



Faculté des Sciences
Département d'informatique.

Mémoire présenté par :

- AMIMER Mohamed.
- Ayache Messaoud.

En vue d'obtenir le diplôme de Master

Domaine : Mathématique et Informatique.

Filière : Informatique.

Spécialité : Informatique.

Option : ingénierie du logiciel.

Sujet:

***Gestion coopérative d'un stock pharmaceutique
dans un Système d'information de santé une
approche Système multi-Agents/Services web***

Soutenu le : /07/2011 devant le jury composé de:

M. Président

Mme Leila Ouahrani Rapporteur. Maître Assistant (USDB).

M. Examineurs

2010/2011

Dédicaces

Avant tout je remercie le bon Dieu.

En témoignage de ma gratitude et mon respect, j'ai un grand plaisir à dédier ce modeste travail, fruit de mes études en exprimant ma profonde gratitude à tous mes proches particulièrement :

A mes précieux parents pour leur amour, affection et compréhension. Mon père pour sa patience, ses encouragements et ses sacrifices. Ma mère pour l'éducation qu'elle m'a inculquée et toutes les peines qu'elle s'est données pour ma réussite.

À mes frères Omar, Housseem, Khalil

À mes sœurs Hamida et Khaoula

À mon binome Amimer Mohamed

À mes amis Ramzi et Zinou et ma chère amie Feriel.

A tous mes amis et collègues de la promotion sortante 2010 - 2011, surtout Bellarbi youcef, sarra, Imen, Mourad, Abdelkrim amine

A tous ceux et celles dont les noms n'ont pu être cités.

Messaoude « Amine »

Dédicaces

Ô Alla A Toi la louange comme il sied
à la majesté de Ton visage et à la
Grandeur de Ton autorité.

Je dédie ce travail à mes parents
pour leur patience, leur soutien et
surtout ses sacrifices, merci
infiniment et que dieu vous garde.

À mes frères Abdelkader, Rachid,
Ramadan et Hamza.

À mes sœurs Salima, Karima, Atika et
Djamila.

À mon binôme Ayache Messaoude

À tous mes amis et collègues de la
promotion surtout Bellarbi Youcef,
Tabtroukia Mourad, Abdelkrim Amine,
Benhoura Abderrahmane ,Baouya Hakim.
Et bien sur batel Sarra et Aicha .

Mohamed

Remerciements

Nos remerciements vont à :

Allah le tout puissant, qui nous a aidés à mener à terme ce modeste travail.

Nos remerciements vont aussi à :

Madame Leila Ouahrani, pour l'honneur qu'elle nous a fait en acceptant d'être notre promotrice et pour tous les conseils et l'aide qu'elle nous a donnés, afin de pouvoir mener à terme ce travail.

Nous tenons à remercier l'ensemble des enseignants du département d'informatique pour avoir assuré notre formation, le long des cinq années d'études, et pour nous avoir transmis leur savoir sans réserve.

Nous tenons aussi à remercier les membres du jury pour l'honneur qu'ils nous font en acceptant de juger ce travail.

Enfin, nous tenons à remercier tous nos amis et collègues pour leur soutien moral tout au long de la préparation de ce mémoire.

Amimer Mohamed

Ayache Messaoud

Résumé

Nous nous intéressons à la conception et à la réalisation d'un système multi-agents à savoir un ensemble d'agents autonomes communicants pour l'équilibrage du stock pharmaceutique distribué dans un système de santé. L'équilibrage doit se faire de manière complètement transparente et distribuée. Nous utilisons les Services Web comme cadre technologique pour mettre en œuvre le système.

Mots clés : agent, ressource, service web, équilibrage, coopération.

Summary

We are interested in the design and the implementation of a multi-agent system that is a set of autonomous agents communicating to the balancing of distributed pharmaceutical stocks in a healthcare system. The balancing must be completely transparent and distributed. We use Web Services technology as a framework to implement the system.

Key words: agent, resource, web service, balancing, co-operation.

LISTE DES TABLEAUX

Liste des tableaux

Tableau		Pages
Tab.1 :	Objectifs système d'information hospitalier	20
Tab.2 :	Les différentes politiques d'approvisionnement	32
Tab.3 :	Quelques exemples de performatifs	51
Tab.4 :	Les attributs les plus courants de KQML	51
Tab.5 :	Le scénario d'équilibrage de stock (Suite à une demande)	76
Tab.6 :	Le scénario d'équilibrage de stock (Suite à une offre)	77
Tab.7 :	Tableau des termes utilisés	78
Tab.8 :	Tableau des messages échangés.	80
Tab.9 :	Tableau des relations	80
Tab.10 :	Les messages échangés	94

LISTE DES FIGURES

Liste des figures

Figures		P
Fig.1 :	Organisation structurelle de système d'information hospitalier	22
Fig.2 :	Découpage du Système d'Information hospitalier	23
Fig.3 :	Organisation structurelle de système d'information hospitalier	23
Fig.4 :	Evolution de system d'information hospitalier	24
Fig.5 :	Système d'information hospitalière centralisé	25
Fig.6 :	Système d'information hospitalier Départementaux	26
Fig.7 :	Système d'information hospitalière distribué	27
Fig.8 :	Les indicateurs de gestion de stock	31
Fig.9 :	Schéma explicatif des cas de rupture et de sur stockage	36
Fig.10 :	Modèle d'un agent réactif	40
Fig.11 :	Modèle d'un agent cognitif	40
Fig.12 :	Principe de fonctionnement de l'AOS	56
Fig.13 :	Structure du message échangé dans SOAP	59
Fig.14 :	Structure de WSDL	61
Fig.15 :	Structure d'UDDI	61
Fig.16 :	Etapas d'invocation de service web	62
Fig.17 :	Schéma de fédération des pharmacies	68
Fig.18 :	Système Multi-Agents	69
Fig.19 :	Diagramme de cas d'utilisation	73
Fig.20 :	Le scénario de Consultation de stock	74
Fig.21 :	Le scénario de gestion des demandes	74
Fig.22 :	Le scénario de gestion des offres	75
Fig.23 :	Le scénario d'équilibrage de stock (Suite à une demande)	76
Fig.24 :	Le scénario d'équilibrage de stock (Suite à une offre)	77
Fig.25 :	Diagramme de classe d'agent	78
Fig.26 :	Diagramme de Séquence « Consultation régulière de stock »	81
Fig.27 :	Diagramme de Séquence « gestion des demandes »	81
Fig.28 :	Diagramme de Séquence « gestion des offres »	82
Fig.29 :	Diagramme de Séquence «équilibrage de stock Suite à une demande»	82
Fig.30 :	Diagramme de Séquence « équilibrage de stock Suite à une offre»	83
Fig.31 :	Diagramme d'activité Consultation périodique	84
Fig.32 :	Diagramme d'activité de gestion des demandes et des offres	84
Fig.33 :	Diagramme d'activité d'équilibrage de stock (Suite à une demande)	85
Fig.34 :	Diagramme d'activité d'équilibrage de stock (Suite à une offre)	86
Fig.35 :	Schéma explique la détection d'états de sur/sous stockage	88
Fig.36 :	Le mécanisme d'équilibrage d'une ressource par l'intégration des services web	92
Fig.37 :	Le mécanisme d'équilibrage de plusieurs ressources par l'intégration des services web	93
Fig.38 :	Diagramme de séquence «Elaboration de demande »	95

LISTE DES FIGURES

Fig.39 :	Diagramme de séquence «Elaboration d'une offre »	95
Fig.40 :	Diagramme de séquence «Recherche et invocation d'une offre»	96
Fig.41 :	Diagramme de séquence «Recherche et invocation d'une offre»	97
Fig.42 :	Architecture générale de logiciel	101
Fig.43 :	menu principale de module locale	102
Fig.44 :	Fenêtre paramétrage de système	102
Fig.45 :	Message d'alerte détection d'anomalie	103
Fig.46 :	Message d'alerte recevoir un message	103
Fig.47 :	Menu principal du module global	104
Fig.48 :	Fenêtre gestion des agents fédérateurs	105
Fig.49 :	Fenêtre d'actualité de la fédération	105
Fig.50 :	Fenêtre liste des offres	106
Fig.51 :	Résultat de la recherche sur un site « 1 »	107
Fig.52 :	présentation graphique du résultat de la recherche sur un site	107
Fig.53 :	Résultat de la recherche sur une ressource « 1 »	108
Fig.54 :	Fenêtre Liste des demandes	109
Fig.55 :	Résultat de la recherche sur un site « 2 »	110
Fig.56 :	Résultat de la recherche sur une ressource « 2 »	111
Fig.57 :	Liste des offres par apport aux demandes	112
Fig.58 :	Présentation graphique des offres par apport aux demandes	113
Fig.59 :	liste d'affectation offres/demandes	114
Fig.60 :	Les interactions entre les agents	115

Table des matières

Introduction générale	15
PARTIE 1. ETAT DE L'ART.	17
Chapitre I : Système d'information hospitalier dans un environnement de santé.....	18
1. Définition d'un hôpital.....	18
2. Définition de Système d'Information Hospitalier(SIH)	19
3. Définition de Système d'Information de Santé(SIS)	20
4. Organisation du Système d'Information Hospitalier	21
5. Architectures des Systèmes D'information Hospitaliers	24
Chapitre II : Gestion de stock pharmaceutique	29
1. Définition de la pharmacie	29
2. Généralités sur le stock pharmaceutique.....	30
3. Politiques de gestion de stock	32
4. Problématique de planification des ressources réparties.....	34
Chapitre III : Les Systèmes Multi-Agents et les Services Web	38
Les Systèmes Multi-Agents	38
1. Définition d'agent	38
2. Différents catégories d'agents.....	39
2.1 Agents réactifs.....	39
2.2 Agents cognitifs	40
2.3 Agents hybrides.....	41
2.4 Agents mobiles.....	42
3. Définition de Systèmes Multi-Agents.....	42
4. Les avantages des Systèmes Multi-Agents	43
5. Les interactions entre agents	44
5.1 Coopération.....	45
5.2 Méthodes de coopération	46
5.3 Communication entre agents.....	48

5.4 Stratégies de communication	48
5.5 Les actes de langage.....	49
5.6 Les langages de communication	50
Les Services Web.....	53
1. Notion de service.....	53
2. L'Architecture Orientée Services(AOS)	54
2.1 Composants de l'architecture orientée services	55
2.2 Principe de fonctionnement.	56
2.3 Avantages de l'Architecture Orientée Services	57
3. Les Services Web.....	57
3.1 Pour quoi les Services Web ?	58
3.2 Infrastructures des Services Web	58
A. SOAP pour la communication.....	59
B. WSDL pour la Description	60
C. UDDI pour la Découverte	61
3.3 Invocation des Services Web	62
3.4 Bus de services d'entreprise (ESB)	62
4. Vers un couplage SMA/SW	64

PARTIE.2. CONCEPTION ET REALISATION	66
I. Conception du Système	67
I.1. Analyse de l'environnement	67
I.2. Organisation du système Multi-Agent	69
I.3. Typologie des agents	70
I.3.1. Agent gestionnaire de stock(A.G.S)	70
I.3.2. Agent de Ressource (A.R)	71
I.3.3. Agent Fédérateur (A.F)	72
II. Modélisation du Système	73
II.1. Diagramme de cas d'utilisation	73
II.2. Les scénarios	74
II.2.1. Consultation périodique de stock	74
II.2.2. Gestion des demandes	74
II.2.3. Gestion des offres	75
II.2.4. équilibrage des stocks pharmaceutique (Suite une demande)	76
II.2.5. équilibrage des stocks pharmaceutique (Suite à une offre)	77
II.3. Diagramme de classe	78
II.4. Diagramme de séquence	81
II.4.1. Consultation périodique de stock	81
II.4.2. Gestion des demandes	81
II.4.3. Gestion des offres	82
II.4.4. Équilibrage des stocks pharmaceutique (Suite une demande)	82
II.4.5. Équilibrage des stocks pharmaceutique (Suite à une offre)	83
II.5. Diagramme d'activité	84
II.5.1. Consultation périodique de stock	84
II.5.2. Gestion des demandes et des offres	84
II.5.3. Équilibrage des stocks pharmaceutique (Suite une demande)	85
II.5.4. Équilibrage des stocks pharmaceutique (Suite à une offre)	86
II.6. Modèle mathématique	87
A. Pour la détection des cas de rupture ou du sur stockage	87
B. Pour la sélection et le choix de meilleure offre	89
C. Pour la sélection et le choix de meilleure demande	90

II.7 Les différents Scripts des agents	90
A. Script de l'A.G.S	90
B. Script de l'A.R	90
C. Script de l'A.F	91
II.8. Mise en œuvre de la solution	92
II.8.1. Solution web services	92
II.8.2. Diagrammes de Séquences	94
A. Elaboration des demandes	95
B. Publication d'offre	95
C. Recherche et invocation d'une offre.....	96
D. Recherche et sélection d'une demande	97
Chapitre II : La Réalisation du Système	98
II.1. Les outils de développement	98
II.2. L'architecture général de logiciel	100
II.3. Architecture fonctionnelle et comportement de logiciel	102
Conclusion	116
Bilan du travail.....	117
Bibliographie.	
Web-graphie.	
Annexe.	

PARTIE 1 :

ETAT DE L'ART.

INTRODUCTION GENERALE

Introduction à la problématique

Les applications ont besoin d'accès coordonnés à des ressources distribuées afin de synthétiser leurs informations provenant de ces ressources. Le partage des ressources présente un intérêt majeur puisqu'il permet de réduire le coût d'acquisition.

La gestion de stocks pharmaceutiques multiples est un bon exemple d'accès et de partage de ressources dans un environnement distribué. C'est dans le contexte d'un Système d'Information de Santé distribué que nous voulons proposer une solution distribuée d'équilibrage de stocks en utilisant une approche mixte Agents/Services Web. En effet les Services Web ont pour vocation de favoriser une architecture orientée services, intégrant des systèmes hétérogènes, distribués et pouvant coopérer sans recours à une intégration spécifique et coûteuse.

Cadre du mémoire

Cette étude rentre dans le cadre de la santé. Elle s'inscrit dans la démarche d'amélioration de la qualité des soins et de la maîtrise des coûts dans le secteur sanitaire publique.

Objectifs et besoins

L'objectif du travail est de réaliser un système multi-agent à savoir un ensemble d'agents autonomes communicants supportant les objectifs suivants :

- La gestion du stock multiple d'un SI de santé en domaine distribué : En effet, un stock dépassant un seuil non toléré affecte négativement la qualité des soins : Un sous-stockage provoque une rupture et un sur-stockage provoque des coûts importants (péremption, ...). Il s'agit alors de réaliser et de maintenir une situation d'équilibre optimale.
- Contribution à base d'agents à la modélisation optimale et distribuée du stock pharmaceutique afin d'assurer un gain en terme de coût et de disponibilité des médicaments.
- Une couche à base de Services Web doit être développée afin d'assurer la mise en œuvre de la solution.

Structure du document

Pour mener à bien notre étude, nous avons structuré notre mémoire en deux parties :

La première partie (état de l'art) : elle est consacrée pour les différents concepts liés à cette étude, cette partie contient trois chapitres :

Chapitre 1 : Ce chapitre est consacré aux systèmes d'information dans les établissements de santé à savoir : le Système d'Information Hospitalier(SIH), Le Système d'Information de Santé (SIS).

Chapitre 2 : ce chapitre introduit les notions liées à la gestion de stock pharmaceutique ainsi que le problème de planification des ressources.

Chapitre 3 : le chapitre trois a pour but d'étudier les agents et les systèmes multi-agents ainsi que pour l'architecture orientée service et son instance service web qui constitue notre cadre technologique de référence et nous concluons ce chapitre par les approches de couplage entre les systèmes multi-agents et les services web.

La deuxième partie (Conception et Réalisation) : dans cette partie nous présentons notre solution proposée pour la gestion coopérative de stock pharmaceutique ainsi que l'implémentation et l'expérimentation de cette solution. Cette partie contient trois chapitres :

Chapitre 1 : Dans ce chapitre nous avons proposé une solution multi agents pour l'équilibrage de stock pharmaceutique ainsi que la modélisation de cette solution en utilisant le langage de modélisation pour les agents AUML (Agent Unified Modeling Language), et en fin le modèle mathématique utilisé pour cette solution.

Chapitre 2 : dans ce chapitre nous présentons les différents outils logiciels que nous avons choisis pour l'implémentation, ainsi que le comportement et l'architecture de notre prototype.

Ce mémoire est clôturé par une conclusion générale ou nous mettons en évidence les points positifs, les limites et les perspectives de cette étude.

Chapitre I :

Système d'information Hospitalier dans un environnement de santé.

Introduction

L'informatique ne cesse d'envahir les différents domaines des activités humaines. Cela s'explique par son apport incontestable pour ceux qui l'utilisent. En effet, cet outil permet entre autres l'automatisation des traitements, l'échange d'information soit en temps réel ou soit en différé, la conservation des données, l'exécution rapide des tâches, etc. Ayant constaté qu'ils peuvent bénéficier de ces avantages, les centres hospitaliers ont opté de ne pas se mettre en marge de ce processus général d'informatisation. C'est ainsi que les systèmes d'information hospitaliers (SIH) ont commencé à voir le jour.

L'hôpital est une organisation un peu complexe associant plusieurs unités (soins, médico-techniques...) avec des fonctionnalités distinctes et une certaine autonomie cependant, il ne peut fonctionner correctement que s'il existe une communication et une coopération entre ses unités afin de traiter au mieux les patients.

1. Définition d'un hôpital :

Un établissement de santé dont la mission essentielle est de soigner et de guérir les malades.

Pour accomplir ses mission l'hôpital dispose d'un certain nombre de ressources humaines, matérielles et logicielles, l'hôpital est en effet une fédération de sous-systèmes fonctionnellement distincts mais non disjointes à savoir :

- **Sous système médical :** Concerne l'activité mise en place par l'équipe soignante pour répondre aux besoins des patients.
- **sous système logistique :** Comprend l'ensemble des flux résultant des activités médicales : Prescriptions, résultats, transferts, archivage...
- **sous système d'étude et recherche :** Travaille sur des regroupements de dossiers afin d'apprécier et d'améliorer la qualité des soins dispensés. Outre, il alimente les sous-systèmes planification et administration.

- **sous système administration** : C'est le sous-système de l'hôpital qui s'intéresse à la facturation, à la gestion du personnel, à la gestion des stocks et à la comptabilité d'une manière générale.
- **sous système planification** : Il s'appuie sur les activités ou les études de morbidité hospitalière pour engager des décisions d'investissements structurels, matériels et humains. C'est le sous-système qui est en interaction avec l'extérieur (autorités et autres établissements de santé). [Otm et Smi, 10].

2. Définition de Système d'Information Hospitalier (SIH):

Le système d'information hospitalier (**SIH**) est un ensemble d'éléments en interaction ayant pour objectif de produire, traiter et fournir des informations nécessaires à l'activité hospitalière. [Otm et Smi, 10].

C'est un système d'information appliqué aux métiers de la santé et plus particulièrement aux établissements de santé, il représente l'ensemble des ressources (humaines, matérielles et logicielles) organisées pour collecter, stocker, traiter et communiquer les informations au sein de l'hôpital. Le **SIH** est l'une des composantes du Système d'Information de Santé (**SIS**).

Les principaux établissements pouvant mettre en place un SIH sont :

- Les hôpitaux.
- Les cliniques.
- Les centres radiologies.
- Les centres d'analyses.
- Les centres de soins.
- Les cabinets médicaux.

Les acteurs participant au système d'information hospitalier sont :

- Les patients.
- Les professionnelles de la santé.
- Le personnel soignant.
- Le personnel administratif.

Les Objectifs de système information hospitalier (SIH) sont représentés dans le tableau suivant : [Deg, 04]

Principaux	Constitutifs
Améliorer la qualité des soins	<ul style="list-style-type: none"> - la Faciliter le partage et la communication - Améliorer la continuité des soins - Améliorer la sécurité et la traçabilité - Aide à la prise de décisions
Maitrise des couts	<ul style="list-style-type: none"> - Optimisation des ressources - Réduction de la durée des séjours - Diminution des frais de personnel

Tab.1 : Objectifs système d'information hospitalier

3. Définition du Système d'Information de Santé (SIS):

Par système d'information da santé (SIS) on entend:

Le recueil de l'information au niveau de différent institutions ou organisations (hôpitaux, médecins de ville, services de sécurité sociale...), son traitement, sa mémorisation, et enfin sa transmission vers des organismes habilités à la prise de décision à savoir :

- Les gestionnaires des structures de santé (GSS).
- la direction de la santé et de la population(DSP).
- Le ministère de la santé(MS).

Les acteurs participant au système d'information de santé sont en plus aux différents acteurs de SIH (car le SIH est l'une des composantes de SIS) :

▪ **La population.**

▪ **Le système de soins :**

- système des soins primaires : Le système des soins primaires est celui qui répond à une demande de soins exprimée spontanément par la population : il s'agit des premières visites ou consultations, urgences à l'hôpital etc....
- système des soins secondaires : précise et oriente la demande de soins spontanées par des examens de radiologie, de laboratoire ou carrément par une hospitalisation.

- **Les organismes de sécurité sociale (Branche- assurance-maladie)** : Ces organismes ont pour vocation le financement des établissements publics de santé.
- **Les pouvoirs publics** : ont la charge de l'élaboration des politiques de santé et œuvrent pour la préservation de la santé publique. [Sal, 07].

La diversité des acteurs de SIS (Etat, assurance maladie, structures de soins, professionnels de santé, patients, industriels) s'est traduite par une multiplicité de systèmes d'information ignorant les systèmes voisins, développés isolément, répondant à des objectifs ou des besoins spécifiques. Cette diversité pose des difficultés à la mise en œuvre de SIS.

Les Objectifs de système information de santé (SIS) sont :

- Permettre la prise de décision.
- Produire des informations pertinentes et de qualité pour l'aide à la décision.

Ces deux objectifs vont mener à : l'amélioration de la qualité des soins dispensés à la population d'une part et la maîtrise des dépenses de santé d'autre part.

4. Organisation de système information hospitalière :

Niveau Structurel : [Deg, 04]

- Services administratifs :
 - ✓ Direction, Services économiques, Direction du Personnel,...
- Pharmacie.
- Unités de soins :
 - ✓ Consultation externes, services médicaux, services chirurgicaux, soins intensifs, réanimation, urgences, rééducation....
- Unités Médico-techniques :
 - ✓ Radiologie, Endoscopie.
 - ✓ Laboratoire.
 - ✓ Télémédecine.
- Services logistiques :
 - ✓ Transport, Blanchisserie, Restauration, Entretien

▪ Structure de l'information :

- ✓ Archives.
- ✓ Informatique.
- ✓ Statistiques.
- ✓ Communication.

