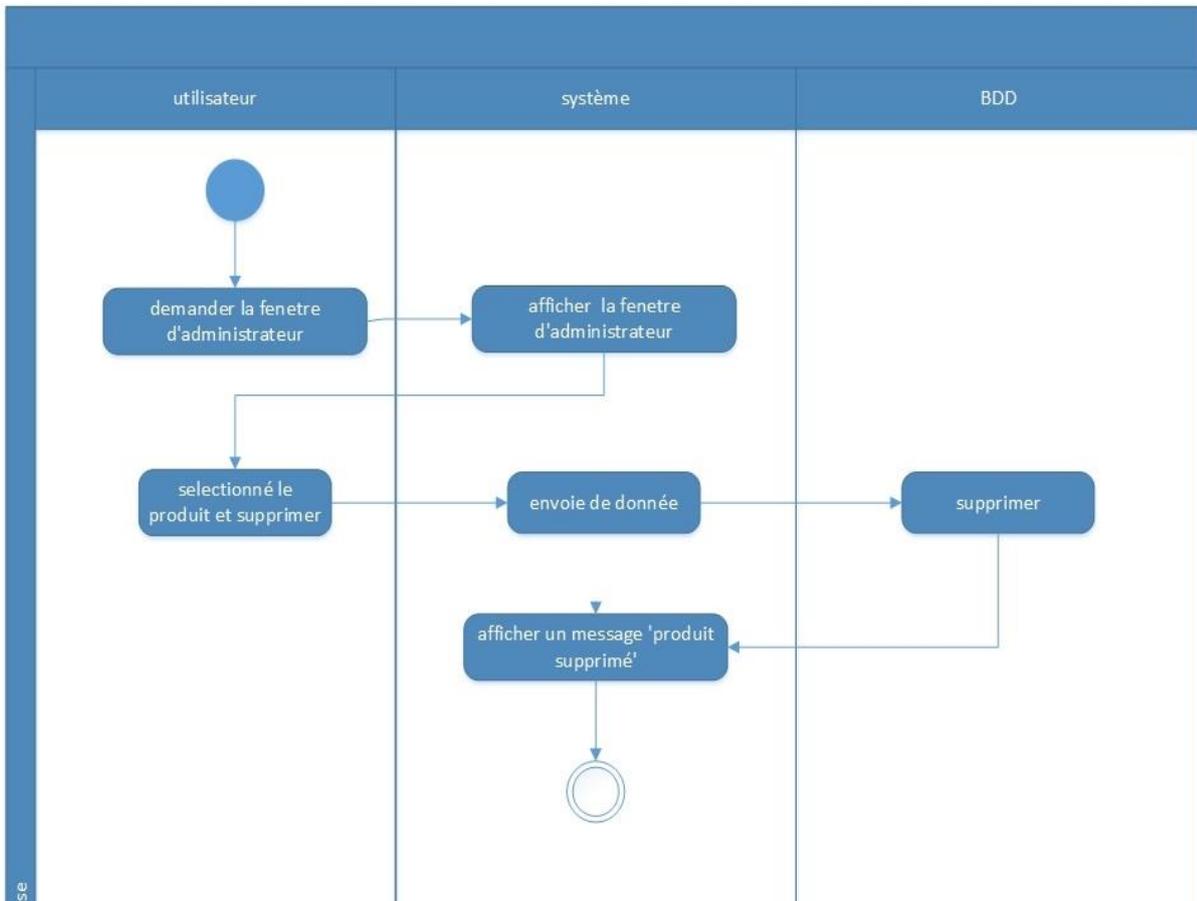


Figure 20 : diagramme d'activité -suppression client

- Les étapes de Cette opération ressemblent à celle de l'ajout.

### d) La création d'un agent acheteur

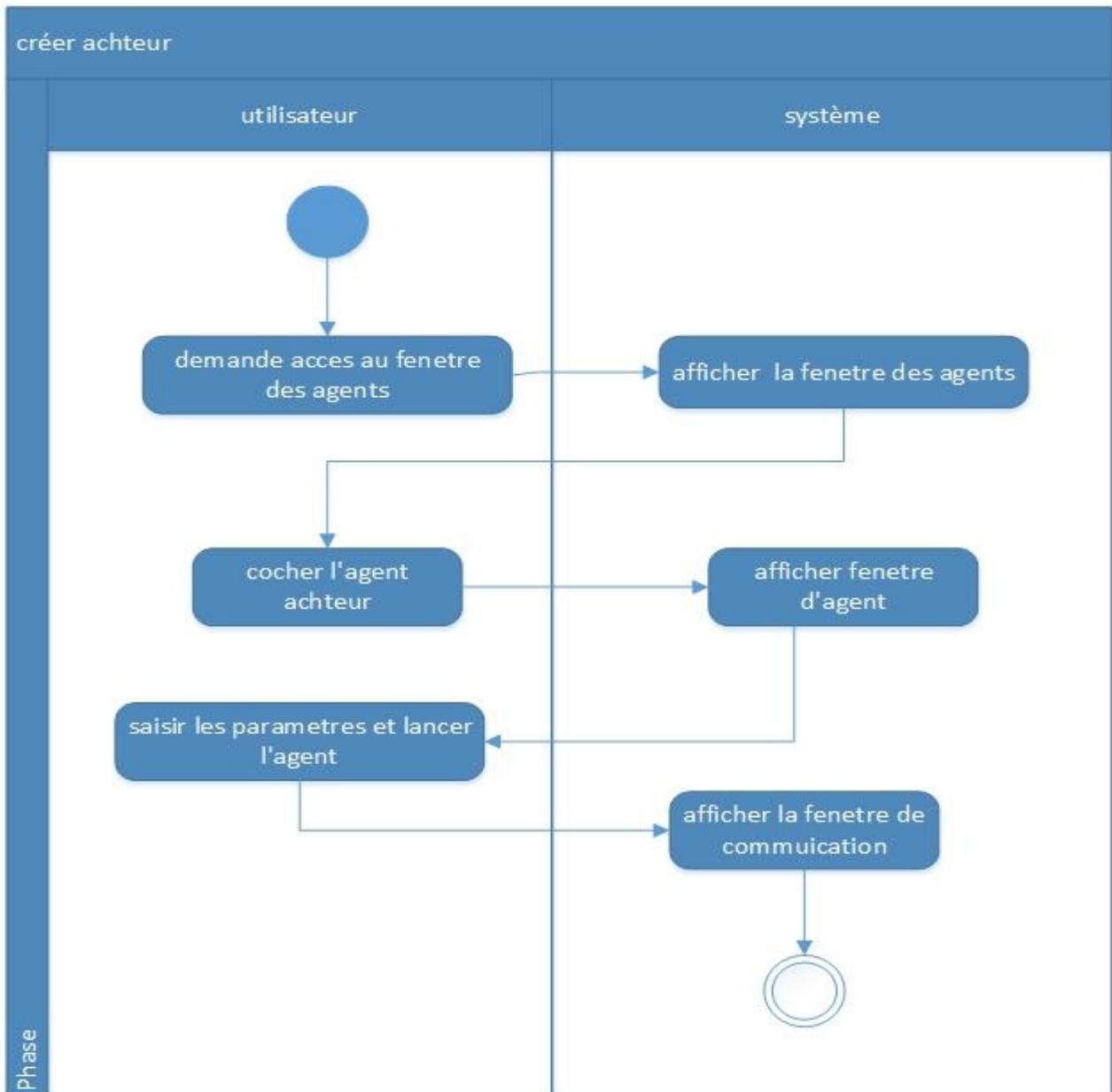


Figure 21 : diagramme d'activité -création agent acheteur

- Dans la fonction « création agent » l'utilisateur demande l'accès à la fenêtre création d'agent du système, ce dernier lui affiche la fenêtre correspondante, l'utilisateur coche la case « acheteur » et clique sur suivant, le système lui affiche une autre fenêtre pour remplir les paramètres de cet agent, puis il confirme la création et lance l'agent.