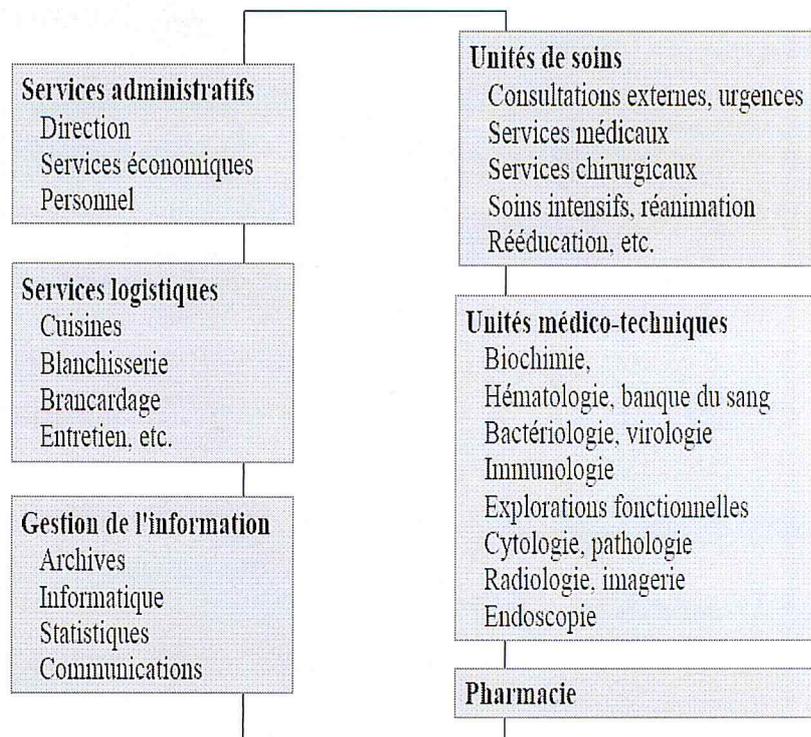


Fig.1 : Organisation structurelle de système d'information hospitalier

Niveau Fonctionnel : [Mon, 04]

Le système d'information hospitalier est divisé en 3 sous-systèmes :

- **Le sous-système médico-administratif :** Ce sous-système assure trois fonctions principales, qui sont à savoir :

La gestion des patients :

- La gestion des admissions (hospitalisation).
- La facturation.
- L'imagerie médicale.
- Le suivi des examens du laboratoire de biochimie.

Gestion financière et comptable :

- Le système de gestion comptable.
- La gestion financière.

La gestion du patrimoine :

- La Gestion des Approvisionnements et des Stocks Magasin.
- La gestion des stocks de médicaments.
- La Gestion du Stock des Dispositifs Médicaux et des Ligatures
- Le Suivi de la Maintenance du Parc Biomédical

- **Le sous-système médical :** Le sous système médical est considéré comme le cœur du SIH autour duquel s'organisent les deux autres sous systèmes. Il est composé de l'ensemble des unités de soins dans lesquelles se déroule la plus grande partie de l'activité médicale de l'hôpital (prescriptions, comptes rendus, soins infirmiers, visites médicales,...).
- **Le sous-système médico-technique :** Le sous-système médico-technique comprend au sens large tous les plateaux d'examens (laboratoires d'analyse, centres d'imagerie médicale, centres d'explorations fonctionnelles...).

Un schéma illustratif des composantes de système d'information hospitalier

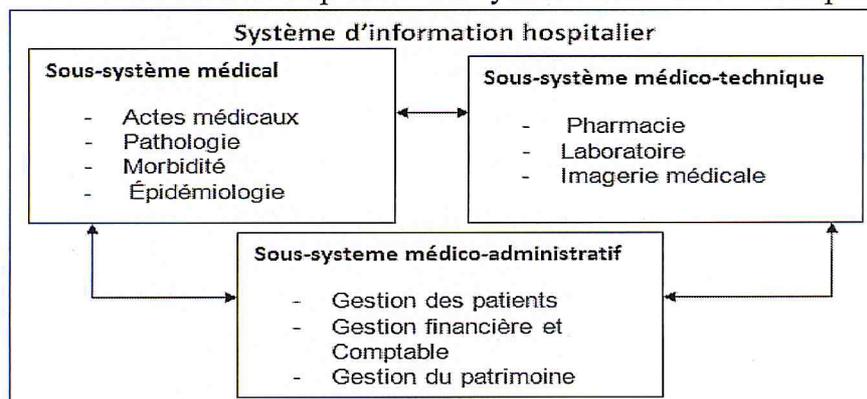


Fig.2 : Découpage du Système d'Information hospitalier

Niveau informationnel :

Système d'information hospitalier
Données administratives
Données médicales
Données de soins

Fig.3 : Organisation structurelle de système d'information hospitalier

Comme cette figure montre nous pouvons classer les données du système d'information hospitalier en deux catégories des données selon leurs natures, les données administratives, et les données médicales et de soins.

- Les données administratives : Ce sont les données relatives aux personnels de l'établissement, factures, comptabilités, gestion du stock etc.

- Les données médicales et de soins : Ce sont les données relatives aux activités des soins des patient, les différents données viennent des fonctions et activités présentes dans l'hôpital (hospitalisation, consultation).

5. Architecture des systèmes d'information hospitaliers :

L'évolution de SIH : [Deg, 04]

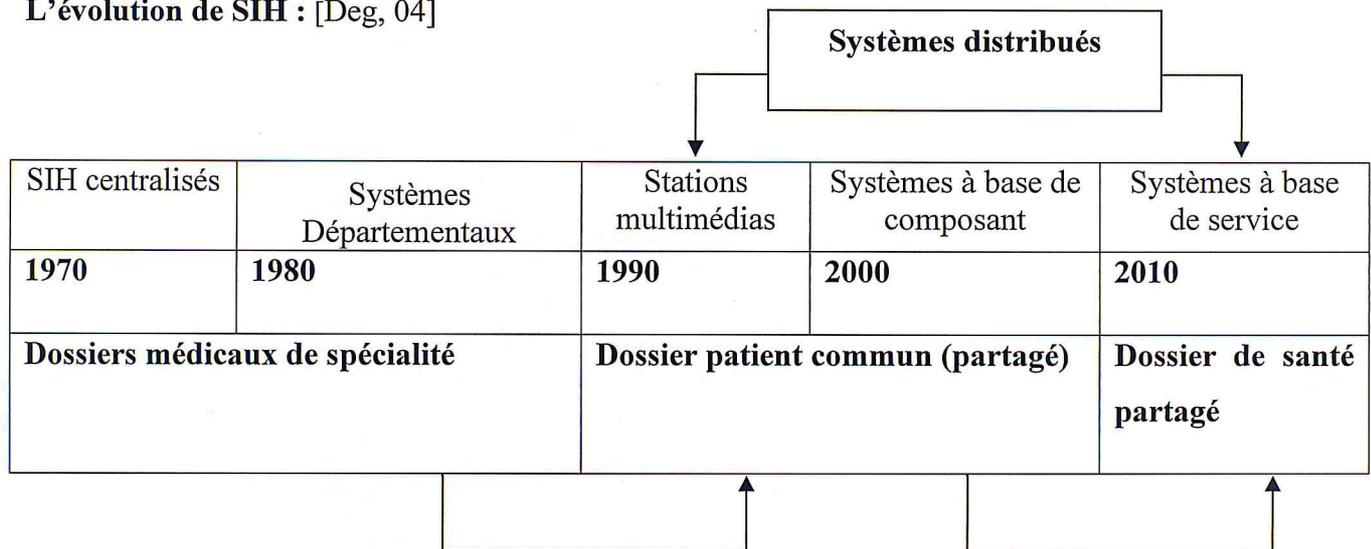


Fig.4 : Evolution de system d'information hospitalier

Les SIH centralisés : [Otm et Smi, 10]

Est un système centralisé en étoile mis en place dans les années 70. Cette architecture comprend un ordinateur principal et des périphériques reliés en étoile.

Le principe de cette architecture est que l'information est saisie une fois (principe de non redondance), stockée en un point unique de la base de données mais accessible de n'importe quel point du réseau (principe de partage). C'est cette approche qui est le plus souvent commercialisé dans les systèmes dits « clé en main ». Cette approche a trouvé sa réussite dans les hôpitaux de petite taille (300-400 lits) et à vocation de recherche peu développé.

Parmi les systèmes conçus par cette approche on trouve :

- PCS (patient care system) est le plus ancien représentant de ce type d'architecture.
- Help est organisé autour d'un dictionnaire intelligent et offre des alarmes en cas de non respect des protocoles.
- TDS est l'une des réalisations les plus démonstratives.

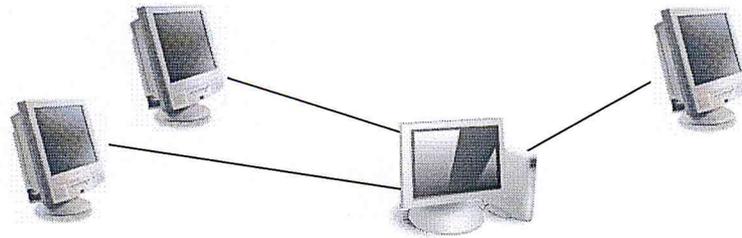


Fig.5 : Système d'information hospitalière centralisé

Avantage :

- Système intégré centré sur le patient.
- Mise en service et maintenance facile des modules applicatifs.
- Contrôle facile du système.
- Système clé en main.

Inconvénients :

- Coûts de maintenance élevée.
- Évolution non progressive.
- Prise en compte partielle des besoins des utilisateurs.
- Standardisation élevée.

Les SIH Départementaux : [Otm et Smi, 10]

Consiste en l'achat des différentes structures de l'hôpital d'applications spécialisées. Les unités médico-techniques ont été les premières à être informatisées après les services administratifs. L'informatisation des unités de soins est beaucoup plus complexe car la demande médicale varie d'un service à un autre et même à l'intérieur d'un service.

Cette approche est utilisée dans plusieurs institutions hospitalières à composante universitaire très forte : Assistance Publique de Paris, John Hopkins, Massachusetts General Hospital.

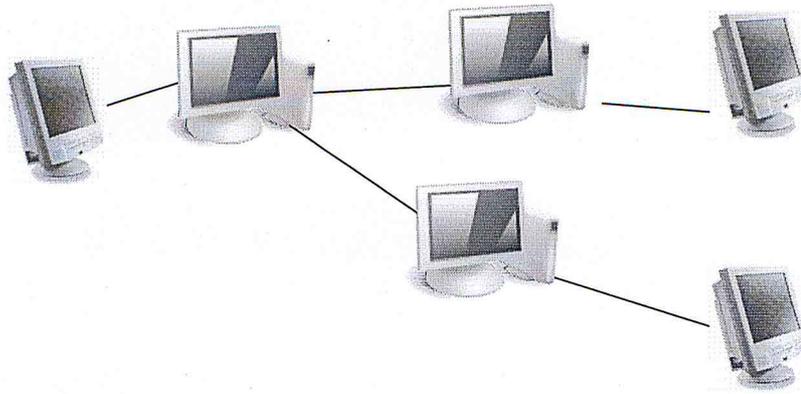


Fig.6 : Système d'information hospitalier Départementaux

Avantage:

- Meilleurs adaptation des produits à la demande des utilisateurs.
- Dissociation du matériel et du logiciel.
- Investissement progressif.
- Applications multi-hospitalières.

Inconvénients :

- Redondance de l'information
- Difficulté de maintenir l'intégrité et la cohérence de l'information
- Coût élevé de l'intégration en l'absence de standards de communication

Les SIH distribués : [Otm et Smi, 10]

L'idée de cette approche est de combiner les avantages des deux premières architectures. Les applications sont réparties sur plusieurs processeurs de traitement suivant une architecture client/ serveur (serveur d'identité, serveur de messagerie, serveur de résultats, serveur de données cliniques).

La complexité de cette approche c'est qu'elle nécessite plusieurs niveaux d'intégration : architecture matérielle, logiciels adaptés, réseaux. Elle nécessite aussi une communication entre les applications en utilisant des normes de bas niveau mais intégrant la sémantique des messages.

Internet peut être considéré comme un système distribué qui a fait son apparition dans le domaine hospitalier sous forme d'un intranet.

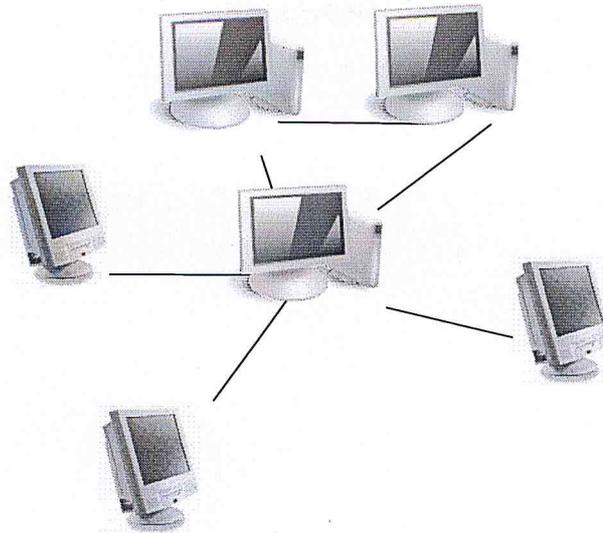


Fig.7 : Système d'information hospitalière distribué

Inconvénients :

- La défaillance d'un serveur peut altérer le bon fonctionnement du système global

Avantage :

- Il englobe les avantages des autres systèmes.

En réalité, il n'existe pas de définition absolue du bon système d'information. En général, un bon système d'information doit comporter les critères mentionnés ci-après [Des, 08] :

- Cohérence.
- Homogénéité.
- Pertinence.
- Adaptabilité.
- Appui sur des standards reconnus.
- Robustesse et Permanence.

Conclusion :

A la fin de cette analyse on peut dire que le système d'information joue un rôle très important dans le fonctionnement de l'hôpital et ses partenaires. En général, le système d'information peut être considéré comme :

• **Un outil de qualité des soins :**

Le système d'information doit servir à améliorer le fonctionnement de l'hôpital, ce fonctionnement est étroitement lié au soin des malades. En conséquence, le système d'information peut être considéré comme un outil important de qualité des soins.

• **Un outil de cohérence organisationnel:**

Le bon fonctionnement d'un hôpital dépend de la coordination et de la communication entre tous les services cliniques et administratifs.

Le système d'information hospitalier conçu doit permettre de faciliter la communication entre les acteurs concernés dans l'hôpital.

• **Un outil de stratégie et de décision :**

Le système d'information est considéré non plus comme un outil d'enregistrement de données, mais comme un outil permettant aux hôpitaux d'analyser les données recueillies, de prendre une décision et d'élaborer une stratégie en gestion.

Chapitre II :

Gestion du stock pharmaceutique

Introduction :

Une des structures les plus importantes dans un SIH est la pharmacie. Notre vision de la pharmacie dans cette étude est une vision orientée vers la gestion et l'équilibrage des ses ressources en stock à savoir les médicaments.

Avant d'entamer cette étude nous allons citer les notions liées cette vision (la pharmacie, des généralités sur le stock et en fin la problématique de la planification répartie des ressources)

1. Définition de la pharmacie :

- Magasin, local où l'on prépare et où l'on met des médicaments à la disposition du public ou des professionnels de la santé. Pharmacie d'un hôpital.
- Petite armoire ou petite trousse portative où l'on range les médicaments [Lar, 08].

La pharmacie a comme rôles :

- Délivrance des médicaments et des produits liés à la santé.
- Améliorer la disponibilité et l'accessibilité des populations aux médicaments.
- Garantir la qualité des médicaments et des réactifs de laboratoire.

Nous remarquons que le médicament constitue l'élément essentiel du stock pharmaceutique est c'est une substance ou préparation administrée en vue de traiter ou de prévenir une maladie, ou de restaurer, corriger, modifier des fonctions organiques. [Lar, 08].

Il constitue l'un des maillons les plus importants dans la chaîne de prise en charge des principales affections rencontrées dans nos communautés.

La disponibilité des médicaments est une condition de bonne fréquentation des services de santé par les populations et c'est important d'assurer une disponibilité pour une bonne prise en charge des malades.

Donc la clé d'une bonne prise en charge est étroitement dépendante de la disponibilité des médicaments au niveau de toute structure sanitaire. Ce qui aura, du reste, un impact certain sur la qualité de services.

2. Généralités sur le stock :

Il nous semble essentiel que pour pouvoir poursuivre notre étude et pour pouvoir bien clôturer notre problématique, il faut présenter quelques généralités sur le stock et sa gestion.

Définition de stock : [Gue et Gue, 10]

Un stock, c'est un ensemble d'articles ou de marchandise (Médicaments) maintenue en magasin ou dans des espaces de stockage (Pharmacie) en attente destinés à satisfaire les demandes des clients (Patients) d'une manière immédiate.

Quand est-ce qu'on a besoin de stocker ?

- Non coïncidence dans le temps ou l'espace de la production et de la consommation.
- Incertitude sur le niveau de demande ou sur le prix.
- Risque de problèmes en chaîne : il s'agit ici d'éviter qu'une panne à un poste ne se répercute sur toute la chaîne d'approvisionnement :
 - Un retard d'exécution au poste précédent
 - Une grève etc.

Ce stock doit être capable de répondre aux besoins des patients d'une manière permanente avec un coût minimal que ce soit pour le patient ou pour la pharmacie. Mais ces objectifs ne sont pas souvent atteints à cause des risques et menaces qui entourent le stock comme (la rupture du stock, le sur stockage...).

* Ruptures de stock : [Mos, 04]

Un stock dépassant un seuil non toléré affecte négativement sur la qualité de soins, un sous stockage provoque une rupture de stock.

Coût des ruptures

- . Temps perdu
- . Risque pour le patient

* Le sur stockage :

Lorsque le stock est en état de sur stockage il y a un risque de :

- péremption des médicaments.
- faire des dépenses supplémentaires surtout pour les médicaments périmés (stockage, destruction).

Pour palier ces menaces la nécessité d'une gestion rigoureuse de ressource (le stock constitue une ressource) devient indispensable.

Gestion de stock :

Activités et techniques permettent :

- De maintenir le stock d'articles à un niveau souhaité, que ce soient des matières premières, des en-cours ou des produits finis.
- D'assurer une utilisation des ressources en vue de garantir la disponibilité des articles

L'objectif de la gestion des stocks est d'éviter toute rupture dans l'approvisionnement, tout en diminuant les coûts de stockage. Il s'agit de trouver une solution optimale.

La gestion de stock permet aussi de définir différents niveaux possibles pour chaque produit qui sont appelés aussi les indicateurs de gestion des stocks :

- **Le stock physique:** les quantités de marchandises réellement dans l'entrepôt
- **Le stock réel:** le stock physique + les commandes fournisseurs en cours de livraison – les commandes clients en cours (à livrer).
- **Le stock de sécurité (SS):** C'est le niveau de stock qui permet de répondre aux aléas les plus fréquents liés à la consommation et à la livraison.
- **Le stock maximum :** c'est le stock à ne pas dépasser en max, utilisé pour les commandes fournisseurs, ceci permet d'effectuer automatiquement la commande pour atteindre la quantité maximum d'un produit.
- **Le stock minimum :** Encore appelé stock tampon ou stock d'alarme. Il correspond à la consommation de l'article durant le délai d'approvisionnement (laps de temps entre la commande et la livraison). Exemple : si le délai d'approvisionnement est de 5 jours et que les consommations quotidiennes sont de 100 unités, le stock minimum est de $5 \times 100 = 500$ unités.

Stock minimum = stock d'alarme – stock de sécurité

- **Le stock d'alarme:** Encore appelé stock critique. C'est le niveau de stock pour lequel on déclenche la commande au risque sinon de connaître une rupture.

Stock d'alarme = stock minimum + stock de sécurité

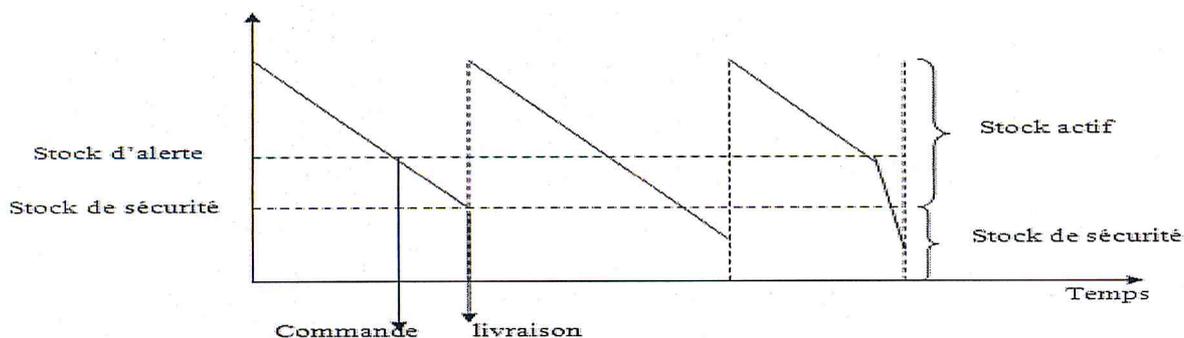


Fig.8 : Les indicateurs de gestion de stock

3. La politique de gestion de stock :

Définir un stock de sécurité et un stock d'alerte

Le stock d'alerte doit permettre à l'entreprise de couvrir les sorties (ou consommations) qui auront lieu entre la date de passation de la commande et la date théorique de livraison.

Le stock d'alerte intègre le stock de sécurité qui doit permettre de faire face à un éventuel retard de livraison ou un aléa de consommation. On peut donc dire qu'une livraison effectuée à la date normale intervient au moment où le stock de sécurité est atteint.

Il est nécessaire de tenir compte du temps qui s'écoule entre la commande et la livraison ceci afin d'éviter des ruptures de stock. Éviter les ruptures Le niveau de ce stock résulte d'un calcul probabiliste en période d'incertitude (coûts, délais).

Choisir une politique d'approvisionnement : [J], 04]

La recherche des plannings de gestion des approvisionnements suppose de prévoir le rythme de consommation.

La problématique d'approvisionnement consiste à définir Quand commander? (Période d'approvisionnement) et Combien commander? (Quantité de commande).

Il y a plusieurs possibilités résumées comme suit :

	Période fixe	Période variable
Quantité fixe	Calendaire	Approvisionnement sur seuil
Quantité variable	Approvisionnement périodique	Réapprovisionnement à la commande

Tab.2 : Les différentes politiques d'approvisionnement

La politique doit être adaptée aux caractéristiques des articles gérés, il est donc fréquent de combiner les différentes politiques pour gérer un approvisionnement global.

➤ **Calendaire** : Les produits sont livrés à dates fixes et en quantités fixes.

Avantage :

- ✓ Simplifie la gestion des stocks

Inconvénients :

- ✓ Si la quantité de réapprovisionnement est mal calculée ou si la consommation n'est pas régulière il y a risque :

- D'inflation du stock
- De rupture du stock.

➤ **Approvisionnement sur seuil (méthode du point de commande) :**

Le seuil, ou point de commande correspond au niveau de stock qui, une fois atteint, déclenchera la passation de commande.

Avantage:

- ✓ Permet d'éviter les ruptures de stocks.
- ✓ Adapté à une consommation irrégulière.

Inconvénients :

- ✓ Impose un suivi permanent des stocks pouvant entraîner des coûts administratifs importants.
- ✓ Peut encourager à faire des stocks de sécurité
- ✓ Difficulté de calculer le DL (Délai de Livraison), qui dépend à la fois :
 - Du délai fournisseur : temps annoncé par le fournisseur pour la livraison de la marchandise après réception de la commande.
 - Du délai administratif : Temps qui s'écoule entre la connaissance du niveau de stock et l'arrivée de la commande chez le fournisseur. Il comprend le temps administratif interne (à l'entreprise) et le temps administratif externe (envoi au fournisseur)
 - Du temps de connaissance du niveau des stocks : Temps qui s'écoule entre le moment où le stock arrive à un niveau physique donné et le moment où les services fonctionnels, chargés des commandes) sont au courant.

➤ **Approvisionnement périodique (ou par complètement de stock) :**

L'approvisionnement périodique consiste à fixer un calendrier d'approvisionnement.

Avantage :

- ✓ Gestion des stocks simple;
- ✓ Immobilisation financière faible ou maîtrisée.

Inconvénients :

- ✓ Possibilité de rupture de stocks.

➤ **Réapprovisionnement à la commande :** Les produits sont livrés à dates variables et en quantités variables, selon les besoins.

Avantage :

- ✓ Permet de profiter des variations de prix

Inconvénients :

- ✓ Impose un suivi permanent des coûts du marché pour effectuer les achats les plus intéressants
- ✓ Ne peut être utilisé que pour un nombre réduit d'articles, sinon risque de d'une gestion extrêmement complexe
- ✓ Peut favoriser la spéculation.

Le choix du type de gestion est fortement lié à :

- La valeur d'achat de la ressource (médicament).
- La quantité de ressources utilisées par unité de temps.
- La stabilité ou non de la demande.
- Le bon fonctionnement des circuits administratifs.
- La relative stabilité des délais d'approvisionnement.

Une bonne gestion de stock pharmaceutique doit permettre :

- Connaître à chaque instant l'état de mon stock (en nombre d'unités de chaque médicament)
- Éviter les ruptures de stock.
- Faciliter le passage des commandes.
- Connaître la consommation de la pharmacie pour chaque médicament.
- Suivi des patients fréquentant la pharmacie.

4. Problématique de planification répartie des ressources pharmaceutiques :

Assurer la disponibilité des médicaments d'une manière permanente sans avoir ni sur stockage ni sous stockage a vue beaucoup de problèmes surtout dans un domaine distribué des ressources comme le cas des pharmacies. Ce problème devient beaucoup plus difficile à cause de manque de coopération et de communication qui caractérisent les pharmacies. Cette réalité nous mène d'étudier les problèmes liés aux distributions et planification des ressources afin de surmonter ces difficultés et pour mieux exploiter les ressources des différents sites de manière optimale.

Définition de la ressource :

Dans la littérature le mot ressource prend plusieurs définitions parmi elles :

- Ce qu'offre le milieu naturel et qui est susceptible d'être exploitée.
- Ce qu'on emploie pour se tirer d'embarras.

- Dans notre cas les ressources c'est l'ensemble des médicaments que possède la pharmacie.

Définition de la planification :

L'activité de planification consiste à concevoir pour l'entreprise un "futur souhaitable" et les moyens nécessaires pour parvenir à la réalisation de ce futur.

La planification vise une programmation prévisionnelle de la production, des approvisionnements et de la distribution, à partir de demandes commerciales prévues ou réelles. [Car, 03].

Pourquoi planifier :

Car les ressources coûtent cher la planification vise à :

- ✓ Établir un système de suivi des ressources.
- ✓ Affecter, exploiter les ressources disponibles de manière optimale.
- ✓ Rendre les informations relatives aux ressources disponibles aux gestionnaires.
- ✓ Éviter les conflits d'usage des ressources.
- ✓ Maitriser les contraintes de délais.

Comment planifier les ressources :

La manière de planifier les ressources diffère selon leurs types mais de manière générale le processus de planification est composé de quatre étapes à savoir :

Avant-projet : Consiste à répondre aux questions pourquoi planifier? Et planifier quoi?

Conception détaillée : Dans cette partie on introduit notre politique de planification.

Planification : Répondre à la question quand planifier et avec quels moyens ?

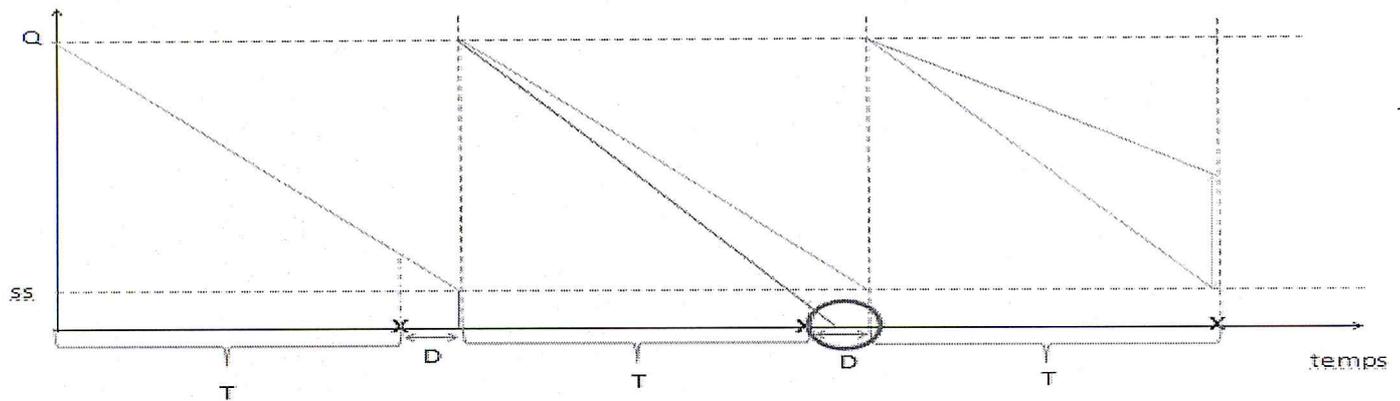
Déroulement : Réalisation du plan de planification.

Clôture : C'est la fin du processus de planification .Cette dernière étape vise à établir un bilan et archives.

Conclusion :

Après l'étude de différentes méthodes de gestion de stock, et après une analyse au sien des hôpitaux nous avons vu que la plupart des pharmacies utilisent la méthode de Recomplètement (date fixe quantité a commandé variable), car l'approvisionnement se fait de manière bien déterminée (trimestre, semestre), et que la consommation des médicaments n'est pas régulière donc les quantités à commander à chaque approvisionnement sont différentes.

Cette méthode peut engendrer des problèmes de sous ou sur stockage, si la consommation de médicaments se change dans une période donnée. Le schéma ci dessous explique les cas de rupture et de sur stockage rencontrés en utilisant cette méthode.



T : l'approvisionnement se fait chaque période T

D : délai de livraison

○ : Rupture de stock

↕ : Sur Stockage

— : Cas normale de la consommation

Ss : Stock de sécurité (en cas de retard de livraison)

Fig.9 : Schéma explicatif des cas de rupture et de sur stockage

La gestion de stock est une tâche très importante pour assurer une bonne exploitation des ressources, dans un domaine distribué cette gestion est difficile et les solutions classiques « gestion locale sans coopération, collaboration » ne prennent pas en considération le problème de manière rigoureuse.

Grâce aux capacités intellectuelles et l'autonomie qui caractérisent les Agents/Système multi-agents que nous allons présenter en détaille dans le chapitre suivant, cette gestion devient plus simple, plus flexible avec très grande transparence.

On entend par le mot transparence :

Transparence de consultation: une opération de consultation doit donner le même résultat sur n'importe quel site.

Transparence de mise à jour: émise sur n'importe quel site, une mise à jour impliquant plusieurs sites, donnera le même résultat.

Transparence de schémas : les schémas de tous les sites peuvent être rendus visibles sur n'importe quel site.

Transparence de performance: les requêtes sont comparables en performance sans tenir compte du site d'émission.

Transparence de transaction: sérialisabilité de toute transaction de mise à jour en fonction des transactions réparties concurrentes.

Transparence de copie (indépendance à la duplication des copies de données peuvent être effectuées sur des sites pour des raisons de performance de traitement sans que cela ne soit visible à l'utilisateur. [She,93].

Chapitre III :

Les Systèmes Multi-Agents et Les Services web

Les Systèmes Multi-Agents :

L'informatique devient de plus en plus diffuse et distribuée. La décentralisation et donc, la coopération entre modules logiciels est un besoin. De plus, avec la croissance de la taille, de la complexité de nouvelles applications informatiques, la vision centralisée et assez statique atteint ses limites. On est ainsi naturellement conduit à chercher une façon de donner plus d'autonomie et d'initiative aux différents modules logiciels. Le concept de système multi-agents propose un cadre de réponse à ces enjeux.

1. Définition d'agent :

De nos jours, le mot « agent » est utilisé dans plusieurs domaines et, de ce fait, plusieurs sens lui sont attachés. D'ailleurs, même à l'intérieur du domaine de l'informatique, plusieurs chercheurs ont défini le concept d'agent de manières différentes.

Dans la littérature spécialisée, on trouve une multitude de définitions des agents. Elles se ressemblent toutes sur les concepts généraux, mais diffèrent selon le type d'application pour lequel est conçu l'agent.

Définition proposée par Ferber [Fer, 95]:

Un agent est une entité autonome, réelle ou abstraite, qui est capable d'agir sur elle-même et sur son environnement, qui, dans un univers multi-agents, peut communiquer avec d'autres agents, et dont le comportement et la conséquence de ses observations, de ses connaissances et des interactions avec les autres agents.

Définition proposée par Jennings, Sycara et Wooldridge [Ngu, 03] :

Un agent est un système informatique, situé dans un environnement, et qui agit d'une façon autonome et flexible pour atteindre les objectifs pour lesquels il a été conçu.

- ❖ **Situé** : l'agent est capable d'agir sur son environnement à partir des entrées qu'il reçoit de ce même environnement (systèmes de contrôle de processus, systèmes embarqués, etc.).
- ❖ **Autonome** : l'agent est capable d'agir sans l'intervention d'un tiers (humain ou agent) et contrôle ses propres actions ainsi que son état interne.
- ❖ **Flexible** : l'agent dans ce cas est :
 - **Capable de répondre à temps** : l'agent doit être capable de percevoir son environnement et d'élaborer une réponse dans les temps requis.
 - **Proactif** : l'agent doit exhiber un comportement proactif et opportuniste, tout en étant capable de prendre l'initiative au bon moment.
 - **Social** : l'agent doit être capable d'interagir avec les autres agents (logiciels et humains) quand la situation l'exige afin d'accomplir ses tâches ou aider ces agents à accomplir les leurs.

2. Différentes catégories et modèles d'agents : [Mar, 03]

Après avoir défini les agents, nous présentons dans cette partie les différents modèles d'agents, afin de comprendre leurs caractéristiques et leurs modes de fonctionnement.

Il existe plusieurs manières de concevoir des agents, mais peu importe l'architecture adoptée, un agent peut toujours être vu comme une fonction liant ses perceptions à ses actions. Plus précisément, un agent perçoit l'environnement à l'aide de ses capteurs et il agit sur son environnement à l'aide de ses effecteurs. Ce qui différencie les différentes architectures d'agents, c'est la manière dont les perceptions sont liées aux actions. Nous distinguons deux grandes familles d'agents les agents réactifs et agents cognitifs.

2.1 Agents réactifs :

Tout d'abord, nous tenons à mentionner que les agents peuvent être conçus de plusieurs manières différentes. Ce sont les caractéristiques de l'environnement qui influencent le choix de conception. Si les agents sont dans un environnement en constants changements et qu'ils doivent, par conséquent, réagir très vite aux événements de l'environnement, alors une architecture réactive est appropriée.

Autrement dit, un tel agent ne fait ni délibération ni planification, il se contente simplement d'acquiescer des perceptions et de réagir à celles-ci en appliquant certaines règles

prédéfinies. Étant donné qu'il n'y a pratiquement pas de raisonnement, ces agents peuvent agir et réagir très rapidement.

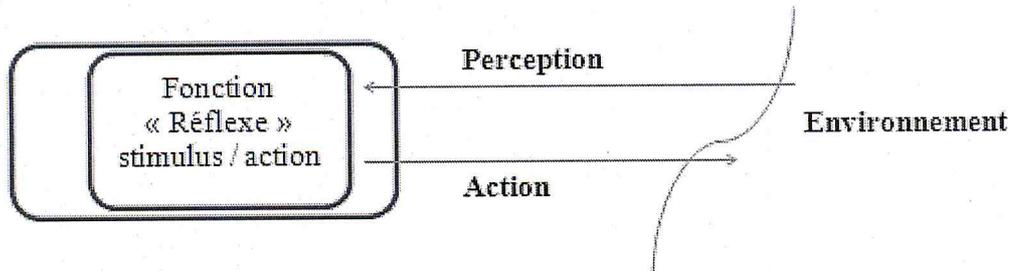


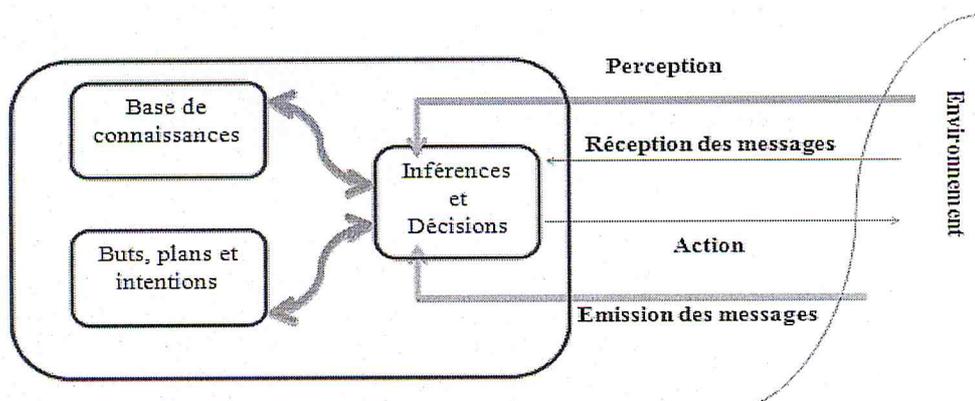
Fig.10 : Modèle d'un agent réactif

Les agents réactifs sont les plus simples à implémenter, et leur efficacité a été vérifiée dans plusieurs applications. Ils présentent toutefois des limites dues aux points suivants :

- l'agent n'a pas la représentation mentale de l'environnement, et doit choisir ses bonnes actions à partir des données locales uniquement,
- le comportement global de l'agent ne peut pas être facilement prévu, par conséquent, il n'est pas toujours possible de concevoir un comportement d'agent en fonction du but spécifié.

2.2 Agents cognitifs :

C'est le premier modèle d'agents qui a été proposé. Il est nommé aussi agent délibératif. Il est basé sur l'IA symbolique, et il permet de planifier les actions d'un agent au sein de son environnement. En effet, les agents cognitifs sont capables à eux seuls de réaliser des opérations relativement complexes. Généralement, ils coopèrent les uns avec les autres pour atteindre un but commun (résolution d'un problème, une tâche complexe, etc.). Ils possèdent un ensemble de représentations explicites (sur l'environnement, sur les autres agents et sur eux mêmes) décrits dans une base de connaissances sur laquelle ils peuvent raisonner. Ils réagissent en fonction de leurs connaissances, leurs buts, de leurs échanges d'informations avec les autres agents et de la perception de l'environnement (voir figure 11). Ils sont dotés de moyens et mécanismes de communication pour gérer les interactions avec d'autres agents (coopération, coordination et négociation).



L'agent cognitif, en terme de BDI pour Beliefs (croyances), Desires (désirs) et Intentions (intentions), postule que ses actions, tout comme celles d'un être humain, doivent être dictées par des représentations abstraites du monde, incluant la représentation de ses propres capacités, ses buts ainsi que ceux des autres agents. Ce modèle est fondé sur des extensions de la logique, et est basé sur les états mentaux suivants :

- les croyances : les connaissances de l'agent sur l'environnement.
- les désirs : les « options » de l'agent, c'est à dire les différents objectifs vers lesquels l'agent peut vouloir transiter.
- les intentions : les états vers lesquels l'agent a choisi de transiter.

Contrairement aux agents réactifs, les agents cognitifs sont beaucoup plus complexes et plus difficiles à mettre en œuvre. Chaque agent se fonde sur ses propres compétences de façon isolée pour résoudre un problème, ce sont les principales limites de cette architecture.

Pour faire face aux inconvénients de ces deux modèles, les chercheurs ont combiné ces deux facettes, opposées mais complémentaires, de la conception des agents. Cette combinaison fait apparaître les agents hybrides.

2.3 Agents hybrides :

Les agents hybrides sont conçus pour combiner des capacités réactives à des capacités cognitives, ce qui leur permet d'adapter leur comportement en temps réel à l'évolution de l'environnement.

Dans ce type d'architecture, les agents sont conçus comme étant composés de niveaux hiérarchiques qui interagissent entre eux. Chaque niveau gère un aspect du comportement de l'agent:

- Au plus bas niveau de l'architecture, on retrouve habituellement une couche purement réactive qui prend ses décisions en se basant sur des données brutes en provenance de l'environnement.
- La couche intermédiaire fait abstraction des données brutes et travaille plutôt avec une vision qui se situe au niveau des connaissances de l'environnement.
- La couche supérieure se charge des aspects sociaux de l'environnement, c'est-à-dire du raisonnement tenant compte des autres agents.

2.4 Agents mobiles :

Les agents mobiles sont des entités logicielles autonomes qui peuvent suspendre leur exécution sur une machine et migrer avec leur code, variables et état vers une autre machine où ils reprennent leur exécution.

Un agent mobile peut ainsi communiquer ou coopérer avec un agent chargé d'informer les agents de la localisation de ses attributs et méthodes, ceci dispense cet agent et rendre publique toutes les informations et données le concernant.

Les raisons qui poussent à recourir aux agents mobiles sont les suivantes :

- Réduction des couts de communication.
- L'insuffisance de ressource locale.
- Une coordination plus facile : il est plus simple de coordonner un nombre de requêtes distantes et d'en collecter simplement les résultats.
- Un traitement asynchrone : on peut initialiser ces agents mobiles et faire autre chose pendant que ceux-ci s'exécutent.

3. Définition de Système Multi-Agents :

La plupart du temps, un agent n'est pas seul dans son environnement, il y a d'autres agents présents autour de lui. Les agents doivent, par conséquent, être capables d'interagir entre eux. Ils peuvent soit coexister, coopérer ou être en compétition.

Un système où évoluent plusieurs agents est appelé système multi agent et il possède généralement plusieurs caractéristiques intéressantes, comme le parallélisme, la robustesse et l'extensibilité.

Définition proposée par à Ferber [Fer, 95]:

Parmi les différentes définitions des systèmes multi-agents nous retiendrons celle de Ferber qui le définit comme un système composé des éléments suivants :

- Un Environnement E** : C'est-à-dire un espace disposant d'une métrique.
- Un ensemble d'objets O** : Ces objets sont situés, c'est-à-dire que, pour tout objet, il est possible, à un moment donné, d'associer une position dans E. Ces objets sont passifs, c'est à dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents.
- Un ensemble A d'agents** : qui sont des objets particuliers ($A \subseteq O$), lesquels représentent les entités actives du système.
- Un ensemble de relations R** : qui unissent des objets (et donc des agents) entre eux.

- Un ensemble d'opérations **Op** : permettant aux agents A de percevoir, produire, consommer, transformer et manipuler des objets de O.
- Des opérateurs : chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'Univers.

Définition proposée par Jennings, Sycara et Wooldridge [Woo et Jen, 95] :

Cette définition présente un SMA comme étant un ensemble d'agents en interaction afin de réaliser leurs buts ou d'accomplir leurs tâches.

Les interactions peuvent être directes par l'intermédiaire des communications, comme elles peuvent être indirectes via l'action et la perception de l'environnement.

Les interactions peuvent être mises en œuvre dans un but de :

.La coordination : organiser la résolution d'un problème de telle sorte que les interactions nuisibles soient évitées, ou que les interactions bénéfiques soient exploitées.

.La négociation : parvenir à un accord acceptable pour toutes les parties concernées.

.La coopération : travailler ensemble à la résolution d'un but commun [Chai et al., 01]. Cette approche est particulièrement bien adaptée à la simulation des systèmes complexes dont le fonctionnement global émerge des actions des individus.

Un SMA est généralement caractérisé ainsi :

- Chaque agent dispose d'informations ou de capacités de résolution de problèmes limités (ainsi, chaque agent a un point de vue partiel).
- Il n'y a aucun contrôle global du système multi-agent.
- Les données sont décentralisées.
- Le calcul est asynchrone.

4. Les Avantages des SMA : [Mar, 03]

L'utilisation des Systèmes Multi-Agents (SMA) présente une série d'avantages:

- **Système dynamique** : Les agents sont structurés afin d'exercer une influence sur chacun pour faire évoluer le système dans son ensemble (système dynamique).

- **Souplesse de l'outil informatique** : Le modèle Multi-Agent est rendu opérationnel grâce à une implémentation informatique qui n'impose alors aucune exigence analytique spécifique mais l'emploi des langages informatiques évolués (Langage Orienté Objet (LOO)) qui permet de développer le programme de façon modulaire.

- Une résolution distribuée de problèmes.

- Les SMA peuvent « répondre » de l'échec individuel d'un des éléments, sans dégrader le système dans son ensemble.

Ils possèdent également les avantages traditionnels de la résolution distribuée et concurrente de problèmes [Cha et al., 01]:

- **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 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.

Après une revue générale sur les SMA, nous nous intéressons maintenant à la partie essentielle dans les SMA, les interactions entre les différents agents dans le système.

5. Les Interactions entre agents [Fer, 95]:

Une interaction est une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques. Les agents interagissent le long d'une suite d'évènements pendant lesquels les agents sont d'une certaine manière en contact les uns avec les autres, que ce contact soit direct ou qu'il s'effectue par l'intermédiaire d'un agent ou de l'environnement.

Les systèmes multi agents ont surtout l'avantage de faire intervenir des schémas d'interaction sophistiqués. Ils peuvent ainsi coexister, être en compétition ou coopérer.

S'ils ne font que coexister, alors chaque agent ne considère les autres agents que comme des composantes de l'environnement, au même titre que toutes les autres composantes.

S'ils sont en compétition, alors le but de chaque agent est de maximiser sa propre satisfaction, ce qui se fait généralement aux dépens des autres agents. La situation de compétition la plus fréquente se produit lorsque plusieurs agents veulent utiliser ou acquérir la même ressource. Les agents doivent donc pouvoir communiquer entre eux pour résoudre le conflit.

S'ils sont en coopération, alors le but des agents n'est plus seulement de maximiser sa propre satisfaction mais aussi de contribuer à la réussite du groupe. Les agents travaillent

ensemble à la résolution d'un problème commun. Dans ce type de système, les agents communiquent ensemble, à l'aide de messages plus ou moins sophistiqués, dans le but d'améliorer la performance du groupe. Ils peuvent s'échanger des informations sur l'environnement pour augmenter leurs perceptions individuelles, ou bien se transmettre leurs intentions pour que les agents puissent avoir une idée de ce que les autres font.

Les principales situations d'interaction peuvent être classées par rapport à trois critères :

Buts compatibles et incompatibles : Les objectifs d'interaction des agents sont évidemment au centre des problématiques de l'interaction, les agents ont-ils des buts concordants, c'est-à-dire allant dans le même sens, ou bien leurs objectifs sont ils contradictoires, voire opposés ?

On dira que des agents sont dans une situation de coopération ou d'indifférence (coexiste) si leurs buts sont compatibles et dans une situation d'antagonisme (compétition) dans le cas contraire.

Relations aux ressources : On entend par ressources tous les éléments environnementaux et matériels utiles à la réalisation d'une action. Les conflits apparaissent essentiellement lorsque plusieurs agents ont besoin des mêmes ressources en même temps et en même endroit.

Capacités des agents par rapport aux tâches : Est-ce qu'un agent peut réaliser seul une tâche ou bien a-t-il besoin des autres pour parvenir à son but ?

Une des interactions la plus fréquente dans le monde des agents et que nous allons l'utilisées dans notre travaille est la coopération.

5.1 Coopération :

Il existe plusieurs points de vue sur la coopération, selon que l'on considère que la coopération est une attitude des agents qui décident de travailler en commun ou que l'on se pose comme un observateur qui interprète a posteriori les comportements en les qualifiant de coopératifs ou non à partir de critères sociaux, tels que l'interdépendances des actions ou le nombre de communication effectuées.

La coopération comme attitude intentionnelle : Dans le premier cas, la coopération est caractéristique d'une attitude des agents. On dit que des agents coopèrent s'ils s'engagent dans une action commune après avoir identifié et adopté un but commun.

La coopération du point de vue de l'observateur : est une qualification de l'activité d'un ensemble d'agents par un observateur extérieur qui n'aurait pas accès aux états mentaux des agents.

La coopération a pour avantages d' :

- Accomplir des tâches impossibles à réaliser seul.
- Améliorer la productivité de chacun des agents.
- Augmenter le nombre de tâches réalisées dans un délai imparti ou diminuer le temps de réalisation d'une tâche.
- Améliorer l'utilisation des ressources grâce à la rapidité des traitements.