### e) La création d'un agent vendeur

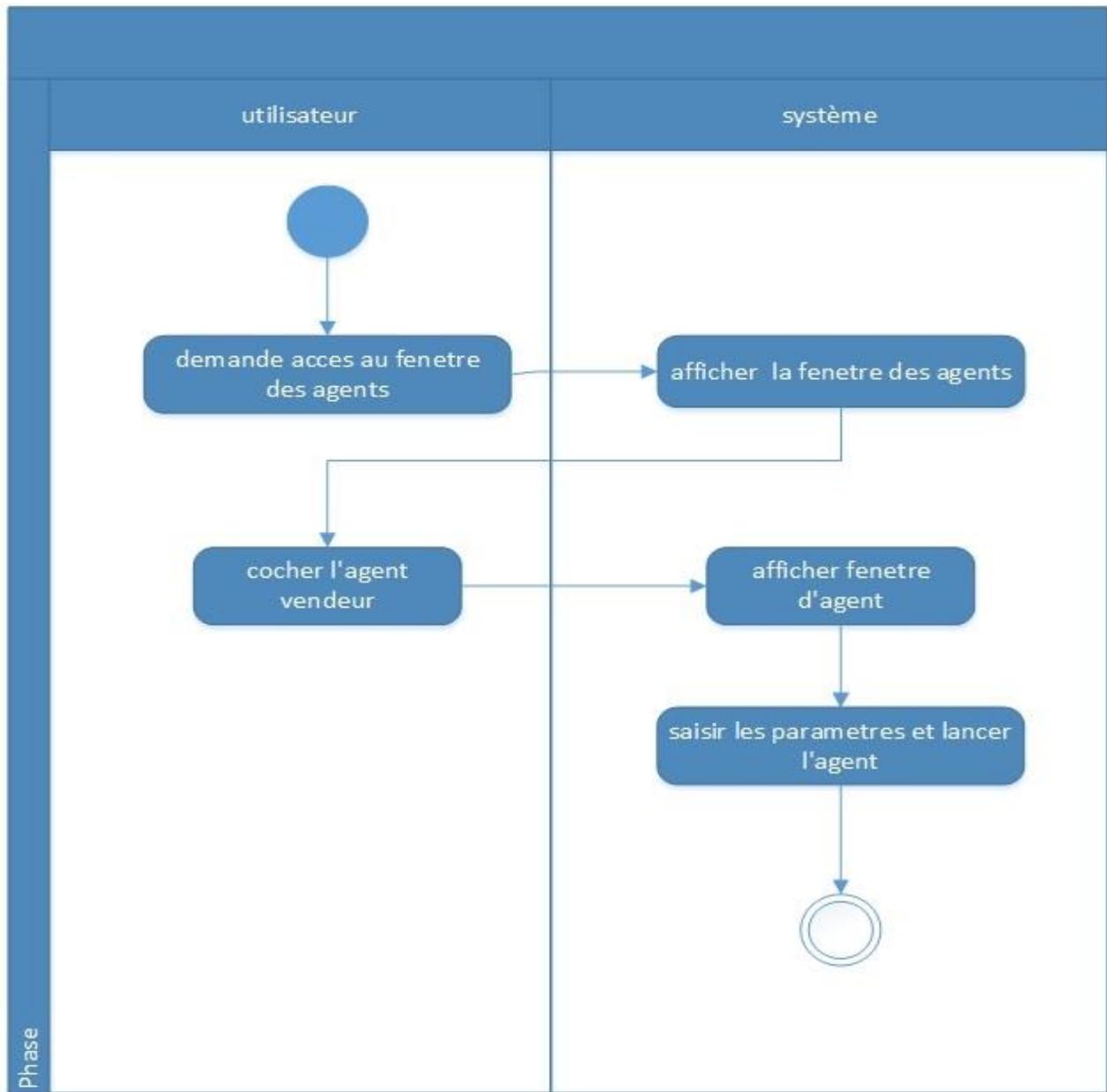


Figure 22 : diagramme d'activité -création agent vendeur

- Dans la fonction « création agent » l'utilisateur demande l'accès à la fenêtre création d'agent du système, ce dernier lui affiche la fenêtre correspondante, l'utilisateur coche la case « vendeur » et clique sur suivant, le système lui affiche une autre fenêtre pour remplir les paramètres de cet agent vendeur, puis il confirme la création et lance l'agent.

# Chapitre

---

## **IV** L'implémentation

## JADE

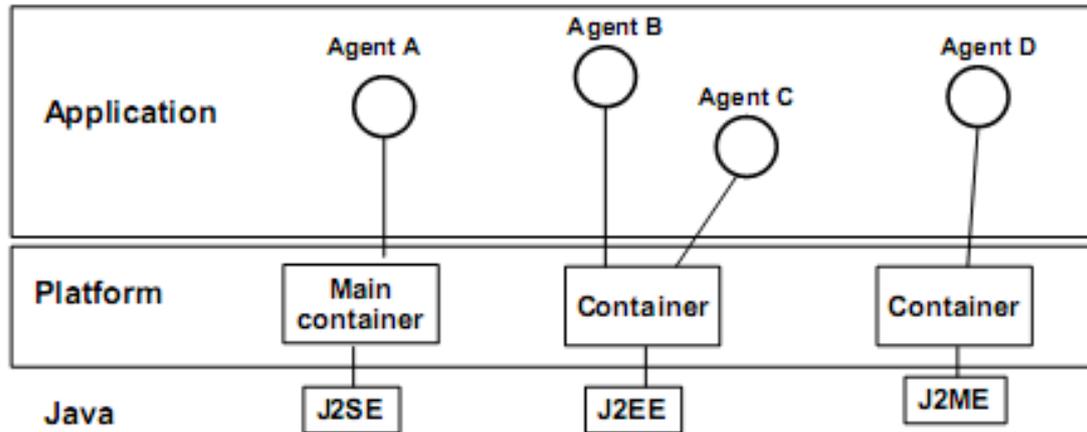
### 1. Définition

JADE est une plate forme multi agents développée en entier en JAVA, par F. Bellifemine & A. Pogy, G. Rimassa, P. Turci pour la société CSELT (Italie) en 1999, qui a comme but de faciliter le développement des applications agent conformément à la norme FIPA97 (Foundation for Intelligent Physical Agents) pour réaliser les systèmes multi agents interopérables [1].

Pour atteindre ce but, JADE offre les caractéristiques suivantes :

- Une plate forme multi-agents compatible FIPA, elle possède trois modules principaux qui incluent le AMS, DF, ACC, voir ci dessous. Ces trois agents sont automatiquement créés et activés à l'activation de la plate-forme.
- Une plate forme d'agents distribuée. JADE peut être distribuée sur plusieurs machines, à condition qu'il n'y ait pas de pare-feu entre ces machines. Une seule application Java (Machine Virtuelle Java) est exécutée sur chaque machine. Les agents sont implémentés comme des threads d'exécution Java et les événements Java sont utilisés pour la communication efficace et légère entre agents sur une même machine. Un agent peut exécuter des tâches parallèles et JADE planifie ces tâches d'une manière plus efficace (et même plus simple pour le programmeur) que la planification faite par la Machine Virtuelle Java pour les threads d'exécution.
- Un certain nombre de DF (Facilitateurs d'Annuaire) compatibles FIPA peuvent être activés quand on lance la plate forme pour exécuter les applications multi-domaines, où le domaine est logique comme décrit dans FIPA97 Part1.
- Une interface de programmation pour simplifier l'enregistrement de services d'agents avec un ou plusieurs domaines (exemple : DF).
- Un mécanisme de transport et une interface pour l'envoi et la réception des messages de et vers les autres agents.
- Le protocole IIOP compatible avec le document FIPA97 pour connecter les différentes plates-formes multi agents.
- Le transport léger de messages ACL sur la même plate forme d'agents. Dans le but de simplifier la transmission, les messages internes (sur la même plate forme) sont transférés et codés comme des objets Java et non comme des chaînes de caractères. Quand l'expéditeur ou le récepteur n'appartient pas à la même plate forme, le message est automatiquement converti en chaîne de caractères spécifiée par la FIPA. De cette façon, la conversion est cachée au programmeur d'agents, qui a seulement besoin de traiter la classe d'objets Java.
- Une bibliothèque de protocoles d'interaction compatibles FIPA.
- L'enregistrement automatique d'agents dans le Système de Gestion d'Agents (AMS).

- Un service d'attribution de noms compatibles FIPA ; quand on lance la plate forme, un agent obtient un identificateur unique (Globally Unique Identifier - GUID).
- Une interface graphique utilisateur pour gérer plusieurs agents et plate formes multi agents en partant d'un agent unique. L'activité de chaque plateforme peut être supervisée et enregistrée. [46]



## Structure de la plate forme JADE

### 1.1. Conteneurs

Un conteneur est un environnement d'exécution JADE :

- Environnement complet d'exécution pour un agent.
- Exécution concurrente de plusieurs agents.
- Contrôle le cycle de vie des agents.
- Assure la communication entre agents.

Un seul conteneur héberge l'AMS, le DF, c'est le conteneur principal (main container) :

- AMS (Agent Management System)
  - Service de Pages Blanches : référence automatiquement les agents suivant leurs noms dès leurs entrées dans le système.
- DF (Directory Facilitator)
  - Service de Pages Jaunes : référence à leurs demandes les agents suivant leur(s) service(s).

Le conteneur principal peut être répliqué via des services de réplification. [47]

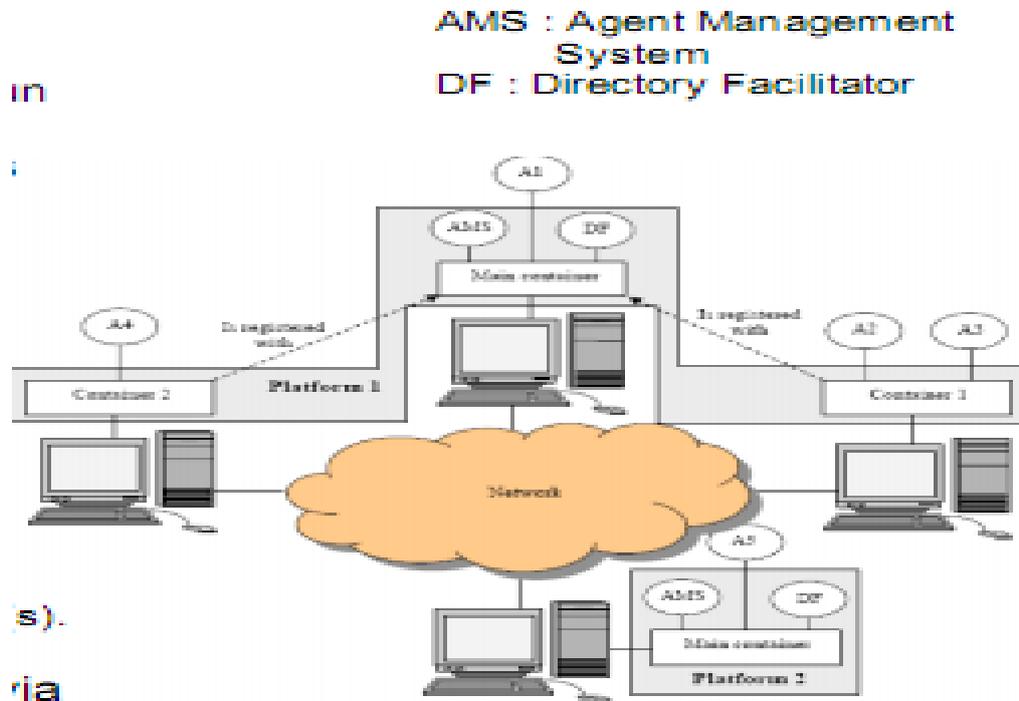


Figure 23 : Conteneurs

## 1.2. Outils utiles au débogage

- Dummy Agent
  - Visualisation des messages
  - Envoie des messages aux agents et réceptionne leurs réponses, présents sur la plateforme
  
- Sniffer Agent
  - Visualisation de l'enchaînement des messages et des messages eux même.
  - Vérification interactive de la correction des protocoles.
  
- Introspector Agent
  - Visualisation des messages envoyés/reçus,
  - Visualisation des Behaviours actifs et non actifs,
  - Contrôle de l'état de l'agent. [47] actifs,

### 1.3. Architecture de la plateforme Jade

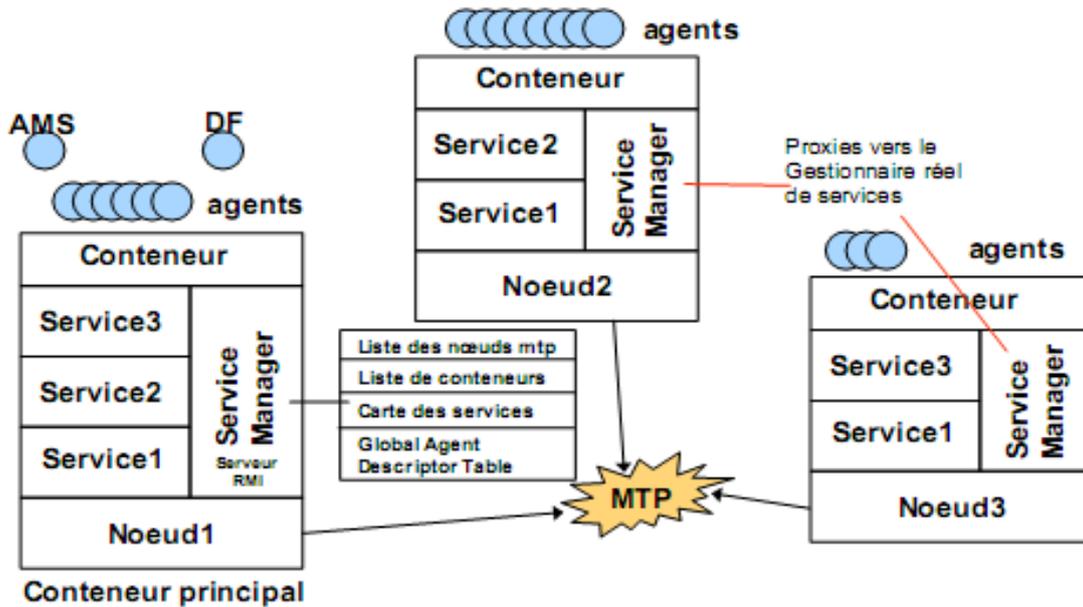


Figure 24 : Architecture de la plateforme Jade

### 1.4. Services Automatiques

Messaging : échange des messages en ACL et gestion du Message Transfert Protocol

- jade.core.messaging.MessagingService

Agent-Management : gestion du cycle de vie de l'agent, support au RMA, arrêt plate forme et conteneur.

- jade.core.management.AgentManagementService

Agent-Mobility : support à la mobilité des agents.

- jade.core.mobility.AgentMobilityService

Notification : notifications d'événements du niveau plate forme (requis pour le Sniffer et l'Introspector).

- jade.core.event.NotificationService

**Remarques :** par défaut, ces services sont activés, si d'autres services sont activés, alors Agent-Mobility et Notification doivent être activés explicitement.[47]

## 1.5. Agents donnant accès aux services

AMS : Agent Management System.

- Gestion des pages blanches.
- Gestion du cycle de vie des agents sur la plate forme.

DF: Directory Facilitator.

- Gestion des pages jaunes.
- Association entre description de services proposés et agents.
- Gestion des descriptions de services proposés par les agents.

Outils de débogage :

- Introspector, Sniffer, Dummy Agent. [47]

### 1. WampServer :

WampServer [48] est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL

La version 2.4.9 utilisé dans ce projet intègre Apache 2.4.17, MySQL 5.7.9, PHP 5.6.16 et PHP 7, PhpMyadmin 4.6.2, SQLBuddy 1.3.3, XDebug 2.2.3.



Figure 25 WampServer Logo

### 2. Connecteur JDBC (Java Database Connectivity) MySQL :

JDBC est une interface de programmation créée par Sun Microsystems depuis racheté par Oracle Corporation, pour les programmes utilisant la plate forme Java. Elle permet aux applications Java d'accéder par le biais d'une interface commune à des sources de données pour lesquelles il existe des pilotes JDBC. Normalement, il s'agit d'une base de données relationnelle, et des pilotes JDBC sont disponibles pour tous les systèmes connus de bases de données relationnelles [49].