5.2 Les méthodes de coopération :

Les moyens de coopération qui nous appellerons méthodes sont :

Le regroupement et la multiplication :

Elle consiste tout simplement pour les agents à se rapprocher physiquement, c'est-à-dire à constituer soit un bloc plus ou moins homogène dans l'espace, soit un réseau de communication permettant à plusieurs agents de se comporter comme s'ils étaient physiquement les uns à côté des autres.

La communication :

Le système de communication qui lie plusieurs agents agit comme un système nerveux qui met en contact des individus par fois séparés. La communication en effet agrandit les capacités perspectives des agents en leur permettant de bénéficier des informations et du savoir faire des autres agents. Les communications sont indispensable à la coopération et il est difficile de concevoir un système d'agents coopérants s'il n'existe pas un système permettant aux agents d'échanger des informations ou véhiculer des envois de messages.

La spécialisation :

La spécialisation est le processus par lequel des agents deviennent de plus en plus adaptés à leurs tâches. Il est souvent difficile de faire des agents qui soient spécialisés dans toutes les tâches. La réalisation performante d'une tâche suppose souvent des caractéristiques structurelles et comportementales qui ne peuvent permettre d'effectuer d'autres tâches avec efficacité.

La collaboration par partage de tâches et de ressources :

La collaboration consiste à travailler à plusieurs sur un projet, une tâche commune. Nous appellerons collaboration l'ensemble des techniques permettant à des agents de répartir des tâches, des informations et des ressources de manière à réaliser une œuvre commune. Résoudre un problème de collaboration consiste donc à répondre à la question « qui fait quoi ? » par rapport à un travail donné.

Le plus connu de ces systèmes est le réseau contractuel ou «Contract Net Protocol » (CNP).

Ce protocole se compose de cinq étapes et se déroule entre clients et fournisseurs, de la manière suivante :

- Le client sollicite des propositions de fournisseurs potentiels en émettant un appel d'offre décrivant la tâche et les conditions nécessaires à son exécution.
- Les fournisseurs potentiels évaluent les appels d'offres provenant des différents clients.
- Les fournisseurs potentiels fournissent des propositions pour accomplir la tâche.
- Le client évalue les propositions et fait son choix sur un, ou plusieurs fournisseurs pouvant accomplir la tâche. Dans certains cas il ne peut pas y avoir de choix.
- Finalement client et fournisseur établissent un contrat pour l'exécution de la tâche.

Coordination d'actions :

Gérer un ensemble d'agents suppose l'exécution d'un certain nombre de tâches supplémentaires qui ne sont pas directement productives mais qui servent simplement à faire en sorte que les actions productives puissent s'accomplir dans les meilleures conditions.

La résolution de conflit par arbitrage et par négociation :

L'arbitrage et la négociation sont deux moyens utilisés par les systèmes multi agents pour résoudre les conflits et empêcher que des désaccords entre individus ne se transforment en luttes ouvertes et que le système dans son entier ne dégrade ses performances.

L'arbitrage conduit à la définition de règles de comportement qui agissent comme des contraintes sur l'ensemble des agents.

Lorsque des agents cognitifs entrent en conflit, on préfère souvent ne pas recourir à un arbitre, mais laisser plutôt les agents résoudre eux-mêmes le conflit par la recherche d'un accord au cours d'un processus de négociation.

5.3 Communication entre agents :

Les communications, dans les SMA comme chez les humains, sont à base des interactions et de l'organisation sociale. Sans communication, l'agent n'est qu'un individu isolé, sourd et muet aux autres agents. C'est parce que les agents communiquent qu'ils peuvent coopérer, coordonner leurs actions, réaliser des tâches en commun et devenir ainsi de véritables êtres sociaux.

5.4 Stratégies de communication :

A. Communication directe :

Lorsqu'un agent désire envoyer un message, celui-ci est pris par le canal de communication et apporte le message directement à son destinataire sans tenir compte de la distance et sans donner la possibilité à d'autres agents de le recevoir.

Il y a deux types de communication directe :

Communication point à point :

Lorsque le destinataire est connu de l'émetteur, ce dernier lui adresse des messages en particulier et ainsi instaure une communication individuelle. On dit alors que les communications sont effectuées selon le mode point à point. C'est ce type de communication qui est généralement le plus employé dans les agents cognitifs.

Communication par diffusion :

Si le destinataire n'est pas connu de l'émetteur, le message est envoyé sous un mode appelé diffusion (broadcasting) à tout un ensemble d'agent, ces derniers étant liés à l'émetteur par une relation de voisinage (proximité dans l'environnement, connexité dans un réseau, etc.). Ce type de communication est très utilisé dans les systèmes dynamiques dans lesquels des agents peuvent apparaître ou disparaître. Il s'emploie en particulier dans des protocoles d'allocations de tâches comme celui du réseau contractuel.

B. Communication indirecte :

Communication indirecte via l'environnement :

- **Propagation des signaux:** Dans ce cas, la communication est restreinte à un ensemble fini de signaux ayant des interprétations fixées. Cette classe de communication s'adresse à des agents de type réactifs.

Ces agents n'ont pas de communication « explicite ». Celle-ci a lieu par propagation de stimuli via l'environnement telles les phéromones pour les fourmis, et uniquement par des actions sur l'environnement, par exemple la consommation de ressources de l'environnement, mais sans communication ajoutée.

- **Trace :**
 - un agent laisse tomber des "morceaux radioactifs" en faisant ainsi des traces.
 - un agent suit une trace fait diminuer la trace jusqu'à la disparition de la trace.

Communication indirect via le tableau noir :

Est moins utilisé dans les SMA. C'est typiquement le mécanisme de communication des « petites annonces ». Un agent, s'il désire communiquer, place son message dans un espace commun, appelé tableau d'affichage, tableau noir, visible par tous les agents (ou tous ceux d'une classe particulière). Ce mode combine les caractéristiques des messages directs et ce des messages diffus.

5.5 Les actes de langage :

Un acte de langage désigne un ensemble des actions intentionnelles effectuées au cours d'une communication. On peut distinguer les principaux types d'actes suivants :

1. Les assertifs : servent à donner une information sur le monde en affirmant quelque chose (ex : il fait beau).
2. Les directifs : sont utilisés pour donner des directives au destinataire (ex : Lave toi les mains).
3. Les promissifs : engagent le locuteur à accomplir certains actes dans l'avenir (ex : je te promets de t'envoyer des cartes postales).
4. Les expressifs : servent à donner au destinataire des indications concernant l'état mental du locuteur (ex : je suis heureux).

5. Les déclaratifs : accomplissent un acte par le fait même de prononcer l'énoncé (ex : je déclare la séance ouverte).

5.6 Les langages de communication :

Les premiers travaux de définition d'un langage de communication entre agents (en anglais : **ACL**, Agent Communication Language), standard remontent à 1990, quand la **DARPA** (Defense Advanced Research Project Agency) a initié le **Knowledge Sharing Effort (KSE)**. En effet, si l'on veut partager la connaissance, il faut s'accorder sur un langage commun : c'est pourquoi le KSE s'est tout de suite tourné vers la conception d'un ACL.

La compréhension entre agents passe par différents niveaux :

1. Tout d'abord, il faut que les agents parlent le même langage, ou bien qu'il existe des traducteurs bijectifs d'un langage dans un autre. On est ici au **niveau syntaxique**.
2. Mais il faut également que les mêmes objets, les mêmes concepts aient la même signification pour tous les agents. Il est donc nécessaire que tous les agents partagent une **ontologie** commune, c'est à dire une définition des concepts associée aux relations qui existent entre eux.
3. Enfin, les agents doivent être capables de dialoguer entre eux, pas au niveau du simple envoi de messages, mais à un niveau supérieur : ils doivent pouvoir s'informer, se poser des questions, rechercher d'autres agents, etc. Pour ce faire, les agents ne peuvent pas se contenter d'appeler une procédure; ils doivent spécifier un état désiré dans un **langage déclaratif** : c'est ici qu'intervient l'ACL.

KIF : Knowledge Interchange Format : [Che, 09]

Le niveau purement syntaxique (niveau 1) peut être pris en charge par **KIF** (Knowledge Interchange Format). En effet, un message KIF peut être vu comme une **s-expression**, une s-expression étant elle-même soit un atome, soit une liste de s-expressions entourée de parenthèses.

KQML: Knowledge Query and Manipulation Language

Le langage **KQML** se positionne au 3^{ème} niveau de notre classification. Il est complètement indépendant à la fois de la syntaxe et de l'ontologie sous-jacente.

Un message KQML se compose :

- d'un **performatif**, qui représente l'en-tête du message.

- d'un certain nombre de **couples attribut-valeur**.

Voici quelques exemples de performatifs, parmi les 37 définis dans la norme de 1997 :

Performatif	Catégorie	Signification
Tell	Assertif	L'agent émetteur indique qu'il considère que le contenu du message est vrai.
Ask-if	interrogatif	L'agent émetteur demande à l'agent récepteur s'il considère que le contenu du message est vrai.
Advertise	Promissifs	L'agent émetteur promet qu'il exécutera le code contenu dans le message si jamais il le reçoit à l'avenir.

Tab.3 : Quelques exemples de performatifs

Voyons maintenant quels sont les attributs les plus courants :

Attribut	Signification
: content	Contenu du message
: language	Langage dans lequel le contenu est exprimé
: ontology	Ontologie sous-jacente nécessaire pour comprendre le contenu du message
: sender	Nom de l'émetteur du message
: receiver	Nom du récepteur du message

Tab.4 : Les attributs les plus courants de KQML

D'un point de vue plus théorique, on peut considérer qu'un message KQML se compose de trois couches :

- couche de **contenu** : le contenu du message, c'est à dire la valeur de l'attribut : content.
- couche de **communication** : informations qui permettent l'acheminement du message, par exemple les valeurs des attributs :sender et : receiver.
- couche de **message** : composée du performatif et des valeurs des attributs qui décrivent le message, comme :language ou :ontology.

La FIPA et son ACL

En 1999, la FIPA (Foundation for Intelligent Physical Agents) a défini un ACL, pour répondre principalement à une critique assez vigoureuse contre KQML : ses performatifs sont trop nombreux, donc redondants, et de plus ils ne couvrent pas en entier le champ des performatifs envisageables.

L'ACL de la FIPA ressemble beaucoup à KQML, mais il ne comporte que 22 performatifs. De plus, il est défini de façon beaucoup moins ambiguë.

Conclusion :

Nous avons vus dans cette première partie la définition des SMA ainsi que les Agents, les principales utilités, par la suite nous allons entamer la partie qui introduit les notions liées aux services web.

Les Services Web :

Introduction :

Les applications logicielles d'entreprises deviennent de plus en plus distribuées, complexes et coûteuses en efforts de gestion. Parallèlement, l'espérance de vie de ces applications ne cesse d'être remise en question par le dynamisme qui caractérise de nos jours le monde de l'entreprise. Pour remédier à ces insuffisances, il faut développer de nouveaux concepts technologiques contribuant au développement d'applications plus étendues et plus complexes sur un marché en mutation constante.

Dans ce contexte, se situe les architectures distribuées, partant de 2-niveaux, passant par le 3-niveaux et aboutissant aux N-niveaux, dans les objectifs sont :

- La simplification du fonctionnement du système (i.e. offrir une vue homogène d'un monde hétérogène, définir et respecter des normes et des standards).
- La favorisation de la réutilisation de composants logiciels.
- L'augmentation de l'exigence non-fonctionnelle.
- La garantie de l'évolutivité fonctionnelle et technique du système (i.e. de préserver une certaine indépendance du système vis-à-vis des évolutions techniques potentielles de chacun de ses éléments).

1. Notion de service :

La notion de service logiciel correspond à une fonctionnalité offerte par un composant applicatif et dont l'interface est indépendante de la plate-forme ou de la technologie utilisée pour le développement du composant.

Un Service est un composant logiciel distribué, exposant les fonctionnalités à forte valeur ajoutée d'un domaine métier [Bar, 10].

Les principales caractéristiques d'un service logiciel sont les suivantes :

- Il est auto-descriptif et permet la réutilisation dans le sens où il fournit une description de son interface.
- Il est autonome, complet et cohérent dans la mesure où le service contient toute sa logique d'exécution ce qui lui permet de s'exécuter indépendamment des autres services.
- Le service peut être localisé dans la mesure où il possède une adresse ce qui permet de l'invoquer dynamiquement.

- Les services communiquent par l'envoi et la réception des messages.
- Les échanges de messages sont organisés sous forme de contrats d'échange entre le fournisseur du service et le consommateur de ce service, on trouve trois types de contrat.
 - Lié à la syntaxe du service (opération, messages d'entrée, messages de sortie, ...)
 - Lié à la sémantique du service (définition de règles et de contraintes d'usage, ...)
 - Lié à la qualité de service (temps de réponse attendu, procédures en cas de panne, temps de reprise après interruption, ...)
 - S'appuie sur des standards d'interopérabilité pour faciliter le dialogue (WSDL)
- Composabilité : Un service doit fonctionner de manière modulaire et non pas intégrée.
- Abstraction :
 - le service est une boîte noire il ne doit contenir que les données pertinentes à son invocation.
 - seul le contrat exposé au consommateur du service est connue.
 - Le fonctionnement interne du service ne doit pas être visible ni la logique métier, ni l'implémentation.
- Sans état : pour minimiser la consommation de ressources.
- Couplage lâche : un service n'a pas besoin de connaître a priori les autres services.

2. L'Architecture Orientée Services (AOS) :

L'architecture orientée services (AOS) est une approche architecturale permettant la création des systèmes basés sur une collection de services développés dans différents langages de programmation, hébergés sur différentes plates-formes avec divers modèles de sécurité et processus métier (Un processus métier est un enchaînement de tâches métiers réalisées par un ensemble d'acteurs de l'entreprise et produisant une valeur ajoutée pour celle-ci).

Chaque service représente une unité autonome de traitement et de gestion de données, communiquant avec son environnement à l'aide de messages. Les échanges de messages sont organisés sous forme de contrats d'échange [Ben, 09].

L'idée maîtresse de l'architecture orientée service est que tout élément du système d'information doit devenir un service

- Identifiable.

- Documenté.
- Fiable.
- Indépendant des autres services.
- Accessible.
- Et réalisant un ensemble de tâches parfaitement définies.

2.1 Composants de l'Architecture Orientée Services :

L'AOS est axée autour de trois concepts fondamentaux, à savoir : [Izz, 06]

- Le fournisseur de services (service provider) : Le fournisseur permet l'accès à son service à travers une interface.
- Le client de services (service requestor) : Le client désigne une personne, un serveur ou une autre application qui accède au service et l'invoque à travers son interface.
- L'annuaire de publication (service registry) : L'annuaire joue le rôle d'intermédiaire entre le fournisseur et le client. Les fournisseurs y enregistrent leurs services, et les clients y cherchent le service satisfaisant leurs besoins.

Ainsi que trois types d'opérations basiques :

- Publier (publish)
- Rechercher (find)
- Interagir (bind).

Toute entité d'une SOA peut jouer un ou plusieurs de ces rôles et peut effectuer une ou plusieurs de ces types d'opérations.

2.2 Principe de fonctionnement :

Dans le modèle SOA, chaque service est défini par un fournisseur. Le fournisseur de services publie la description de son service dans des registres de services spécialement conçus en vue d'être interrogés par des clients. Les clients de services localisent leurs besoins en termes de services en effectuant des recherches sur le registre de services. Une fois le service localisé, le client récupère sa description du registre et sur la base des informations fournies dans la description du service, le client interagit alors avec le service en vue de l'exécuter [Ben, 09].

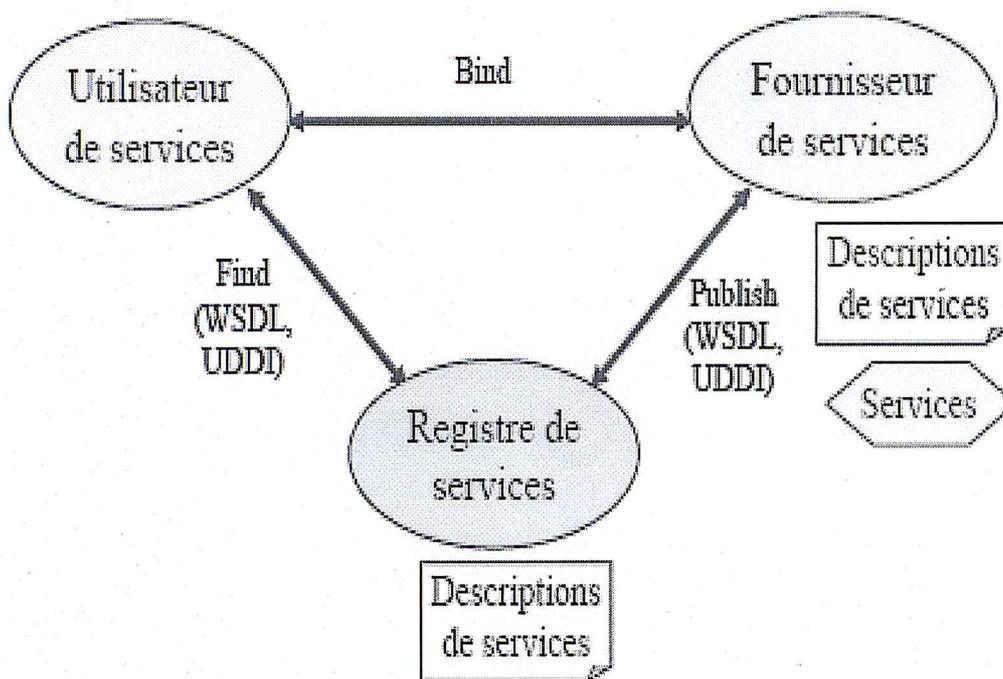


Fig.12 : Principe de fonctionnement de l'AOS

2.3 Avantages de l'Architecture Orientée Service :

L'AOS encapsule plusieurs avantages bénéfiques pour le domaine de la technologie d'information et de communication

- ❖ Elle offre la simplicité à travers les concepts de décomposition, de découplage et de réutilisation.
- ❖ elle participe à la réduction du coût de développement des grands projets et rend le développement plus efficace.
- ❖ L'architecture orientée service est apparue pour palier les limites des architectures distribuées.

Cette architecture n'est pas simplement une mode, elle se place plutôt dans la continuité logique des multiples tentatives de distribution de traitements, de répartition de données, d'intégration d'applications, d'homogénéisation du système d'informations, etc. L'adoption de la SOA a été grandement facilitée par l'émergence opportune de la technologie des services web et leurs standards bien définis.

La technologie des services web représente la technologie la plus utilisée pour migrer vers ce type d'architectures.

3. Les services web :

Les services web sont des applications auto-descriptives, modulaires et faiblement couplées qui fournissent un modèle de programmation et de déploiement d'applications, basé sur des normes, et s'exécutant au travers de l'infrastructure web.

Les Services Web sont considérés comme le résultat de convergence du Web et des Technologies objet distribuées.

Ils sont définis comme des services logiciels déployés sur l'Internet.

Les Services Web constituent l'instanciation la plus importante du modèle SOA dans le domaine industriel.

Les principaux apports de la technologie des Services Web sont notamment :

- l'universalité.
- Standardisation.
- couplage faible entre les services.
- la virtualisation qui permet de définir une indépendance vis à vis de la localisation et de l'implémentation des services.

3.1. Pour quoi les services web :

- Ils offrent un moyen standard d'interopérabilité entre différentes applications qui s'exécutent sur une variété de machine et plateforme.
- Ils sont caractérisés par leur grande interopérabilité et extensibilité, ainsi qu'une description interprétable/compréhensible automatiquement par la machine grâce au standard XML.
- Ils peuvent être combinés d'une façon faiblement couplée afin de réaliser des opérations complexes.
- Les services web sont des compléments aux programmes et applications existantes, développées en différents langages de programmation, et servent de pont pour que ces programmes communiquent entre eux [Ben, 09].
- les services web permettent d'interfacer des systèmes d'information hétérogènes et ont pour atouts :
 - Un faible couplage avec les technologies employées en interne.
 - Une grande flexibilité de mise à jour des systèmes employés de part et d'autre.
 - L'emploi de protocoles réseau simples, répandus et bénéficiant d'implémentations dans toutes les technologies majeures.

3.2. Infrastructures des services web :

Les services offerts par l'infrastructure des services web concernent essentiellement deux aspects fondamentaux :

- Un service de communication qui permet l'échange de données entre les services web.
- Un ensemble de services techniques destinés à automatiser le processus de localisation et d'invocation des composants.

L'originalité de l'infrastructure des services web consiste à mettre en place ces services en se basant exclusivement sur les protocoles les plus répandus d'Internet tels que http (HyperText Transfer Protocol), MIME (Multipurpose Internet Mail Extensions), XML. L'infrastructure des services web s'est concrétisée autour de trois spécifications considérées comme des standards, à savoir SOAP, UDDI et WSDL. Nous les détaillons dans ce qui suit.

A. SOAP pour la communication:[Ben, 09].

Simple Object Access Protocol (SOAP) aussi appelé (Service Oriented Access Protocol) est le protocole qui assure l'échange de messages dans les architectures orientes services. Du fait qu'il est basé sur XML, il permet l'échange de données structurées indépendamment des langages de programmation ou des systèmes d'exploitation. SOAP permet l'échange d'informations dans un environnement décentralisé et distribué, comme Internet, indépendamment du contenu du message. Il peut être employé dans tous les styles de communication : synchrones ou asynchrones, point à point ou multipoints. SOAP utilise principalement les deux standards HTTP et XML :

- HTTP comme protocole de transport des messages SOAP. Il constitue un bon moyen de transport en raison de sa popularité sur le web.
- XML pour structurer les requêtes et les réponses, indiquer les paramètres des méthodes, les valeurs de retours, et les éventuelles erreurs de traitements.

SOAP se base sur XML pour encoder les données. Les messages échangés via ce protocole jouissent donc des avantages que lui procure le langage XML pour Structurer les données.

Le message est englobé dans une **enveloppe** qui permet de spécifier la version de **SOAP** cette **enveloppe** est divisés en 2 parties :

- **Entête(Header) :** Optionnel, permet de spécifier certaines directives pour le traitement du message
- **Corps (body) :** offre un mécanisme simple d'échange des informations mandataires destinées au receveur du message SOAP. Cette partie contient les paramètres fonctionnels tels que le nom de l'opération à invoquer, les paramètres d'entrés et de sortis ou des rapports d'erreur

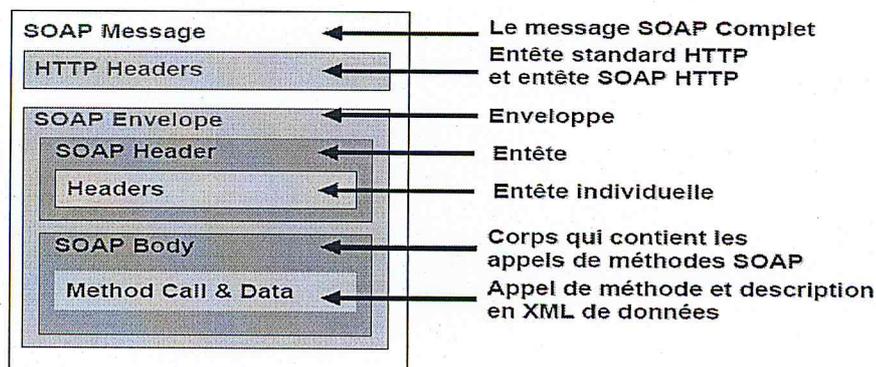


Fig.13 : Structure du message échangé dans SOAP

B. WSDL pour la Description : [Izz, 06]

WSDL est l'abréviation de (Web Services Description Language) est un standard du W3C qui a été créé dans le but de fournir une description unifiée des services web basé sur la syntaxe XML, pour décrire les méthodes et paramètres des Web Services invocables par le biais de messages au format SOAP.