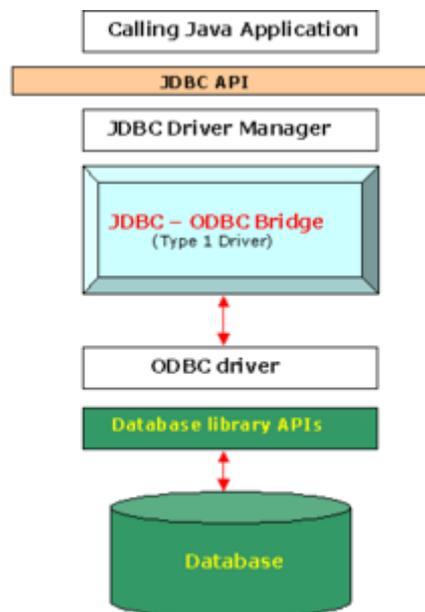


Figure 26 JDBC Connector

Dès le lancement du système deux fenêtres apparaissent, figure 27, la fenêtre de l'accueil de l'application et Figure 28, la fenêtre SNIFER qui montre les échanges de communication entre les agents existants.

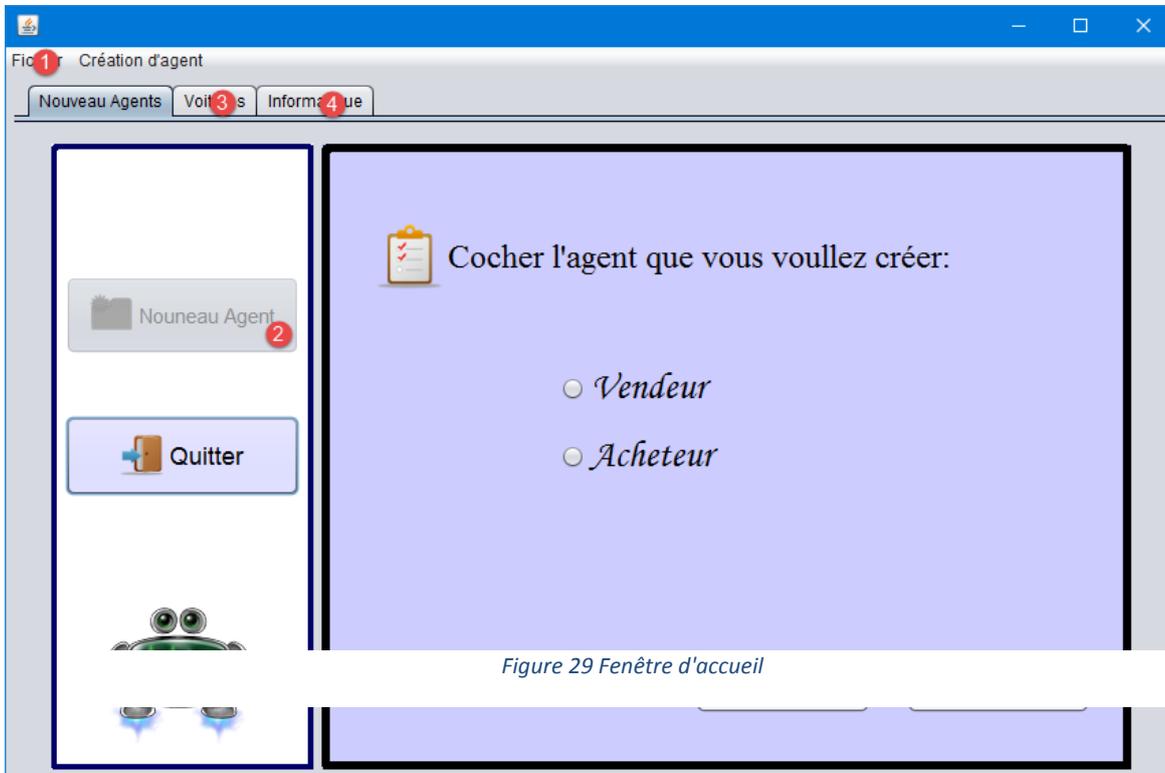


Figure 29 Fenêtre d'accueil

Figure 27: fenêtre principale

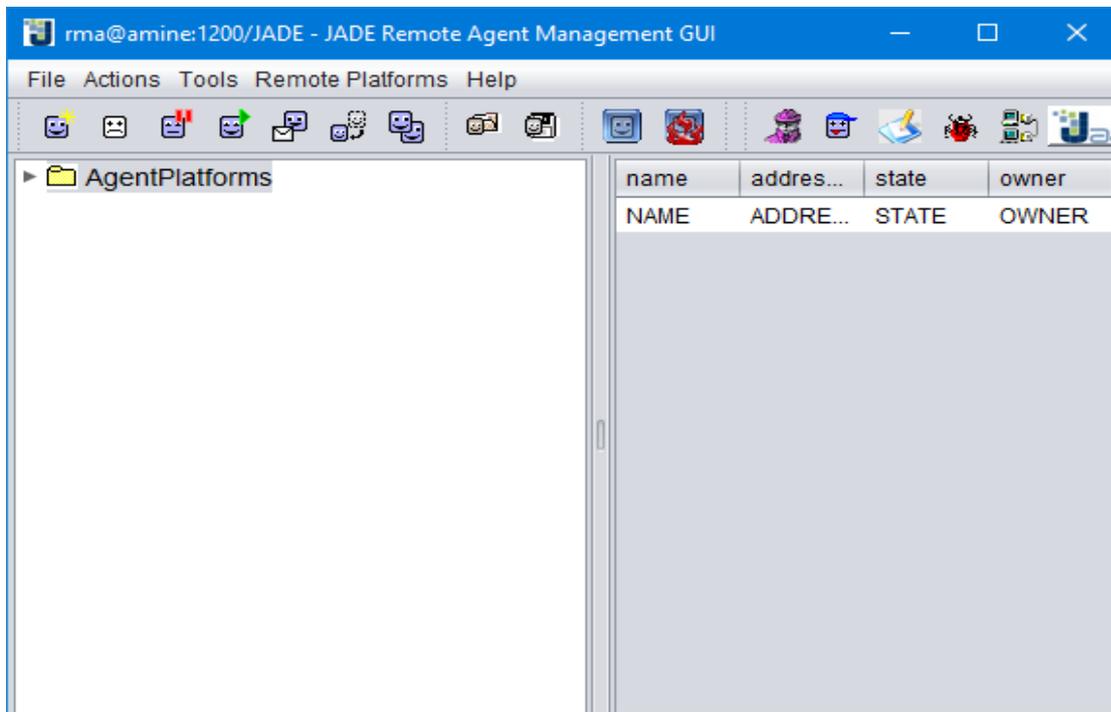


Figure 28 : Fenêtre SNIFER

Quand on appuie sur le bouton ① une fenêtre de gestion administrative des produits s’affiche figure #. On aura la possibilité d’ajouter de nouveaux produits et de modifier et supprimer des produits existants.