- Il permet de décrire le service web, en précisant les méthodes disponibles, les formats des messages d'entrée et de sortie, et comment y accéder.

Un WSDL a la structure suivante :

- Définitions** : Il contient le nom du service décrit et les name spaces faisant référence aux types utilisés dans le document.
 - Types**: pour la définition de types de données échangées.
 - Message**: qui définit d'une manière abstraite (noms et types) le format des données échangées
 - Opération**: Elle permet de définir les opérations invoquées sur le service web. Chaque opération est décrite de façon optionnelle à l'aide d'un, deux ou trois messages :
 - Le message reçu par le service web lorsqu'il est sollicité par une requête (input).
 - Le message émis en réponse par le service web (output)
 - Le message d'erreur retourné par le service web en cas de problème (fault)
- A noter qu'un ou plusieurs messages précédents peuvent être absents à la description d'une opération.
- Port Type** (appelé Interface) : qui représente un ensemble d'opérations correspondant chacune à un message entrant ou sortant.
 - Binding** (Rattachement ou liaison) : spécifie une liaison entre un <portType> et un protocole concret (SOAP, HTTP...).
 - Port**: qui est un point ou une adresse d'accès au service web.
 - Service**: cette section caractérise un service web en énumérant ses divers points d'accès ou ports.

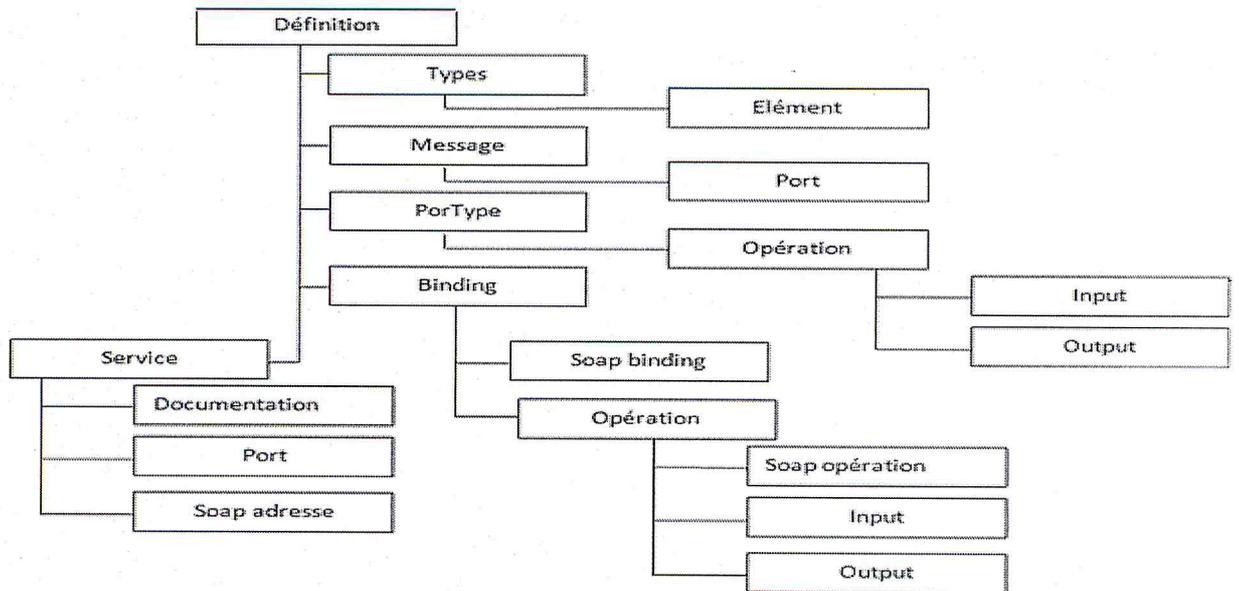


Fig.14: Structure de WSDL

C. UDDI pour la Découverte :

Universal Description, Discovery and Integration(UDDI) est une norme d'annuaire de services web appelée via le protocole SOAP.

Pour publier un nouveau service web, il faut générer un document appelé Business Registry, il sera enregistré sur un UDDI Business RegistryNode. Le Business Registry comprend trois parties :

- Pages blanches : noms, adresses, contacts, identifiants des entreprises enregistrées.
- Pages jaunes : informations permettant de classer les entreprises, notamment l'activité, la localisation, etc.
- Pages vertes : informations techniques sur les services proposés.

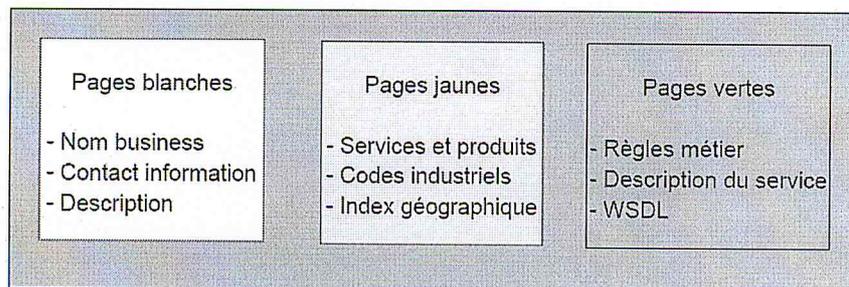


Fig.15 : Structure d'UDDI

3.3. Invocation des services web :

L'invocation de service web passe par les étapes suivantes :

Etape 1 : inscription du service web.

Etape 2 : recherche de service web.

Etape 3 : envoyer la description WSDL de service au client à travers le protocole SOPA.

Etape 4 : Invocation du service Web par le client en utilisant les informations de WSDL envoyé par UDDI.

Etape 5 : après l'exécution de la requête de client le fournisseur envoie le résultat au client à travers le protocole SOAP.

Le schéma suivant explique ces étapes :

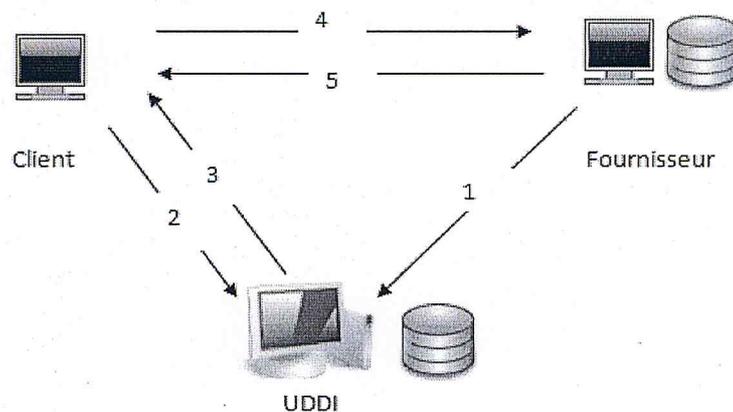


Fig.16 : Etapes d'invocation de service web

3.4. Bus de services d'entreprise (ESB) [Izz, 06] :

L'ESB (Enterprise Service Bus) ou Bus de Services d'Entreprise est un concept qui a été défini en 2003 par le Gartner Group.

Pour que les composants d'une SOA puissent communiquer entre eux, il est nécessaire de disposer d'une infrastructure permettant de les mettre en relation. C'est le rôle de l'ESB

Donc l'ESB propose une solution d'intégration distribuée basée sur un message-oriented middleware (MOM) et fournissant des services de transformation des données, de routage et ce en s'appuyant sur l'utilisation systématique des standards des Services Web (SOAP, WSDL, UDDI) et les normes WS-*.

L'ESB est centrée sur la notion de bus qui permet d'implémenter une solution d'intégration distribuée. Le bus permet aux différents services métiers (encapsulant des applications d'entreprise) de communiquer et de coopérer. Le bus se base sur certains de services de base comme :

- le moteur d'orchestration qui joue le rôle de service d'orchestration permet d'exécuter des processus basés sur BPEL4WS;
- le service de localisation qui permet de trouver de façon transparente les services;
- les services utilitaires qui sont des services techniques et qui sont sollicités par les services métier (ex: services permettant le routage, la transformation, le support des transactions, etc.);
- les services d'infrastructure qui sont des services qui permettent de fournir un support système et d'infrastructure aux services métiers (ex: services liés à la sécurité et au monitoring, etc.).

Conclusion :

Un service web est une entité logicielle permettant la communication et l'échange de données entre applications et systèmes hétérogènes dans des environnements distribués.

Il vise à assurer l'interopérabilité, et ce à travers une présentation standardisée des services offertes et d'un protocole standard de communication permettant de structurer les messages échangés entre les composantes logicielles. Il offre aussi une spécification de publication et de localisation de services. La particularité des services web réside dans le fait qu'elle utilise la technologie Internet comme infrastructure pour la communication entre les composants logiciels. Les architectures orientées service constituent un paradigme de conception et de réalisation applicable aux différents niveaux d'interaction d'un système communicant. Elles sont amenées à jouer un rôle de plus en plus important dans la conception des futurs systèmes de communication et leurs applications logicielles. Le concept du bus logiciel se généralise au niveau de couches hautes jusqu'à la couche application.

4. Vers un couplage SMA/SW :

Après l'analyse que nous avons réalisée sur les (SMA) et les Services web nous pouvons dire que dans un domaine distribué qui demande une certaine intelligence dans la résolution de problème nécessite l'utilisation d'une méthode qui gère la distribution de problème sur les différents sites ainsi que l'aspect intelligence qui se trouve au niveau de résolution de problème donc utiliser seulement les services web ne peut pas répondre à nos besoins qui demande une certaine intelligence car les services web sont des méthodes statiques qui n'ont aucune intelligence, par contre si nous utiliserons seulement les (SMA) on va rencontrer des problèmes au niveau de l'interopérabilité à cause de l'hétérogénéité des systèmes utilisés dans les différents sites, donc au lieu de s'intéresser au vrai problème qui est comment partager les ressources de manière optimale, on doit aussi prendre en considération les spécifications techniques de chaque site.

Donc l'utilisation d'une méthode ou une approche de couplage entre les SMA et les services web est indispensable vu les avantages qui les caractérisent ensemble.

Afin de maximiser l'intégration de systèmes hétérogènes et de favoriser une interopérabilité au moindre coût, il est nécessaire de minimiser les exigences en termes de spécifications.

Comme nous avons vu précédemment, les SW :

- Se présentent comme un support crédible permettant à des applications d'exposer leurs fonctionnalités au travers d'interfaces standardisées et de plus en plus éprouvées.
- Ont pour vocation de favoriser une architecture orientée services, intégrant des systèmes hétérogènes complexes, fortement distribués et pouvant coopérer sans recourir à une intégration spécifique et coûteuse.

Aujourd'hui le rapprochement entre les SMA et les services Web devient nécessaire dans des environnements distribués qui nécessitent des raisonnements intelligents. Ce rapprochement se manifeste selon différentes directions dont les principales sont les suivantes : [Seg et al., 08]

- L'utilisation des SMA en tant qu'entité médiatrice dans le modèle fonctionnel des services Web.
- L'utilisation des services Web comme cadre architectural et technologique pour mettre en place des SMA accessibles à travers le Web. Dans ce type d'application on retrouve des applications agents qui offrent leurs capacités à travers des services Web.

Ici on distingue deux catégories qui diffèrent essentiellement par leur conception :

- Une conception intégrée : ce sont des services Web développés suivant un modèle agent afin de réaliser des tâches complexes.
- Une conception découplée : à partir d'un SMA donné a priori, une couche à base de services Web rend les capacités de l'agent accessibles à travers le Web que ce soit à d'autres agents de SMA ou à des applications clientes traditionnelles.

Conclusion

Dans ce chapitre nous avons parlé des Systèmes Multi-Agents, les avantages, inconvénients. Nous avons parlé aussi des services web qui se présentent comme un cadre architectural et technologique et enfin nous avons discuté entre ces deux aspects (SMA/SW).

PARTIE 2 :

*CONCEPTION ET
REALISATION.*

Chapitre I :

Conception et Modélisation du Système

I. Conception du système :

Dans cette partie nous allons présenter la structure de notre système à base des SMA et expliquer son fonctionnement à l'aide de différents diagrammes qui sont présentés en AUML.

I.1. Analyse de l'environnement :

Nous avons vu dans le chapitre de l'état de l'art que notre domaine est distribué où les différentes pharmacies sont complètement isolées ce qui rend le coût de gérer les ressources pharmaceutique très important, de ce fait on assiste souvent à des situations de sur stockage d'une ressource dans un site S1 au moment où un autre site S2 souffre en terme de disponibilité de la même ressource. Cette situation peut engendrer :

- Des problèmes liés à la santé des patients (un sous stockage)
 - Un stock dépasse un seuil non toléré affecte négativement sur la qualité de soins, un sous stockage provoque une rupture de stock.
- Problèmes financiers (des coûts supplémentaires et inutiles par fois)
 - des coûts supplémentaires: pour les produits périmés (stockage, destruction)

Donc la solution que nous allons présenter est une solution qui prend en considération les problèmes cités ci-dessus, et elle vise à :

- Créer une fédération inter pharmacies : une fédération est un ensemble de pharmacies qui décident de partager leurs ressources afin :
 - D'équilibrer les ressources de stock pharmaceutique pour éviter toute rupture de stock ou sur stockage.
 - D'exploiter de manière optimale les ressources existantes dans la fédération des pharmacies.
- Assurer l'interopérabilité entre les ressources : ce qui rend les ressources partageable entre les différents sites pharmaceutique.

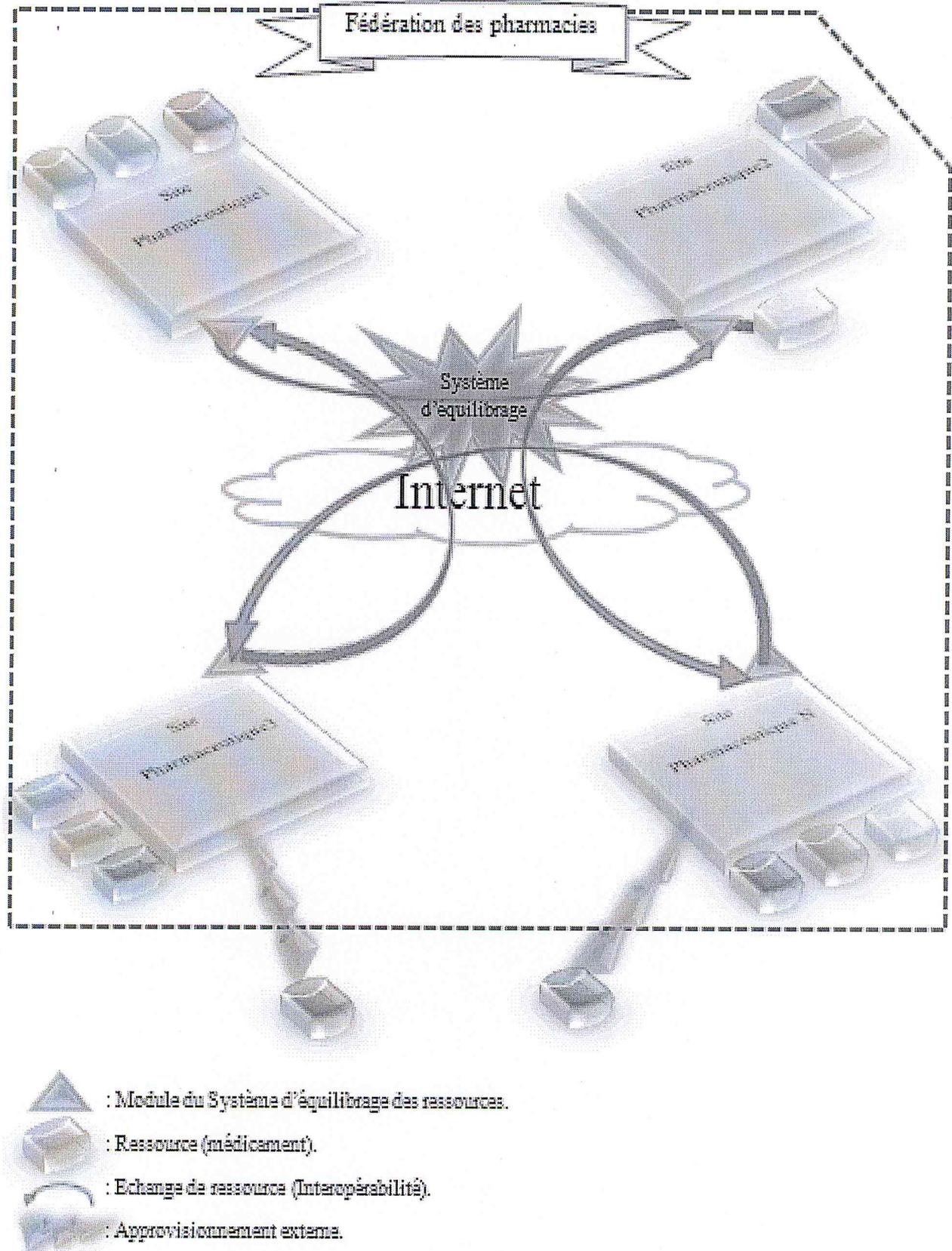


Fig. 17 : Schéma de fédération des pharmacies

I.2. Organisation du système Multi-Agents :

Afin d'assurer l'interopérabilité des ressources, nous associons :

- Un agent gestionnaire de stock (A.G.S) Pour chaque site pharmaceutique.
- Un agent ressource (A.R) pour chaque catégorie de médicament au niveau de chaque site.
- Un agent fédérateur(A.F) pour chaque catégorie de médicament qui existe dans toute la fédération.

Le système Multi-Agents est composé d'agents de ressource et d'agents fédérateur qui interagissent entre eux, la figure 18 représente le diagramme qui illustre l'organisation de notre système multi-agents :

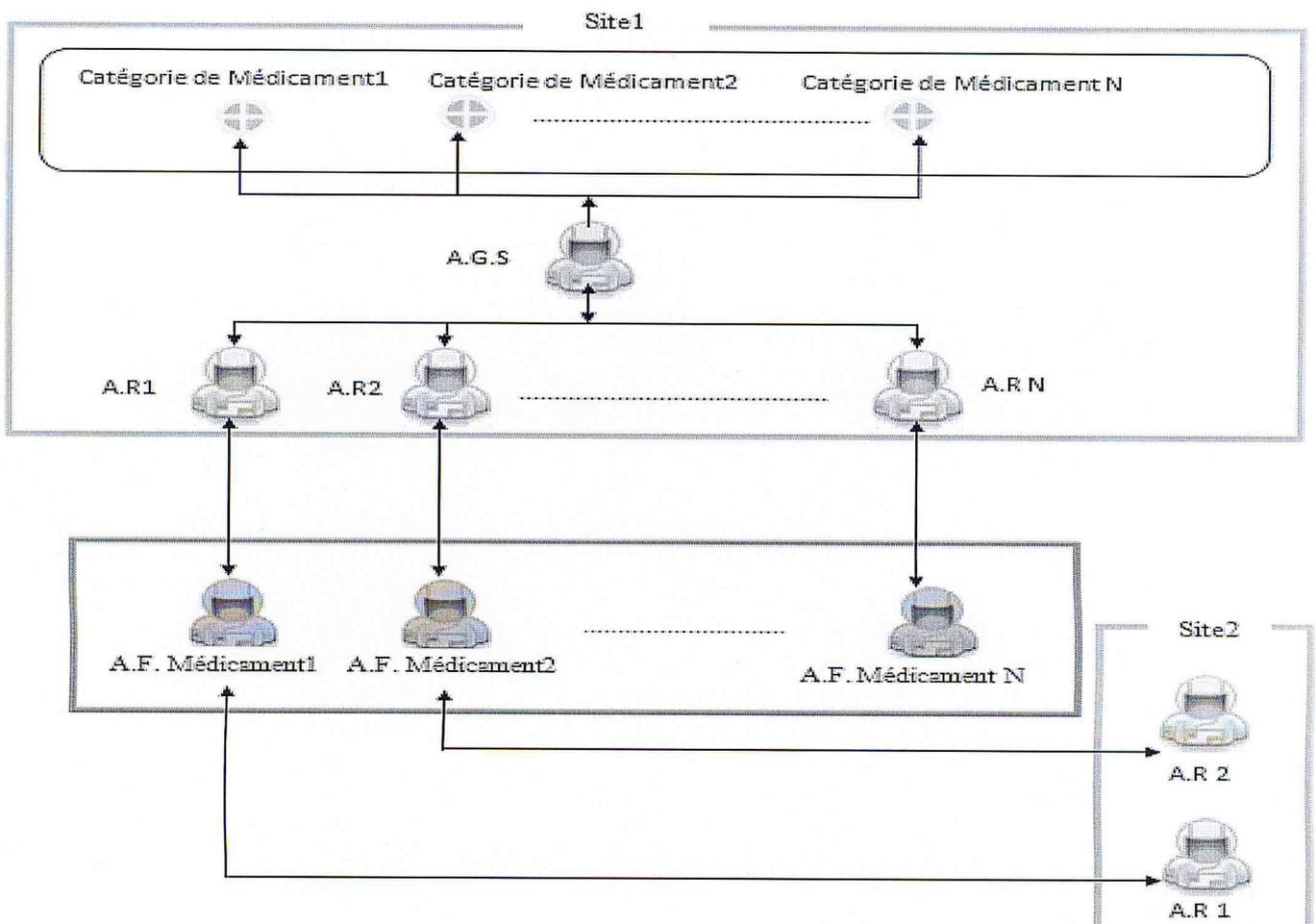


Fig. 18 : Système Multi-Agents

Comme la figure le montre nous retrouvons :

- Pour chaque site pharmaceutique le nombre des agents de ressource égale au nombre des types (catégories) de médicaments existants dans le site.
- Le nombre des agents fédérateurs égale au nombre de catégories de médicaments dans toute la fédération

I.3. Typologie des agents :

Les agents que nous avons définis précédemment sont des agents hybrides qui ont la capacité de réagir avec le facteur de temps et avec les messages reçus, ces agents ont des connaissances et des rôles précis qui coordonnent le processus d'équilibrage de stock pharmaceutique.

Dans un site donné nous trouvons deux types agents :

- Agent Gestionnaire de Stock (A.G.S).
- Agent de Ressource (A.R).

Et via internet nous trouvons un autre type d'agent c'est :

- Agent Fédérateur (A.F).

I.3.1. Agent gestionnaire de stock(A.G.S) :

Il existe un seul agent gestionnaire par site.

-Compétences :

- Consulter le stock (La base de données) d'une façon continue et périodique.
- Guetter les informations pertinentes sur les différents médicaments existant dans le stock de site, à savoir : la consommation moyenne mensuelle (CMM), le stock de sécurité (SS), la quantité à stocker (stock max QS), la quantité actuelle de médicament (QA), et la date du dernier approvisionnement (DateDA).
- Envoyer à chaque agent de ressource les informations sur le médicament dont il gère.

-Connaissances : Il a des informations sur :

- Le site pharmaceutique dont il est associé (le nom, l'adresse IP, l'adresse géographique).
- Les agents de ressources qui existent dans ce site (Les noms des agents).

I.3.2 Agent de Ressource (A.R):

Chaque agent de ressource gère une catégorie de médicament, et il peut jouer deux rôles différents : Agent de Ressource Client (A.R.C).

Agent de Ressource Fournisseur (A.R.F).

-Compétences : l'A.R a des compétences communes entre les deux rôles :

- La réception des informations sur le médicament dont il gère.
- à la base de ces informations reçues, l'A.R calcule l'état de stock.
- Selon l'état de stock l'A.R joue un rôle d'un client ou d'un fournisseur.

Cas de Rupture : L'A.R devient un client avec les compétences suivantes :

- Elaborer une demande contenant : la ressource demandée, la quantité, l'agent client demandeur, le site, la date de l'élaboration de la demande.
- Envoyer la demande à l'Agent Fédérateur au quel il est rattaché.
- Répondre aux demandes de confirmation de la demande effectuée par accord ou désaccord.
- Demander une offre d'un A.R.F.

Cas de sur stockage : L'A.R devient un fournisseur d'offre avec les compétences suivantes:

- Elaborer une offre contient : la ressource offrir, la quantité, l'agent client fournisseur, le site, la date de l'élaboration de l'offre.
- Envoyer l'offre à l'Agent Fédérateur au quel il est rattaché.
- Répondre aux Appels d'offre. par accord ou désaccord.

-Connaissances : Il a des informations sur :

- Le site pharmaceutique dont il est associé (le nom, l'adresse IP, l'adresse géographique).
- Le médicament dont il gère (DCI : dénomination commune internationale).
- L'agent fédérateur au quel il est attaché (le nom, adresse IP).

-Buts : Le but principal de l'A.R dans la stratégie globale de notre système est de maintenir l'équilibre de ressource dont il le gère pour arriver à un équilibre global.

I.3.3 Agent Fédérateur (A.F):

C'est un agent coordinateur entre l'ensemble des agents de ressource des différents sites pharmaceutique qui gèrent la même catégorie de médicament.

-Compétences :

- La réception des offres et des demandes provenant des A.Rs.

À la réception d'une demande :

- Sélectionner les offres qui peuvent satisfaire la demande.
- Envoyer une appelle d'offre (demande de confirmation) aux A.R.Fs qui ont publiés les offres sélectionnées.
- La réception des réponses des A.R.Fs.
- Choisir la meilleure offre parmi les offres confirmées.
- Envoyer les informations sur l'offre à l'A.R.C.

À la réception d'une offre :

- Sélectionner les demandes qui demandent cette offre.
- Envoyer une demande de confirmation aux A.R.Cs pour confirmer leurs demandes.
- La réception des réponses des A.R.Cs.
- Choisir la meilleure demande (La plus urgence) parmi les demandes confirmées.
- Envoyer les informations sur l'offre à l'A.R.C.

-Connaissances : Il a des informations sur :

- Le médicament dont il gère (DCI : dénomination commune internationale).

-But : Il n'a pas un objectif spécifique dans la stratégie globale de notre système, parce qu'il joue le rôle d'intermédiaire entre l'A.R.C et l'A.R.F (associer une offre à une demande).

II. Modélisation du système :

Le langage AUML est choisi pour la modélisation des différentes interactions produites dans le système et illustré par des diagrammes, à savoir (une description d'AUML est donnée en annexe)

- Diagramme de cas d'utilisation.
- Diagramme de classe.
- Diagramme de séquence.
- Diagramme d'activité.

II.1. Diagramme de cas d'utilisation :

Ce diagramme représente les cas d'utilisation et le comportement du système vu par les acteurs y participant.

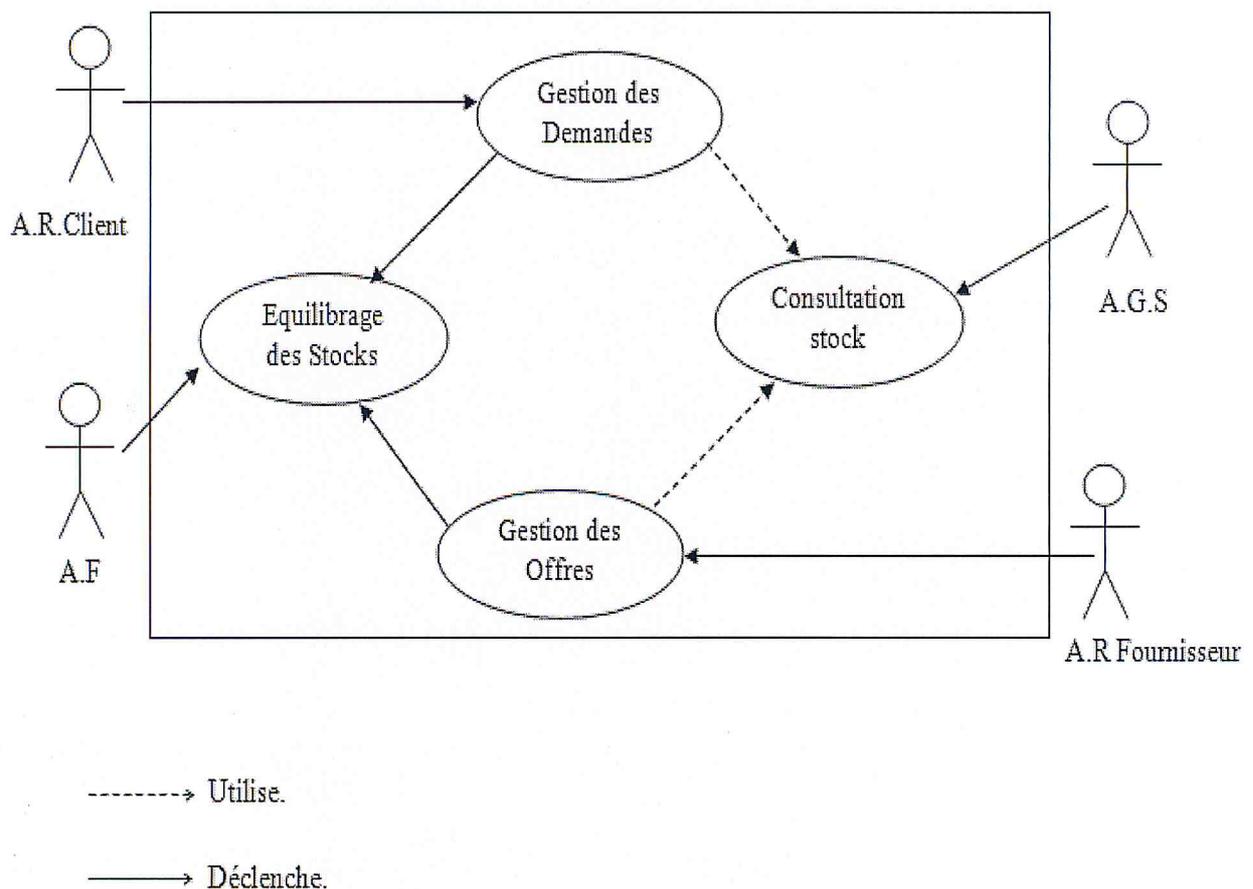


Fig.19 : Diagramme de cas d'utilisation

II.2. Les Scénarios :

Les scénarios expliquent le fonctionnement du système.

II.2.1. Scénario de consultation périodique de stock :

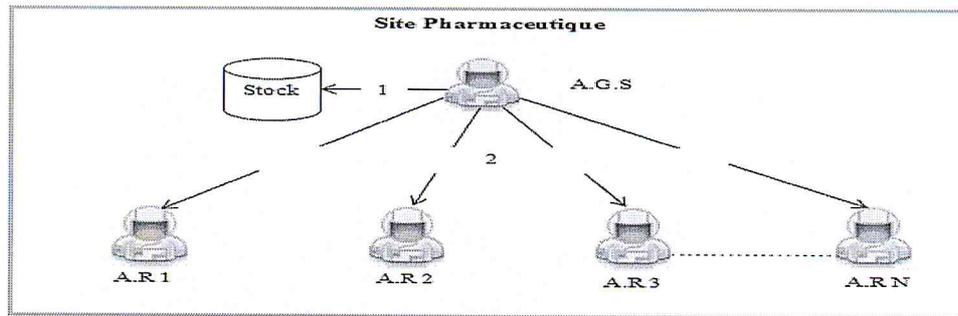


Fig.20 : Le scénario de Consultation de stock

1. L'A.G.S consulte le stock de manière régulière et extrait les données (Consommation Moyenne Mensuelle (CMM), Stock de Sécurité (SS), Quantité à Stocker (QS), Quantité Actuelle (QA), Date Dernière Approvisionnement (DateDA)) qui concernent chaque médicament.
2. L'A.G.S envoie à chaque A.R les informations sur le médicament dont il gère.

II.2.2. Scénario de gestion des demandes :

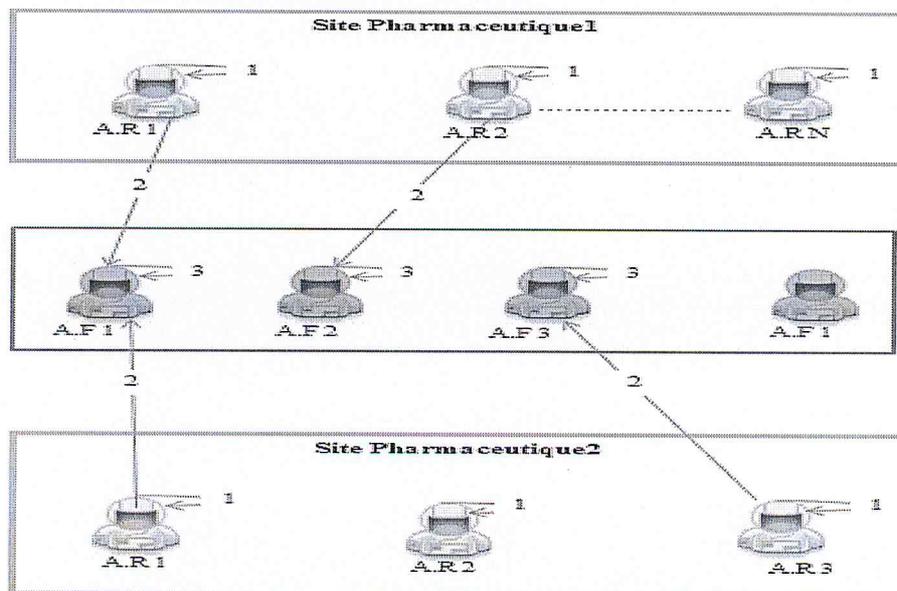


Fig.21 : Le scénario de gestion des demandes

1. L'A.R vérifie l'état de stock et guette la situation de sous stockage, il élabore une demande contenant : la ressource demandée, la quantité, l'agent client demandeur, le site, la date de l'élaboration de la demande.
2. Chaque A.R.C envoie sa demande à l'A.F au quel il est attaché.
3. L'A.F lance le processus d'équilibrage de stock.

II.2.3. Scénario de gestion des offres :

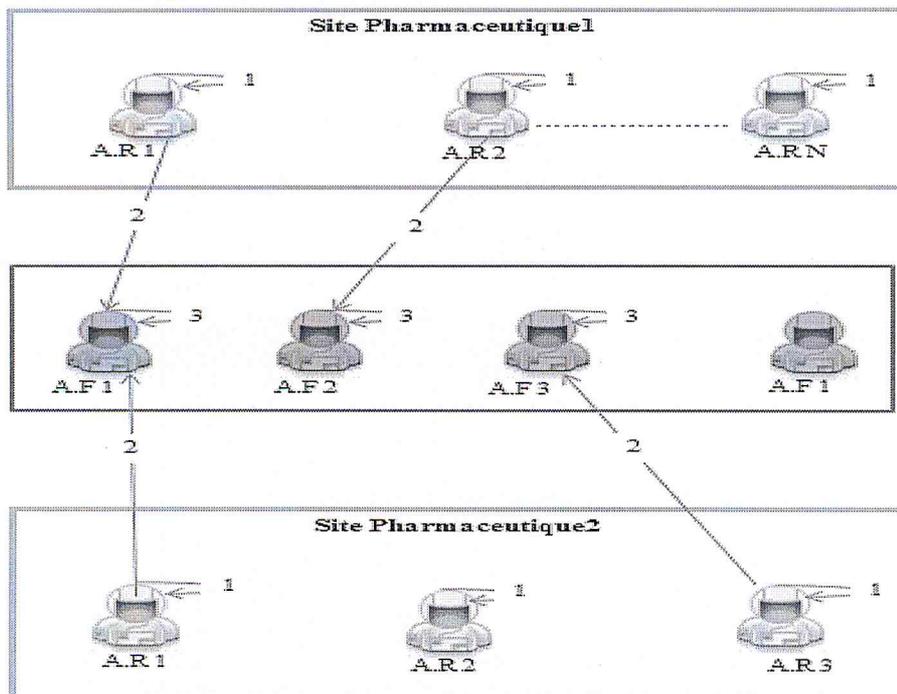


Fig.22 : Le scénario de gestion des offres

1. L'A.R vérifie l'état de stock et guette la situation de sur stockage, il élabore une offre contenant : la ressource à offrir, la quantité, l'agent client fournisseur, le site, la date de l'élaboration de l'offre.
2. Chaque A.R.F envoie son offre à l'A.F au quel il est attaché.
3. L'A.F lance le processus d'équilibrage de stock.

II.2.4. Scénario globale d'équilibrage des stocks pharmaceutique (Suite une demande):

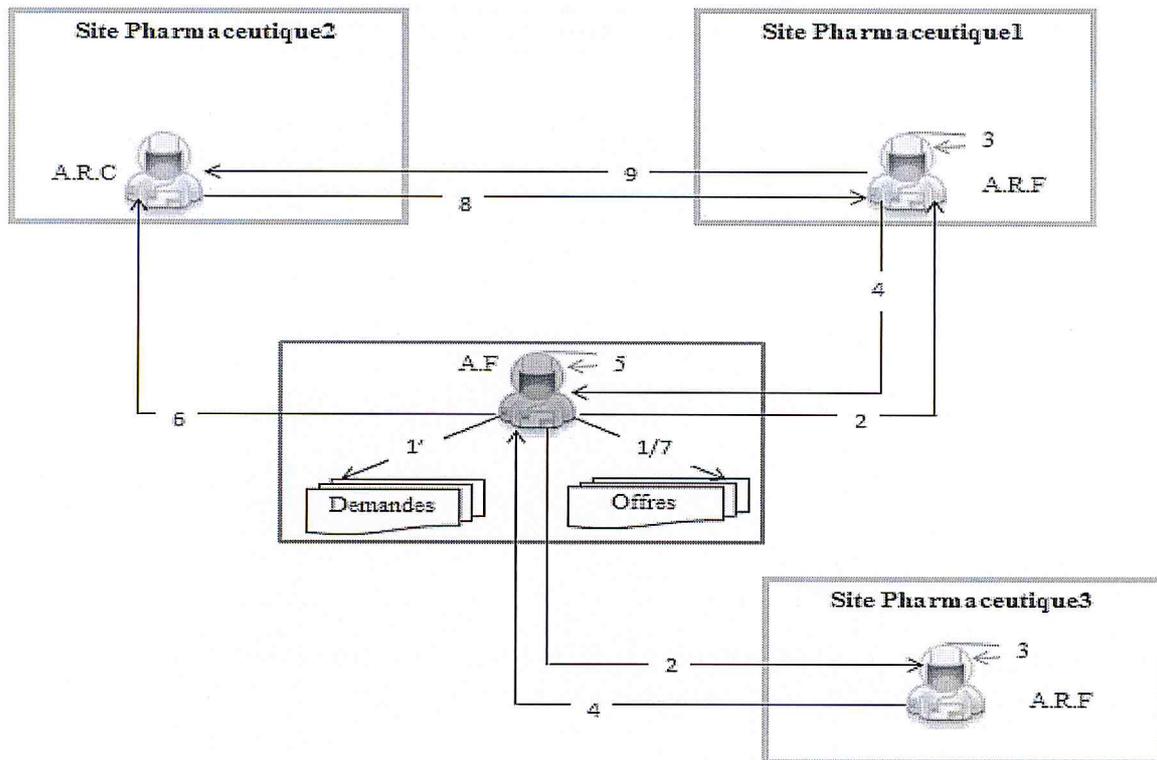


Fig.23 : Le scénario d'équilibrage de stock (Suite à une demande)

N° Action	Action
1	à l'arrivé d'une demande l'A.F consulte les offres existantes et sélectionne les serveurs potentiels s'il existe si non (1') il enregistre la demande.
2	L'A.F demande la confirmation des offres (lance un appel d'offre pour les offres sélectionnées).
3	L'A.R.F prend la décision concernant la demande (accepter ou refuser).
4	L'A.R.F envoie la décision à son A.F
5	L'A.F choisit la meilleure offre parmi les offres confirmées.
6	L'A.F envoie les informations sur l'offre choisit à l'A.R.C.
7	L'A.F Supprime l'offre choisie.
8	L'A.R.C demande l'offre de l'A.R.F.
9	L'A.R.F répond à la demande (Envoie de l'offre).

Tab.5 : Le scénario d'équilibrage de stock (Suite à une demande)

II.2.5. Scénario globale d'équilibrage des stocks pharmaceutique (Suite à une offre):

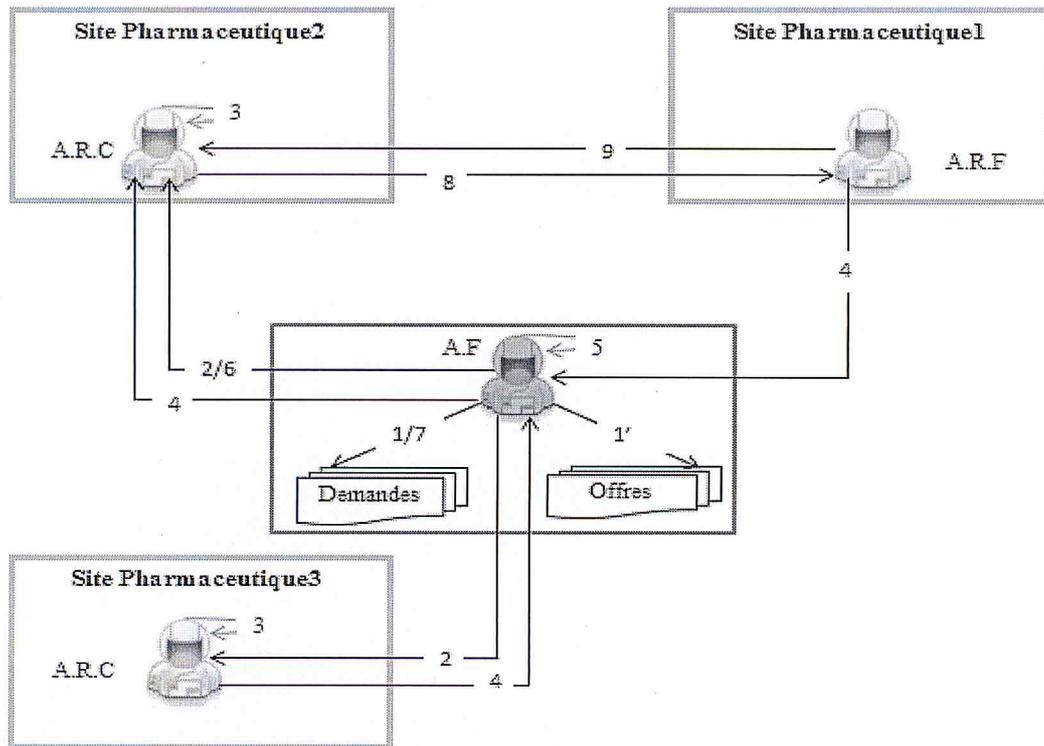


Fig.24 : Le scénario d'équilibrage de stock (Suite à une offre)

N° Action	Action
1	à l'arrivé d'une offre l'A.F consulte les demandes existantes et sélectionne les clients potentiels s'il existe si non (1') il enregistre l'offre.
2	L'A.F demande la confirmation des demandes (pour les demandes sélectionnées).
3	L'A.R.C prend la décision concernant l'offre (accepter ou refuser).
4	L'A.R.C envoie la décision à son A.F
5	L'A.F choisit la meilleure demande parmi les demande confirmées.
6	L'A.F envoie les informations sur l'offre choisit à l'A.R.C.
7	L'A.F Supprime la demande choisit.
8	L'A.R.C demande l'offre de l'A.R.F.
9	L'A.R.F répond à la demande (Envoie de l'offre).

Tab.6 : Le scénario d'équilibrage de stock (Suite à une offre)

II.3. Diagramme de classe :

Ce diagramme représente la vue statique de la conception, il permet de capturer les agents et les interactions entre eux.

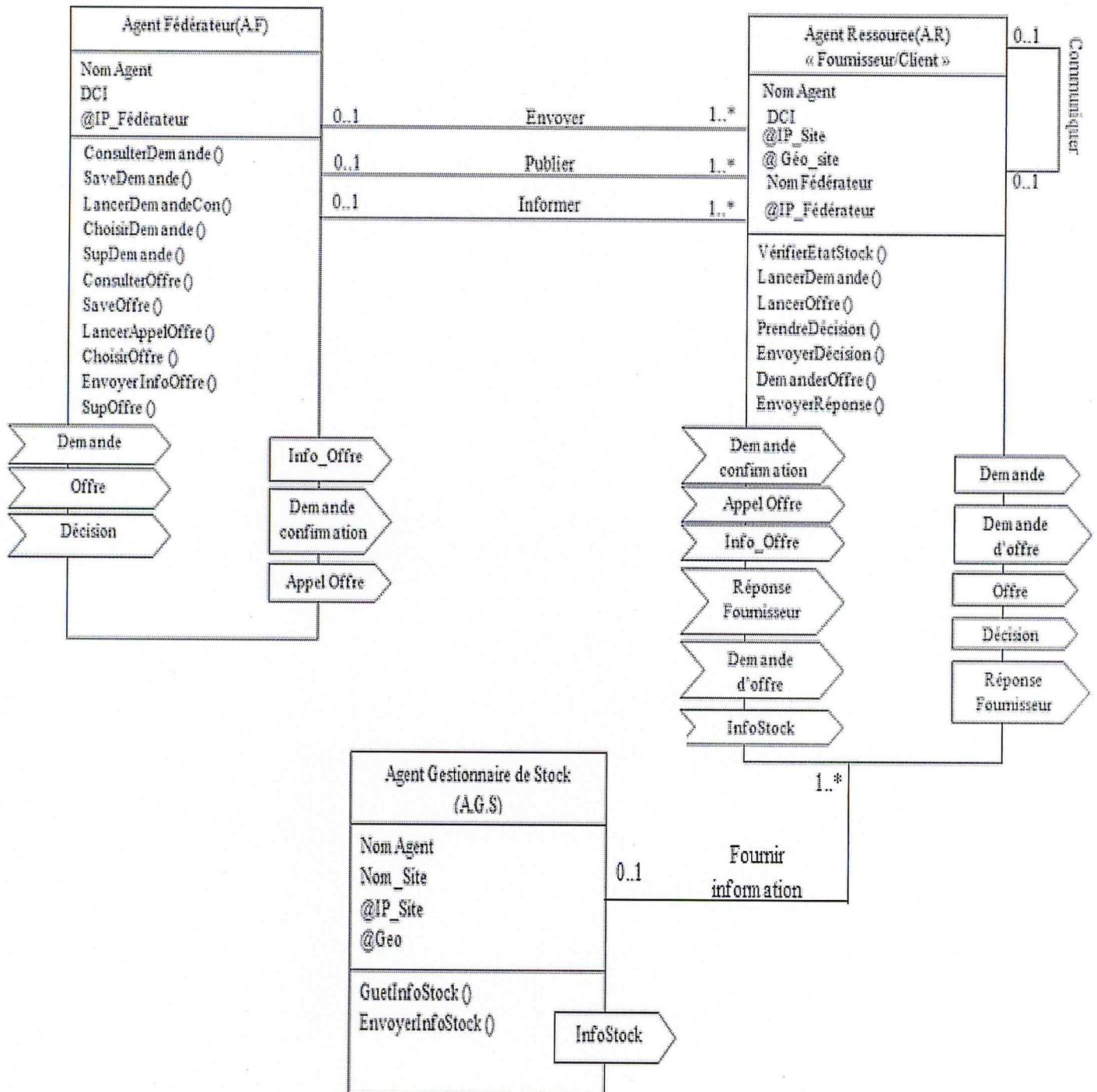


Fig.25 : Diagramme de classe d'agent

Les termes utilisés :

Terme	Signification
NomAgent	Le Nom d'agent.
DCI	Dénomination Communes International de médicament.
@IP_Fédérateur	Adresse IP de l'agent Fédérateur.
NomSite	Le nom de site
@IP_Site	Adresse IP de Site pharmaceutique.
@Géo Site	Adresse Géographique de site.
ConsulterDemande ()	Consulter et sélectionner la meilleure demande pour une offre.
EnrDemande ()	Enregistrer la demande.
LancerDemandeCon ()	Lancer une demande de confirmation d'une demande.
ChoisirDemande ()	Choisir une demande permet les demandes confirmées.
SupDemande ()	Supprimer la demande.
ConsulterOffre ()	Consulter et sélectionner les meilleures offres pour une demande.
EnrOffre ()	Enregistrer l'offre.
LancerApelOffre ()	Lancer un appel d'offre.
ChoisirOffre ()	Choisir une offre permet les offres confirmées.
EnvoyerInfoOffre	Envoyer les informations sur l'offre à l'agent client.
SupOffre ()	Supprimer l'offre.
VérifierEtatStock ()	Vérifier si le stock se trouve en état de rupture ou de sur stockage.
LancerDemande ()	Lancer une demande vers A.F.
LancerOffre ()	Publier une Offre chez l'A.F.
PrendreDécision ()	Répondre à la demande de confirmation ou l'appel d'offre.
EnvoyerDécision ()	Envoyer la décision à l'A.F.
DemanderOffre ()	Lancer une demande d'offre vers l'agent fournisseur.
EnvoyerRéponse ()	Répondre à la demande de l'offre (envoyer l'offre).
GuetterInfoStock ()	Vérifier le stock et guette les informations nécessaires sur le stock.
EnvoyerInfoStock ()	Envoyer l'information (l'actualité) sur le stock vers tous les A.R

Tab.7 : Tableau des termes utilisés

La liste des messages échangés:

Message	Signification
Demande	Envoyer par l'A.R.C vers l'A.F en cas de rupture.
Offre	Envoyer par l'A.R.F vers l'A.F en cas de sur stockage.
Décision	Envoyer par l'A.R.C ou l'A.R.F vers l'A.F pour répondre à la demande de confirmation de demande ou pou répondre à l'appel d'offre.
Info_offre	Envoyer par l'A.F vers l'A.R.C, il contient des informations sur l'offre choisit.
Demande confirmation	Envoyer par l'A.F vers les A.R.C potentiels pour demander la confirmation de leurs demandes.
AppelOffre	Envoyer par l'A.F vers les A.R.F potentiels pour demander la confirmation de leurs offres.
Demande d'offre	Envoyer par l'A.R.C vers l'A.R.F pour demander son offre.
Réponse Fournisseur	Envoyer par l'A.R.F vers l'A.R.C pour lui informer que la demande est satisfaite (envoyer offre).