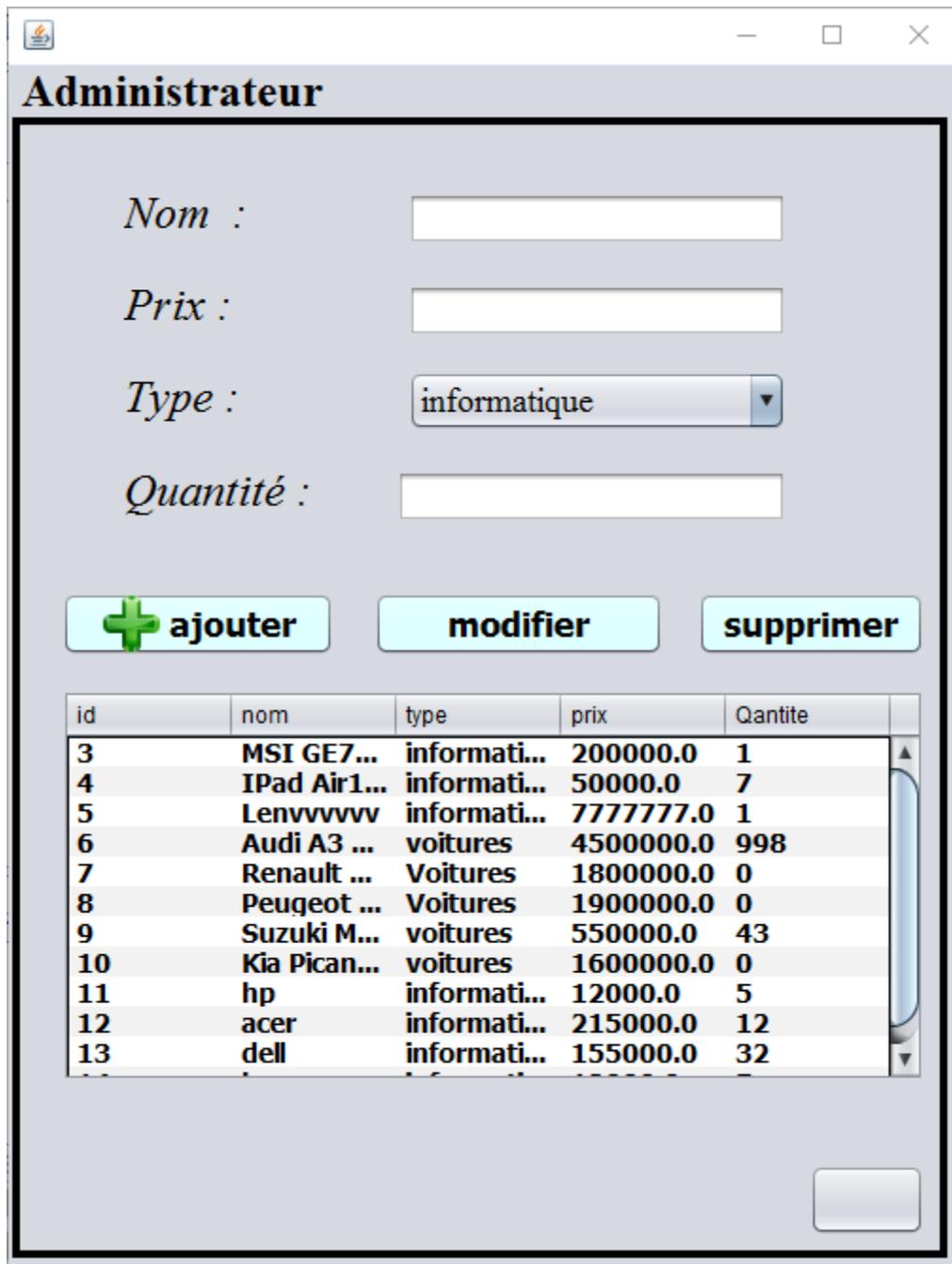
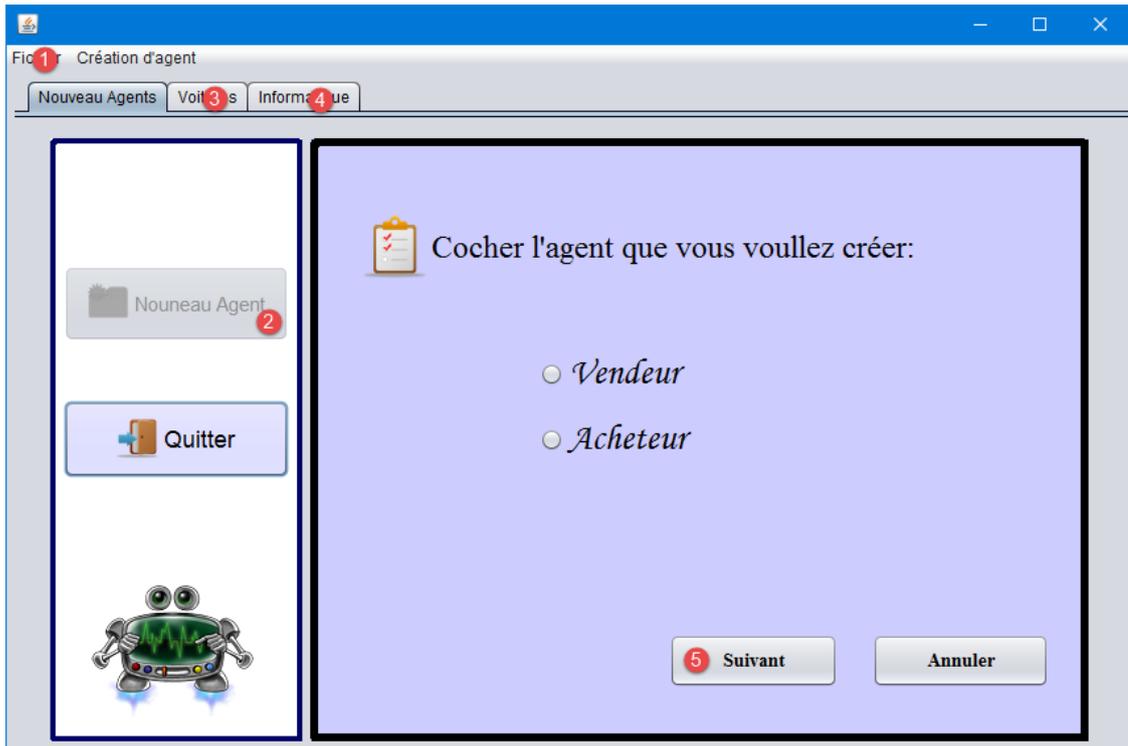


Figure 30 : fenêtre administrateur



② mène à la fenêtre figure # qui permet de choisir le type d'acteurs qui vont intervenir dans le processus de l'achat/vente. Ces acteurs sont des agents et peuvent être vendeur ou acheteur.

Après avoir validé le choix de l'utilisateur en appuyant sur ⑥ la fenêtre d'ajout d'agent dont le type est coché (vendeur / acheteur) figure## apparait.

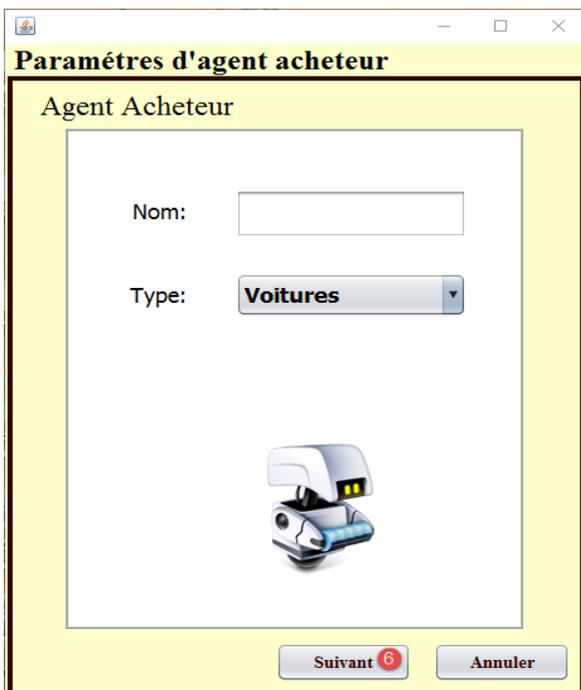


Figure 31 : fenêtres création des agents

En appuyant sur ③ ou ④ une fenêtre contenant la liste d'agent existant par type s'affiche.

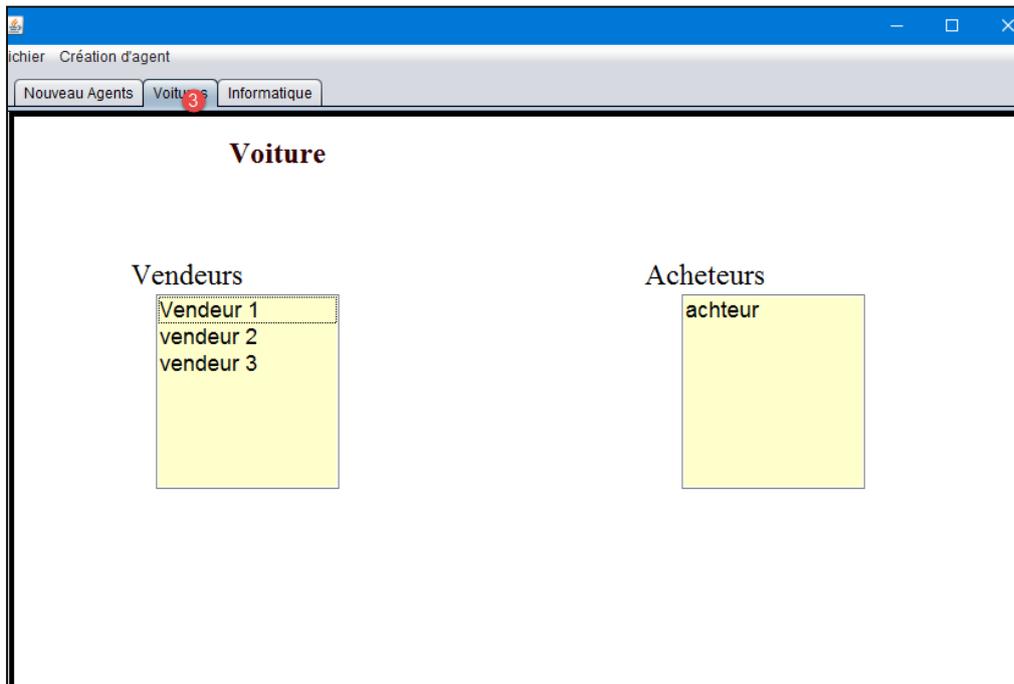


Figure 32: fenêtre d'affichage de type voiture

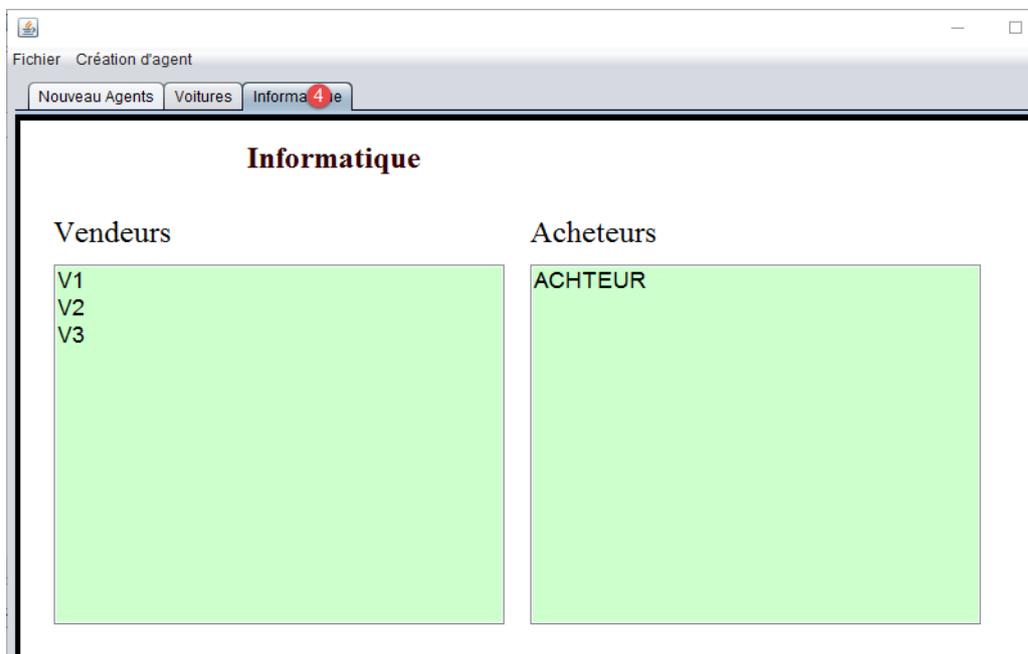


Figure 33:fenêtre d'affichage de type informatique

Après le lancement de l'agent acheteur en appuyant sur ⑥ la fenêtre figure# s'affiche

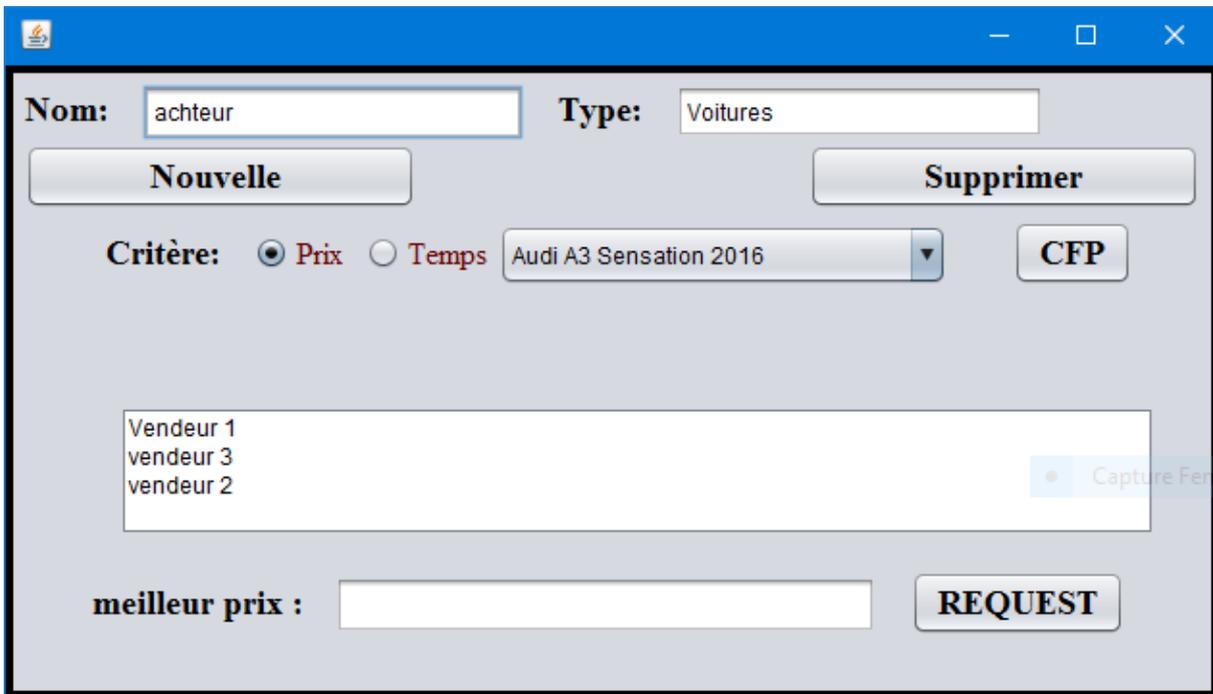


Figure 34 : fenêtre de actualisation des agents

montrant la mise à jour des agents. Après clique sur le bouton ⑦ un appel d'offre est lancé par l'agent acheteur et la liste des intervenants dans le CFP s'affiche. Un clique sur le bouton 8 ⑧ confirme l'accord.

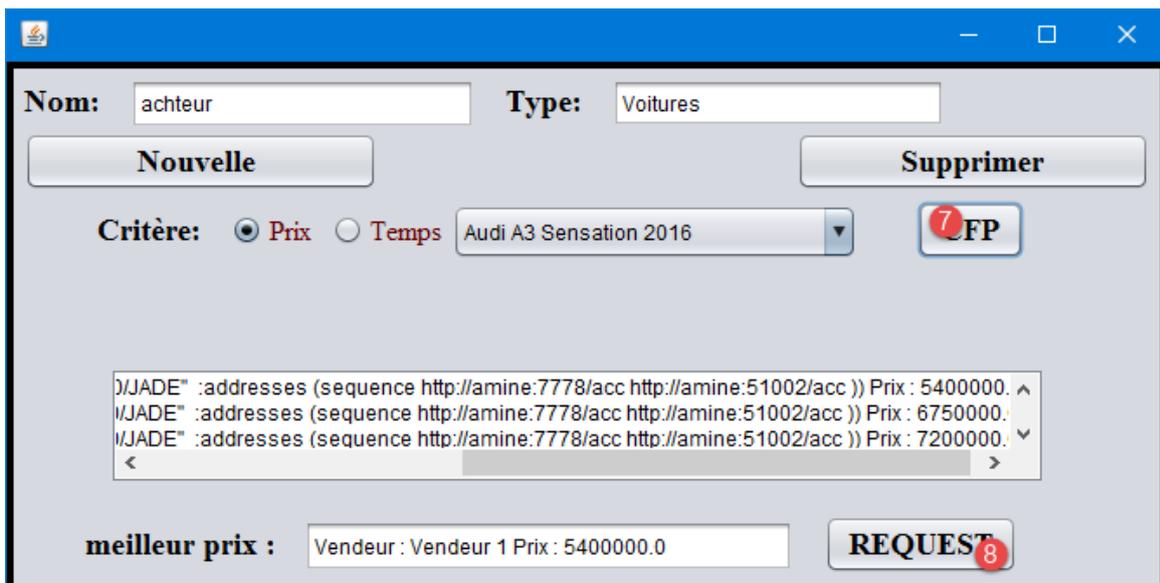


Figure 35 : Fenêtre communication entre les agents

Puis après cliquer sur ⑨ la boîte de dialogue informant le succès de l'opération figure # s'affiche.