Tab.8 : Tableau des messages échangés

Liste des relations :

Relation	Signification
Envoyer	Un A.R.C peut envoyer une demande à un seul A.F, comme il peut ne pas envoyer en cas normal. Un A.F peut envoyer une demande de confirmation à un ou plusieurs A.R, et un A.R peut envoyer la réponse à un et un seul A.F.
Publier	Un A.R.F peut publier une offre chez un seul A.F, comme il peut ne pas envoyer en cas normal.
Informers	Un A.F peut informer plusieurs A.Rs sur les offres existantes ou les demandes.
Communiquer	Un A.R peut rentre en communication avec un seul A.R pour se maitre d'accords sur une offres.
Fournir Information	Un A.G.S peut envoyer des informations sur le stock à un ou plusieurs A.R, et un A.R demande des informations sur le stock de un seul A.G.S.

Tab.9 : Tableau des relations

II.4. Diagramme de Séquence :

Le diagramme de séquence permet de représenter les interactions ainsi que les messages échangés entre les agents de notre système.

II.4.1. Consultation périodique de stock :

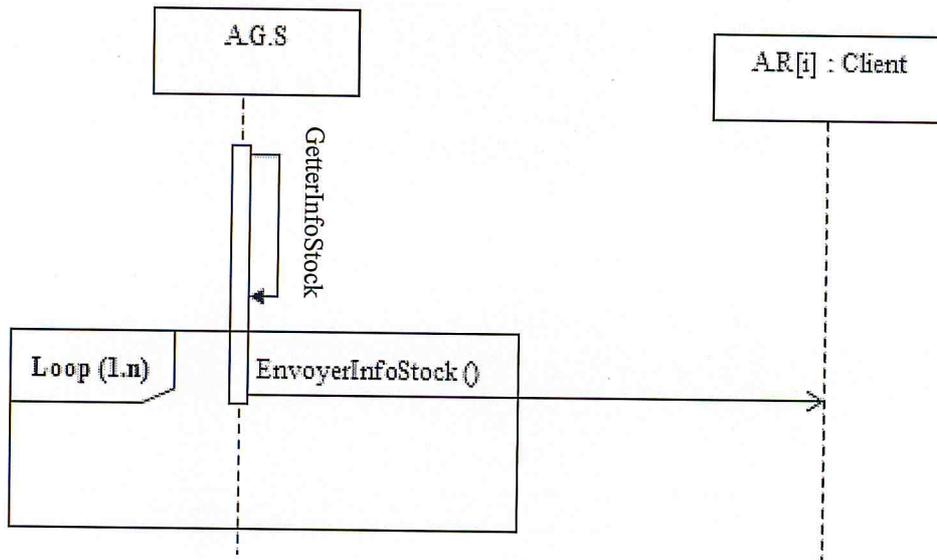


Fig.26 : Diagramme de Séquence « Consultation régulière de stock »

II.4.2. Gestion des demandes :

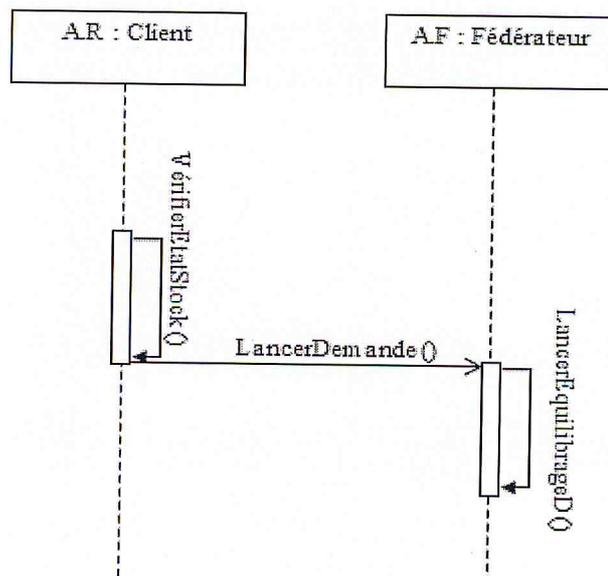


Fig.27 : Diagramme de Séquence « gestion des demandes »

II.4.3. Gestion des offres :

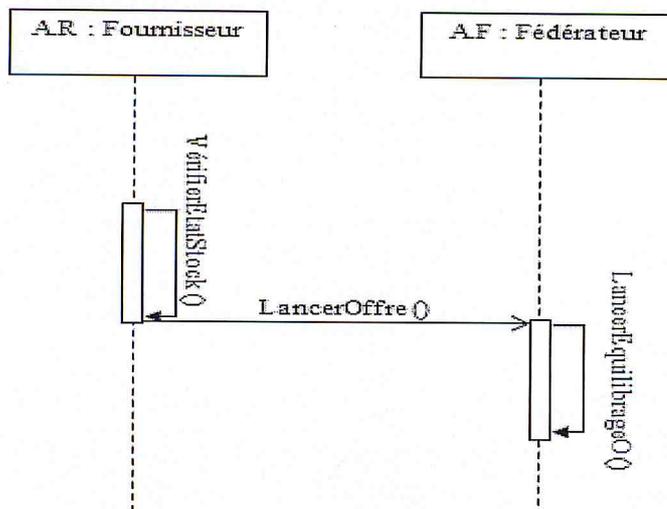


Fig.28 : Diagramme de Séquence « gestion des offres »

II.4.4. Équilibrage des stocks pharmaceutique (Suite à une demande):

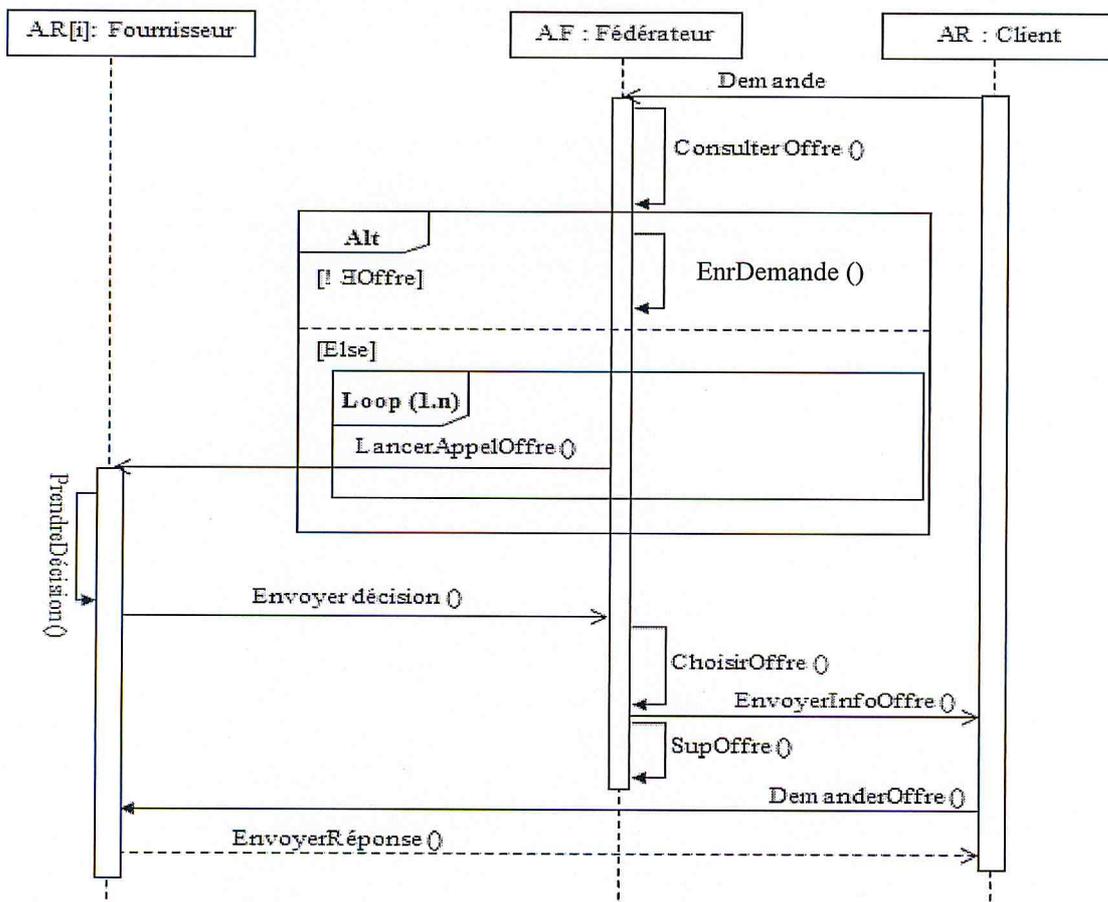


Fig.29 : Diagramme de Séquence «Suite à une demande »

II.4.5. Équilibrage des stocks pharmaceutique (Suite à une offre):

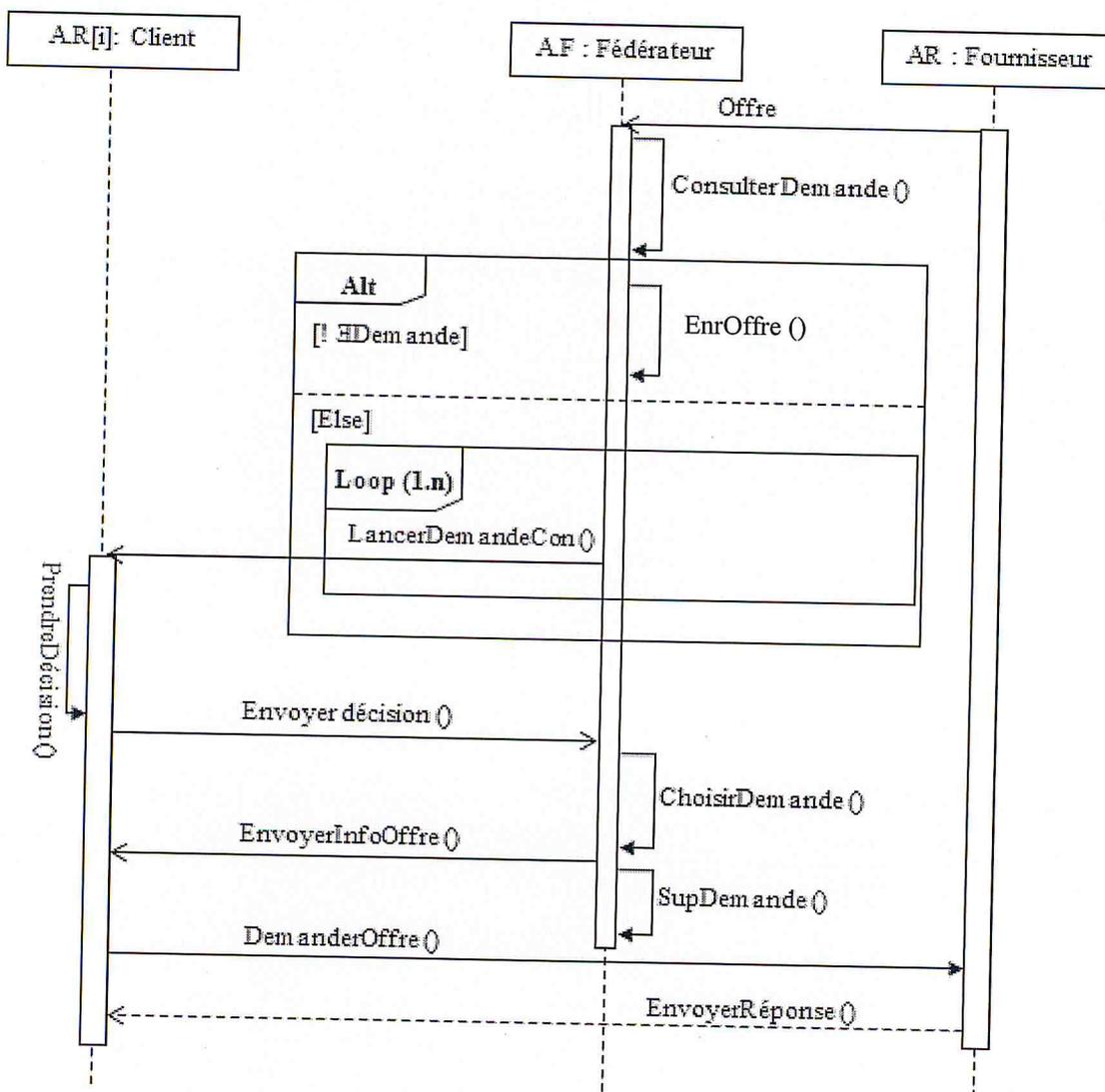


Fig.30 : Diagramme de Séquence « Suite à une offre »

II.5. Diagramme d'activités :

Ce diagramme est utilisé pour la représentation dynamique du système.

II.5.1. Consultation périodique de stock :

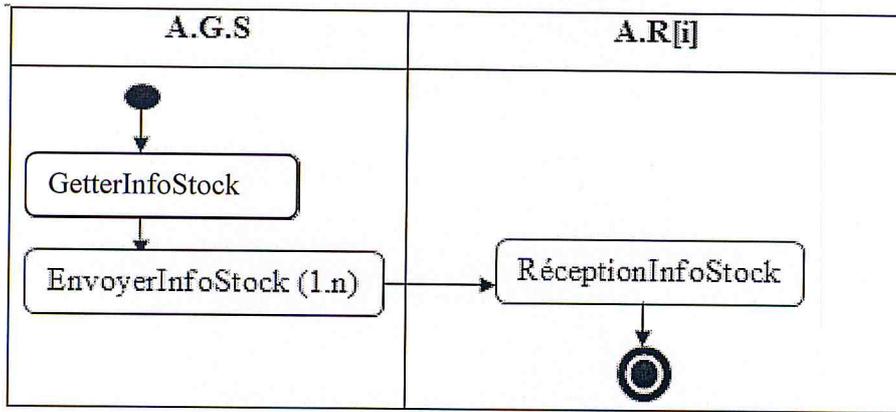


Fig.31 : Diagramme d'activité Consultation périodique

II.5.2. Gestion des demandes et des offres :

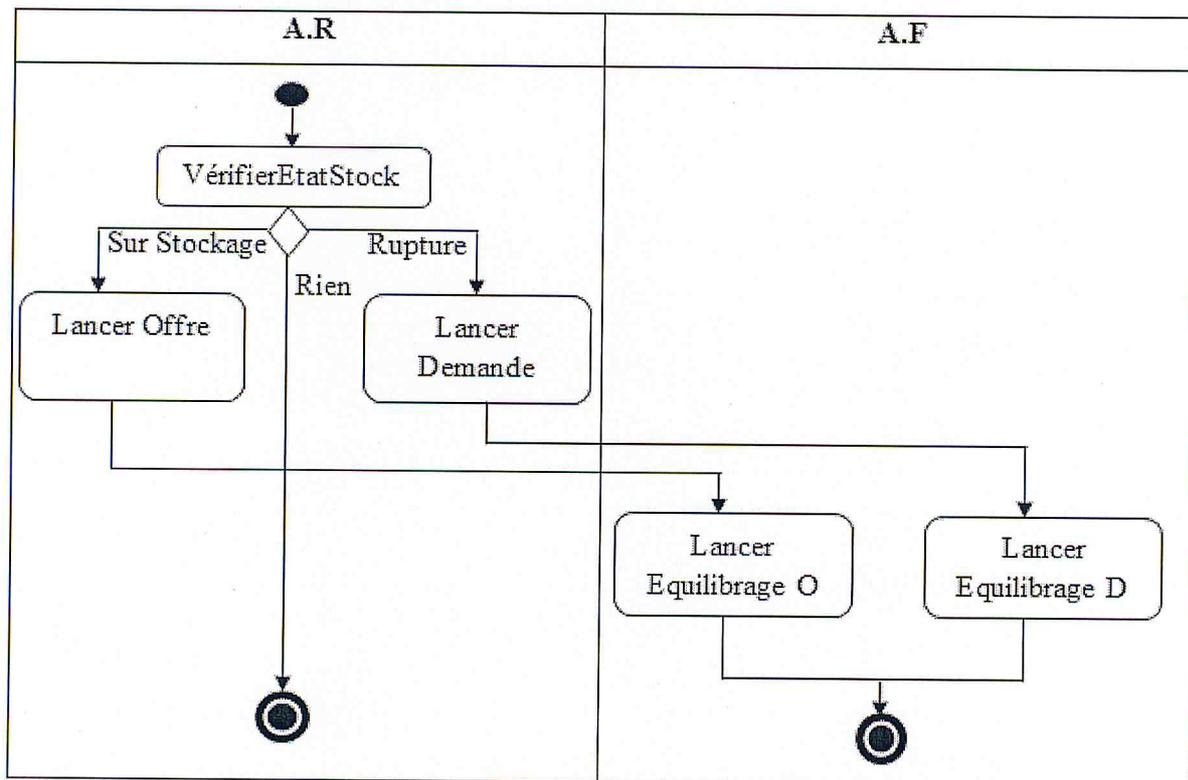


Fig.32 : Diagramme d'activité de gestion des demandes et des offres

II.5.3. Équilibrage des stocks pharmaceutique (Suite à une demande) :

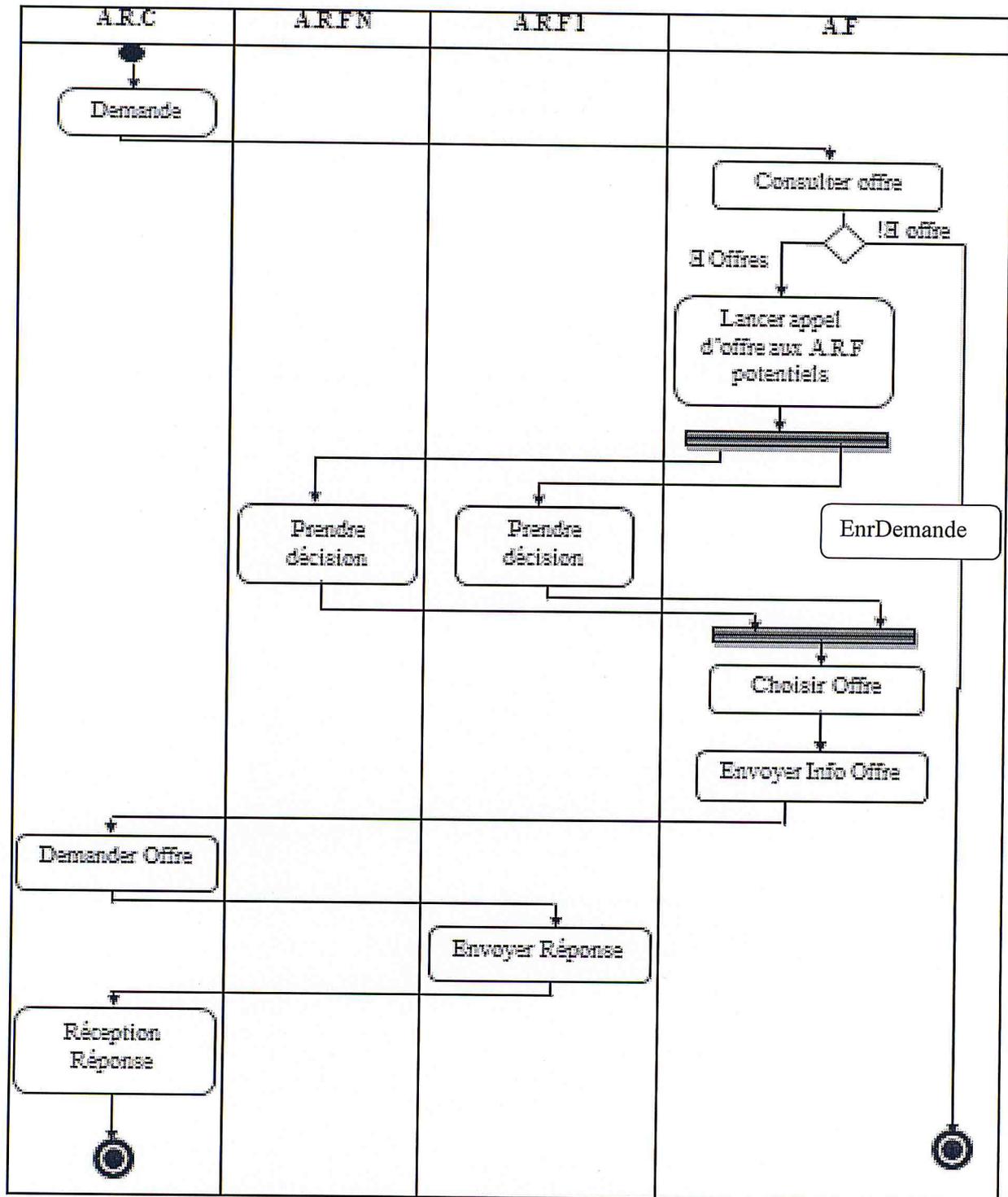


Fig.33 : Diagramme d'activité d'équilibrage de stock (Suite à une demande)