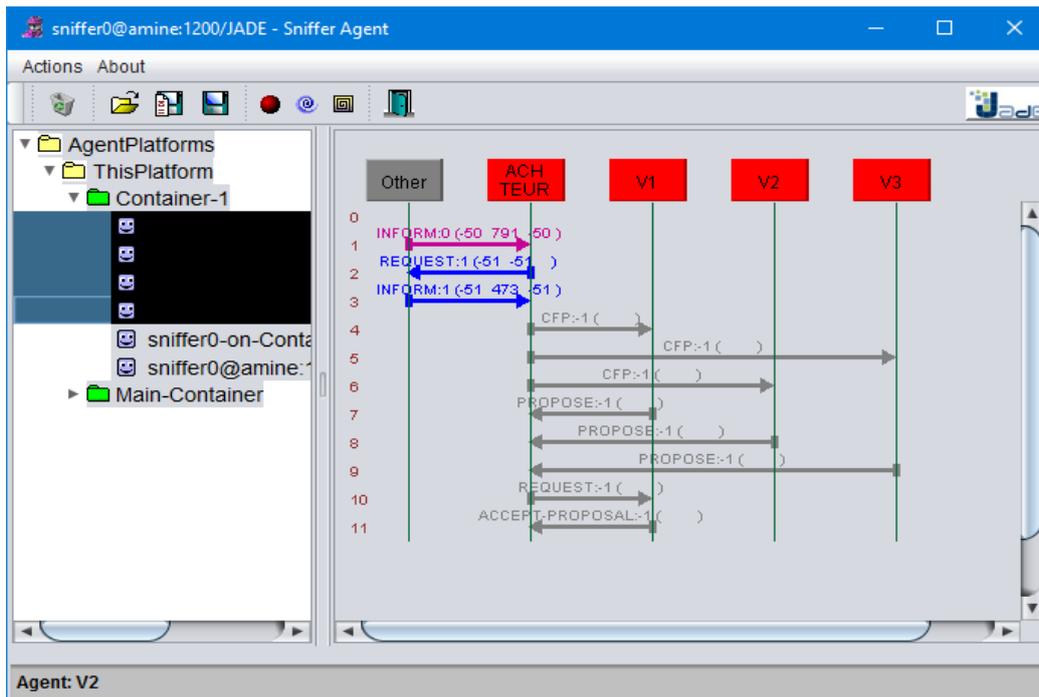
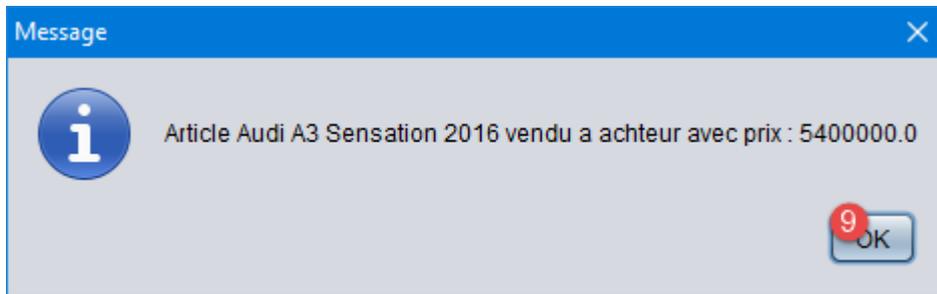


Figure 36: sniffer de confirmation

Dans le cas où la quantité est insuffisante le pop-up figure # apparait la figure# et le SNIFER s'affiche.

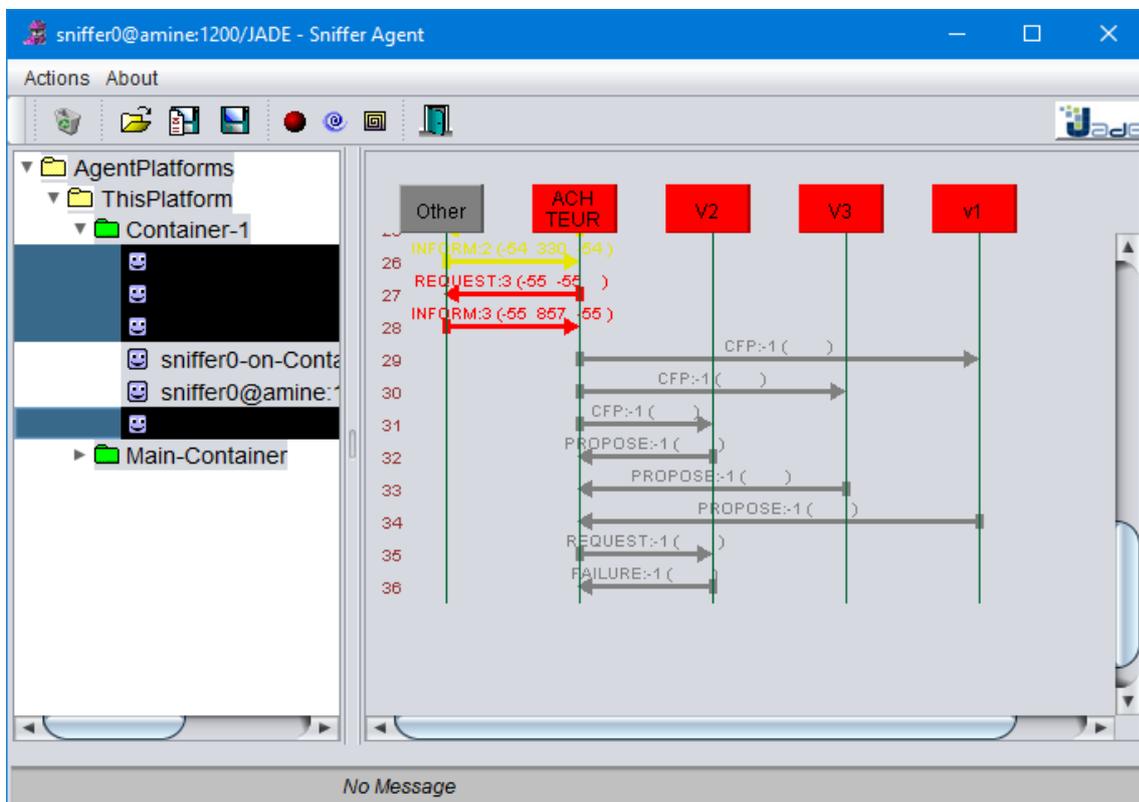
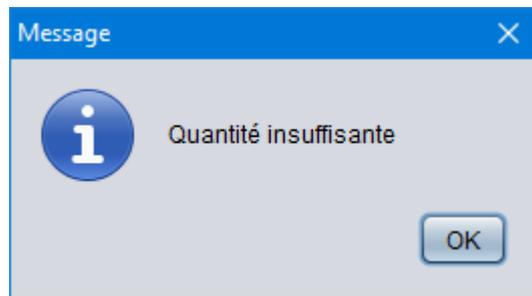


Figure 37: sniffer de refus

## Conclusion générale

La communication entre agents est primordiale pour les systèmes multi agents. Les agents doivent communiquer pour se coordonner et échanger de l'information, ils peuvent donc échanger des connaissances pour déterminer leurs actions en fonction du comportement des autres agents du système.

Les systèmes multi agents sont devenus un paradigme dominant dans le domaine de développement des systèmes distribués complexes grâce à leurs caractéristiques de répartition et d'hétérogénéité des données, de décentralisation du contrôle, d'asynchronisation des traitements et de la vision partielle de chaque composant de ces systèmes, aussi les mécanismes de coordination et de communication préconisés par l'approche agent fournissent des solutions satisfaisantes et élégantes.

Le travail réalisé dans ce mémoire nous a permis de comprendre les concepts fondamentaux des agents et des systèmes multi agents ainsi que leur utilité dans la résolution des problèmes à travers l'application sur un marché électronique.

Il nous semble que l'intérêt des SMA est que les problèmes complexes sont résolus d'une manière ascendante telle qu'ils se déroulent réellement, sans pour autant chercher des solutions complexes. En d'autres termes, si les résultats d'un phénomène n'ont pas de lois pour les calculer, alors on cherche plutôt ce qui produit ce phénomène.

Durant ce travail nous avons constaté que les agents sont généralement utilisés dans un contexte de SMA, de ce fait, il nous semble qu'il sera utile de concevoir des plates formes orientées beaucoup plus SMA qu'agent. En d'autres termes, le point clé dans ces plates formes est l'agent alors qu'il n'est pas – généralement – dépourvu d'un contexte de SMA.

En outre, nous pouvons penser à concevoir des prototypes de SMA, vu que plusieurs problèmes peuvent être résolus pratiquement par le même modèle d'agents.

Aussi, la spécialisation d'une plate forme dans un type particulier de classes de problèmes permet d'améliorer ses performances, par exemple des SMA purement communicatifs.

Les perspectives ouvertes suite à notre travail sont :

- L'enrichissement du modèle par d'autres règles comportementales.
- La distribution de l'application.

### Référence

- [1] Lejouad, W., & Labidi, S. (1993). De l'Intelligence Artificielle Distribuée aux Systèmes Multi-Agents. Rapport de recherche, Sophia-Antipolis: INRIA.
- [2] 51 Nicolle, A. (2002). Les systèmes multi-agents. Maîtrise d<sup>TM</sup> informatique MI6.
- [3] 14 Ferber, J. (1995). Les systèmes multi-agents: vers une intelligence collective. Informatique. Intelligence Artificielle. InterÉditions.
- [4] 54 Fourdrinoy, O. (2002). Création d'une librairie de Balises pour l'insertion d'agents JADE dans des pages Jsp. Support de cours, Lens: Faculté Jean Perrin, Université d'Artois.
- [5] 53 Quinqueton, J. (2005). Introduction aux systèmes multi-agent (SMA). Support de cours, Montpellier: LIRMM et CERIC.
- [6] 55 Florea, A., Kayser, D., & Pentiu, S. (2002). Agents Intelligents. Support de cours, Bucharest: Politechnica University of Bucharest.
- [7] 56 Iffenger, C. (1992). Un système multi-agent pour le support des activités de conception de produit. Thèse de l'Université Paris VI.
- [8] 57 Chaib-Draa, B., Jarras, I., & Moulin, B. (2001). Systèmes multi-agents: principes généraux et applications. Edition Hermès.
- [9] 58 Finin, T., Fritzson, R., McKay, D., & McEntire, R. (1994). KQML as an agent communication language. In Proceedings of the third international conference on Information and knowledge management (pp. 456-463). ACM.
- [10] 59 Chaib-draa, B., Jarras, I., Moulin, B., (2001) .Systèmes multi agents : Principes généraux et applications », Briot J. P., Demazeau Y., Eds., Agent et systèmes multiagents, Hermès, Repitition
- [11] 60: Les systèmes multi-Agents :
- [http://perso.limsi.fr/jps/enseignement/examsma/2005/1.plateformes\\_3/index.html](http://perso.limsi.fr/jps/enseignement/examsma/2005/1.plateformes_3/index.html)
- [12] 62 Downing K, Zvirinsky P. (1999). The simulated evolution of biochemical guilds :Reconciling Gaia theory and natural selection. Artificial Life, 5(4).
- [13] 63 Drogoul A. (1993), De la simulation multi agents à la résolution collective de problèmes: une étude de l'émergence de structure d'organisation dans les systèmes multi agents, thèse de doctorat, Université de Paris VI.
- [14] 64 Duvallet C. (2001). Des systèmes d'aide à la décision temps réel et distribué : modélisation par agents. thèse de doctorat, Université du Havre.

## Références

---

- [15] Ghidini C, Serafini L. (1998). Information integration for electronic commerce. The 1st International Workshop on Agent Mediated Electronic Trading (AMET'98), Minneapolis, USA.
- [16] Guttman R, Maes P. (1998). Agent mediated interrogation negotiation for retail electronic commerce, The 1st International Workshop on Agent Mediated Electronic Trading (AMET'98), p. 70-90, Minneapolis, USA.
- [17] Greaser et al. Autotutor : A simulation of a human tutor. Journal of Cognitive Systems Research, 1999.
- [18] Nédélec, P. Reignier et V. Rodin. Collaborative prototyping in distributed virtual reality using an agent communication language. 2000.
- [19] Artificial Intelligence. <http://www.a-i.com>.
- [20] Chaib-draa et F. Dignum. Trends in Agent Communication Language. Computational Intelligence, 2002.
- [21] Grosz et Y. Labrou. An approach to using xml and a rule-based content language with an agent communication language. 2000.
- [22] Pelachaud, I. Poggi. Towards believable interactive embodied agents. 2001.
- [23] Rao et M. P. George. Modeling rational agents within a BDI-architecture. In Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91), 1991.
- [24] FIPA. Spécifications FIPA, 2000. <http://www.fipa.org>.
- [25] FIPA OS. <http://fipa-os.sourceforge.net>.
- [26] Chicoisne. Dialogue entre agents naturels et agents artificiels. PhD thesis, INPG, Grenoble, 2002.
- [27] Java Agent Development Framework. <http://jade.cselt.it>.
- [28] Java Expert System Shell. <http://herzberg.ca.sandia.gov/jess>.
- [29] Cassel, T. Bickmore et al. An architecture for embodied conversational characters. 1999.
- [30] Austin. Quand dire, C'est faire. Seuil, 1962.

## Références

---

- [31]16J.RickeletW.L.Johnson. Animated agents for procedural training in virtual reality. Applied Artificial Intelligence, 1999.
- [32]17J.R.Searle. Speech Acts. Cambridge University Press, 1969.
- [33]18J.Tisseau. Réalité virtuelle, autonomie in virtuo. 2001. Habilitation à Diriger des Recherches.
- [34]19J.Weizenbaum. A computer program for the study of natural language communication between man and machine. Communications of the ACM, 1966.
- [35]20K.R.Thorisson. Natural turn-taking needs no manual. Multimodality in Language and Speech Systems, 2002.
- [36]21M.L.Mauldin. Chatterbots, tinymuds, and the turingtest : Entering the loebner prize competition. 1994.
- [37]22M.P.Singh. Agent Communication Languages : Rethinking the Principles. 1998.
- [38]23M.Stone and C.Doran. Sentence planning as description using treedjoining grammar. Proceedings of ACL, 1997.
- [39]24Ontoligua. <http://www.ksl.stanford.edu/software/ontoligua>.
- [40]25P.Pasquier et B.Chaib-draa. Engagements, intentions et jeux de dialogue. In les Actes des Secondes Journées Francophones Modèles Formels de l'Interaction, Lille, 2003.
- [41]26T.Bickmore et J.Cassell. Social Dialogue with Embodied Conversational Agents. Natural, Intelligent and Effective Interaction with Multimodal Dialogue Systems, 2004.
- [42]27Y.Labrou. Semantics for an Agent Communication Language. PhD thesis, UMBC, 1996.
- [43]28Y.Labrou, T.Finin et Y.Peng. Agent communication languages : The current landscape. Intelligent Agents, 1999.
- [44] Émergence et tendances du commerce électronique Chapitre 1 2014 Pearson France – E-commerce, 3e édition – Henri Isaac, Pierre Volle
- [45] Institut Des Hautes Études Commerciales De SOUSSE Enseignant Responsable :BOUBAKER Nobel El Houssine Année Universitaire 2004-2005

## Références

---

[46] JADE (Développement et Implémentation de systèmes Multi-Agents) :  
[http://www.limsi.fr/jps/enseignement/examsma/2005/1.plateformes\\_1/jade.htm](http://www.limsi.fr/jps/enseignement/examsma/2005/1.plateformes_1/jade.htm)

[47] Systèmes Multi-Agents JADE Environnement pour la programmation multi-agent Olivier Boissier Olivier.Boissier@emse.fr 05 Janvier 2010

[48] <http://sourceforge.net/projects/wampserver/>

[49] <http://www.oracle.com/technetwork/java/overview-141217.html>