II.5.4. Équilibrage des stocks pharmaceutique (Suite à une offre)

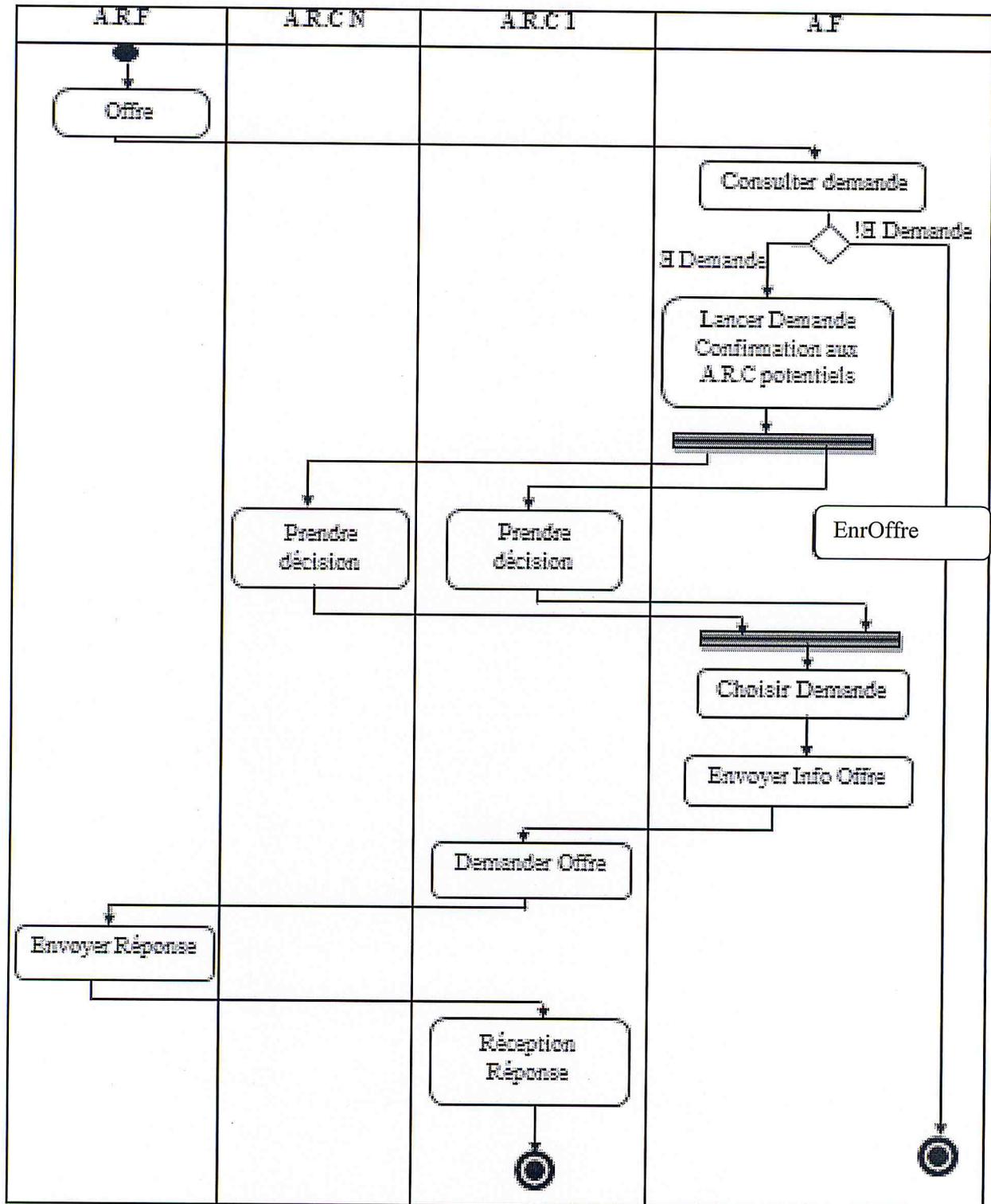


Fig.34 : Diagramme d'activité d'équilibrage de stock (Suite à une offre)

II.6. Modèle mathématique :

A. Pour la détection des cas de rupture ou du sur stockage:

Après la description détaillé de notre SMA, nous allons voir comment réagit notre méthode avec la méthode de gestion de stock utilisée dans les plupart des pharmacies (Recomplètement) ou nous considérons que chaque site pharmaceutique a une application de gestion de stock et cette application calcule :

1. Pour chaque type de médicaments, sa Consommation Moyenne Mensuelle (CMM=consommation pendant n mois/ n).
2. Le Stock de Sécurité (SS=1CMM, généralement) auquel on ne doit toucher qu'en cas d'urgence.
3. La quantité à stocker (QS= SS + La quantité nécessaire pour toute la période à venir avant de recevoir la prochaine commande (par exemple 3 CMM si la commande doit arriver dans 3 mois)).
4. La quantité actuelle de médicament (QA).
5. La date de dernier approvisionnement (DateDA).

Notre travail se base sur ces données pertinentes qui nous aide à détecter les situations anormale de stock à savoir la rupture de stock et le sur stockage.

1. L'A.G.S consulte le stock périodiquement (chaque période P) et guette les données nécessaires (CMM, SS, QS, QA, DateDA).
2. L'A.G.S envoie ces donnés à l'A.R qui les utilise pour détecter les anomalies.

Comment détecter l'état de stock (Rupture/sous stockage) :

1. A la réception des données de la part de l'A.G.S :

Calculer la quantité de médicament prévue (QP) dans cette date :

Pour calculer QP il faut calculer la consommation moyenne prévue par jour (CMPJ)

$$\text{CMPJ} = \text{CMM}/j. \quad (j = \text{Nombre de jour de mois actuel}).$$

$$\boxed{\text{QP} = \text{QS} - (\text{CMPJ} * g).} \quad (g = \text{Nombre de jour entre l'arrivé de dernière approvisionnement et le jour actuel}).$$

Exemple Explicatif : (pour un seul type de médicament)

Dans une pharmacie qui fait l'approvisionnement chaque trimestre :

1. Le 22/05/2011, L'A.G.S vérifie le stock de paracétamol et extraire les informations suivantes :

$$\text{CMM}=310 \text{ unité/mois.}$$

$$\text{SS}=1\text{CMM}=310 \text{ unité.}$$

$$\text{QS}=3\text{CMM}+1\text{CMM} = 4\text{CMM} = 4*310=1240 \text{ unité.}$$

$$\text{QA}=730 \text{ unité.}$$

$$\text{DA}=01/05/2011 \text{ (date de dernière approvisionnement).}$$

2. L'A.G.S envoie ces information à l'A.R_{paracétamol} (CMM, SS, QS, QA, DA).

3. à la réception de ces informations l'A.R_{paracétamol} calcule :

$$\text{CMPJ}=\text{CMM}/j=310/31=10 \text{ unité/jour.}$$

$$\text{QP}=\text{QS}-(\text{CMJP}*g)=1240-(10*21)=1030 \text{ unité}$$

$$(g= (22/05/2011) - (01/05/2011) = 21 \text{ jour})$$

3. Nous supposons que $\text{SD} = 0.1*\text{CMM} = 0.1*310 = 31 \text{ unité.}$

$$4. |\text{QA}-\text{QP}| = |730-1030| = 300 \text{ unité.}$$

Ce déférence est supérieur au seuil fixer ($300>31$) et $\text{QP}>\text{QA}$ donc une situation de rupture est prévue, donc l'A.R_{paracétamol} lance une demande avec une quantité à commander ($\text{QAC}=\text{QP}-\text{QA}=1031-730=300 \text{ unité.}$).

5. à la réception de la demande, le stock sera équilibré.

B. Pour la sélection et le choix de meilleure offre :

Pour sélectionner les fournisseurs potentiels pour une demande arrivée qui a une quantité à demander (QAD), l'A.F se base sur la quantité à offrir (QAO) de chaque offre existante, il sélectionne tous les offres qui ont une $\text{QAO} \geq \text{QAD}$, et il envoie une appelle d'offre pour tous les A.R.F des offres sélectionnées.

À la réception des décisions des fournisseurs potentiels, l'A.F prend en considération que les offres confirmées et il sélectionne l'offre qui a la plus petite QAO (Bien sur $>\text{QAD}$).

C. Pour la sélection et le choix de meilleure demande :

Pour sélectionner les clients potentiels pour une offre arrivée qui a une quantité à offrir (QAO), l'A.F se base sur la quantité à demandée (QAD) de chaque demande existante, il sélectionne tous les demandes qui ont une $QAD \leq QAO$, et il envoie une demande de confirmation de demande pour tous les A.R.C des demandes sélectionnées.

À la réception des décisions des clients potentiels, l'A.F prend en considération que les demandes confirmées et il se base sur la période restée pour tomber en rupture, il sélectionne la demande qui a la plus petite période.

II.7 Les différents Scripts des agents :

A. Script de l'A.G.S :

```
// Je suis l'A.G.S
Pour chaque période P {
  Pour chaque (type de médicament k) {
    GuetterInfoStock (CMM, SS, QS, QA, DateDA)
    EnvoyerInfoStock (A.Rk, CMM, SS, QS, QA, DateDA)
  }
}
```

B. Script de l'A.R :

```
// Je suis l'A.R de médicament (catégorie) k
Faire parallèle {
  IF " Si je reçois (CMM, SS, QS, QA, DateDA) de mon A.G.S " {
    Etat= VérifierEtatStock (CMM, SS, QS, QA, DateDA)
    IF "Etat = rupture"{
      LancerDemande (A.Fk)
    }
    ELSE {
      LancerOffre (A.Fk)
    }
  }
}
IF " Si je reçois un message de demande de confirmation d'une demande de mon A.Fk" {
  PrendreDécision ()
  Envoyerdécision (A.Fk)
}
IF " Si je reçois un message d'appel d'offre de mon A.Fk" {
  PrendreDécision ()
  Envoyerdécision (A.Fk)
}
IF " Si je reçois un message de demande d'offre d'un A.Rl" {
  EnvoyerRéponse (A.Rl)
}
}
```

C. Script de l'A.F :

```

// Je suis l'A.F de médicament (catégorie) k
DO_PARALLELE {
IF " Si je reçois une demande de l'A.R.Ck de site S" {
    ConsulterOffre ()
    IF "Si il n'existe pas d'offre" {
        EnrDemande ()
    }
    ELSE {
        LancerAppelOffre () // pour les fournisseurs potentiels
    }
}
IF " Si je reçois une offre de l'A.R.Fk de site S" {
    ConsulterDemande ()
    IF "Si il n'existe pas de demande" {
        EnrOffre ()
    }
    ELSE {
        LancerDemandeCon () // pour les clients potentiels
    }
}
IF " Si je reçois toutes les décisions concernant une appelle d'offre " {
    ChoisirOffre ()
    EnvoyerInfoOffre(A.R.Ck)
    SupOffre ()
}
IF " Si je reçois toutes les décisions concernant une demande de confirmation de demande " {
    ChoisirDemande ()
    EnvoyerInfoOffre(A.R.Ck)
    SupDemande ()
}

```

II.8. Mise en œuvre de la solution :

II.8.1. Solution web services :

Dans notre cas nous allons utiliser les services Web comme un cadre technologique pour mettre en place notre SMA, pour qu'il être accessibles à travers le Web.

Pour cela nous allons utiliser une conception découplée, à partir de notre SMA, une couche à base de services Web rend la communication entre agents via le Web possible, et le support logiciel « middleware » qui nous allons l'utilisé est le SOAP.

Le transport des messages qui sont encodés en XML se fait par l'intermédiaire de protocoles HTTP, HTTPS ou SMTP dans un environnement décentralisée et distribuée.

Pour notre cas, les trois acteurs logiciels distribués (serveur du service, annuaire des services et le demandeur du service) qui sont sur le web doivent communiquer entre eux en utilisant un protocole de communication de type HTTP ou HTTPS. Cela permet à nos agents de pénétrer n'importe quel pare-feu des réseaux locaux utilisateur d'Internet.

Les figures 35 et 36 montrent la mise en œuvre de notre solution sous forme de Web-Service.

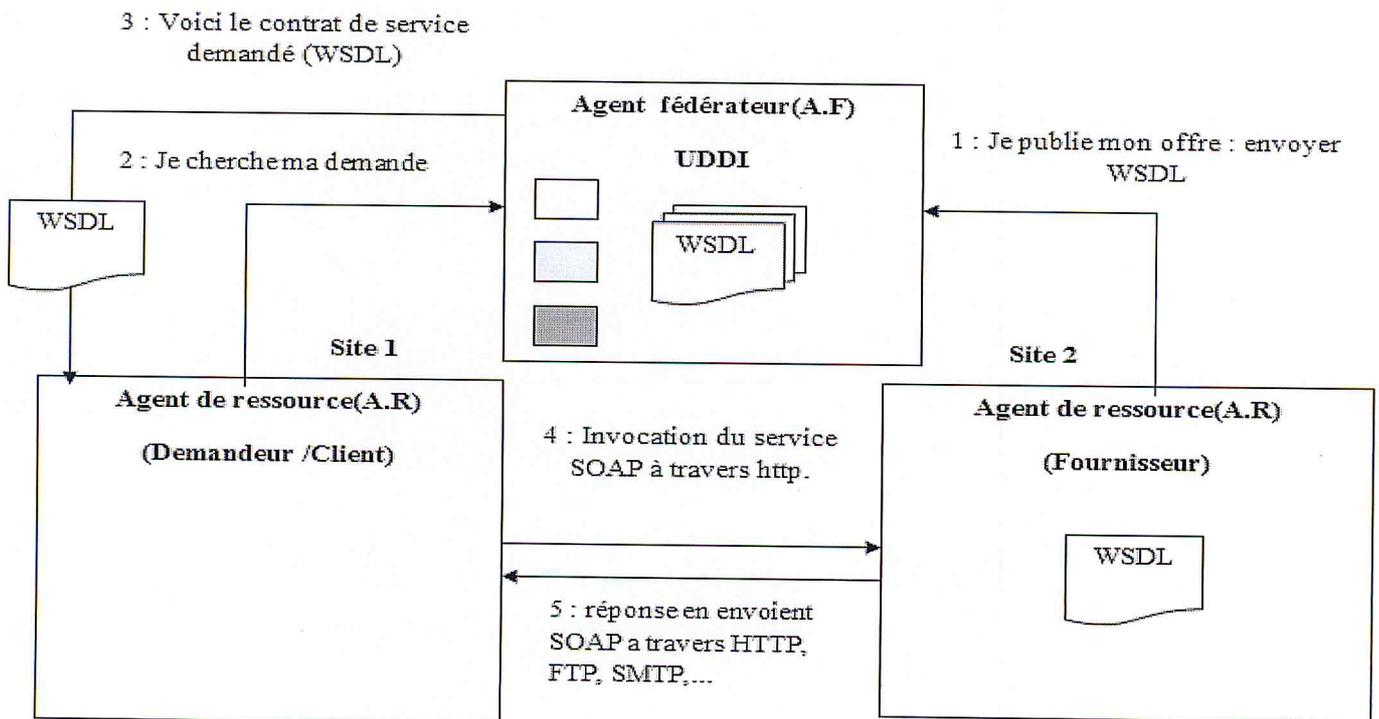


Fig.36 : Le mécanisme d'équilibrage d'une seule ressource par l'intégration des services web

Dans cette approche nous concéderons :

- L'agent fédérateur comme un annuaire (UDDI)
 - L'agent de ressource peut être un client ou un fournisseur
 - Un client dans le cas où cet agent envoie une demande (sous stocke/rupture)
 - Un fournisseur dans le cas où cet agent a une offre
- Des pages regroupent les informations liées aux agents ressources (pharmacies) publient leurs services et leurs demandes, et les moyens de les contacter.
- Des pages regroupent les informations à propos de la classification des offres et les demandes des agents des ressources
- Des pages servent à la description technique des différents services web (offres) offerts par les agents des ressources (pharmacies)
- WSDL C'est la description de l'interface de service (les méthodes, les paramètres d'entrée et de sortie leurs types, et comment y accéder)

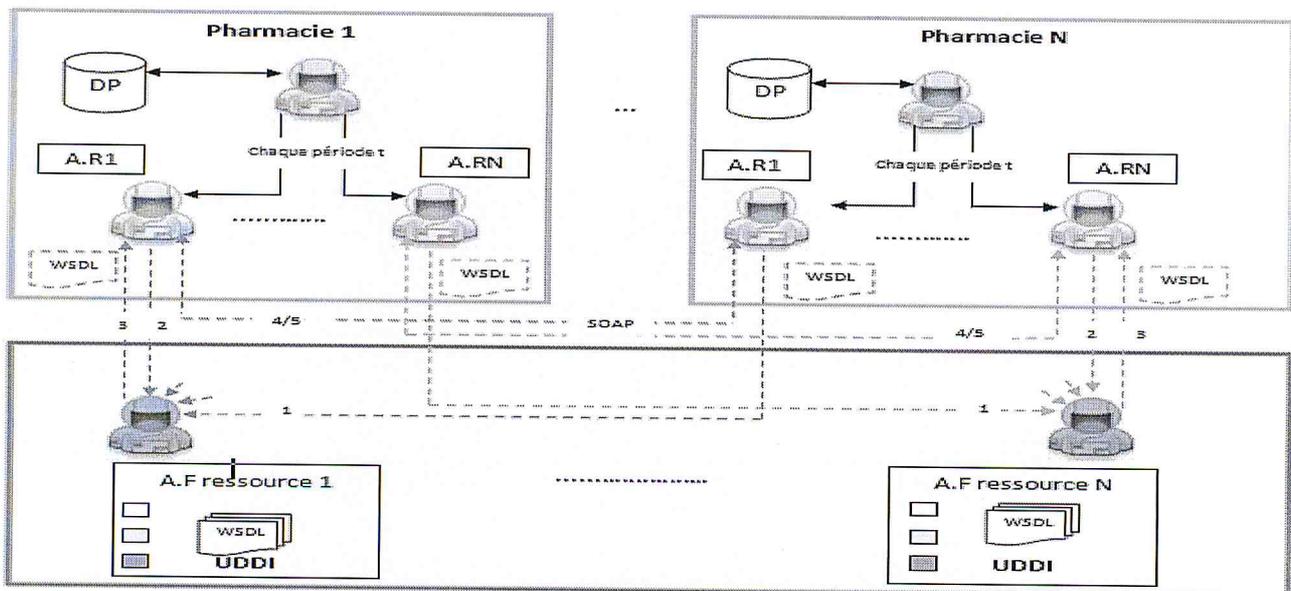


Fig.37 : Le mécanisme d'équilibrage de plusieurs ressources par l'intégration des services web

- Dans cette figure on suppose que l'ARN de la pharmacie 1 et l'AR1 de la pharmacie N sont des agents (fournisseurs) de la ressource N et 1 respectivement.
- L'AR1 de la pharmacie 1 et l'ARN de la pharmacie N sont des agents (demandeurs) de la ressource 1 et N respectivement

Le scénario d'équilibrage du stock pharmaceutique en utilisant les services web:

1. Lorsqu'un agent de ressource (fournisseur) détecte un sur stockage il publie l'offre en envoyant un message WDSL à l'agent fédérateur qui contient toutes les informations pour l'invocation de cette offre.
2. Lorsqu'un agent de ressource (demandeur) détecte un sous stockage il exprime ses besoins en envoyant une demande à l'agent fédérateur.
3. Lorsque l'agent fédérateur trouve une offre qui satisfait cette demande il envoie un message WSDL qu'est le contrat de l'offre à l'agent de ressource (demandeur).
4. L'agent de ressource (demandeur) demande l'offre de l'agent de ressource (fournisseur).
5. L'agent de ressource (fournisseur) répond à l'agent de ressource (demandeur).

II.8.2. Diagrammes de Séquences :

Dans ce qui suit la présentation des diagrammes des séquences d'élaboration des demandes et équilibrage de stock avec l'intégration des services web :

Les messages échangés :

Terme	Signification
WSDLD	La description de l'offre demandée
WSDLO	La description de l'offre a publiée
SOAPA-O	SOAP d'appel d'offre
SOAPDecision	SOAP de décision
SOAPD-O	SOAP de demande de l'offre
SOAPR	SOAP réponse à la demande de l'offre
SOAPD-C	SOAP de demande de confirmation des demandes

Tab.10 : Les messages échangés

A. Elaboration des demandes :

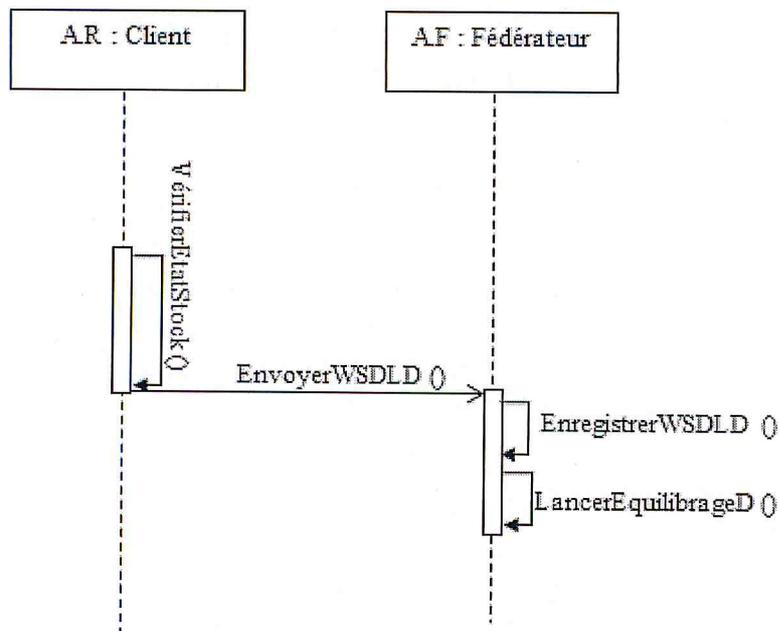


Fig.38 : Diagramme de séquence «Elaboration de demande »

B. Publication d'offre :

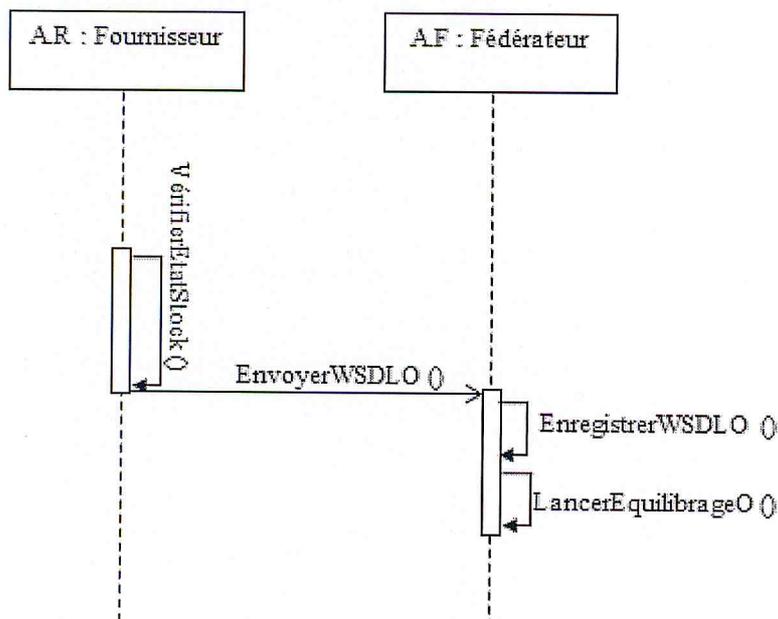


Fig.39 : Diagramme de séquence «Publication d'offre»

C. Recherche et invocation d'une offre :

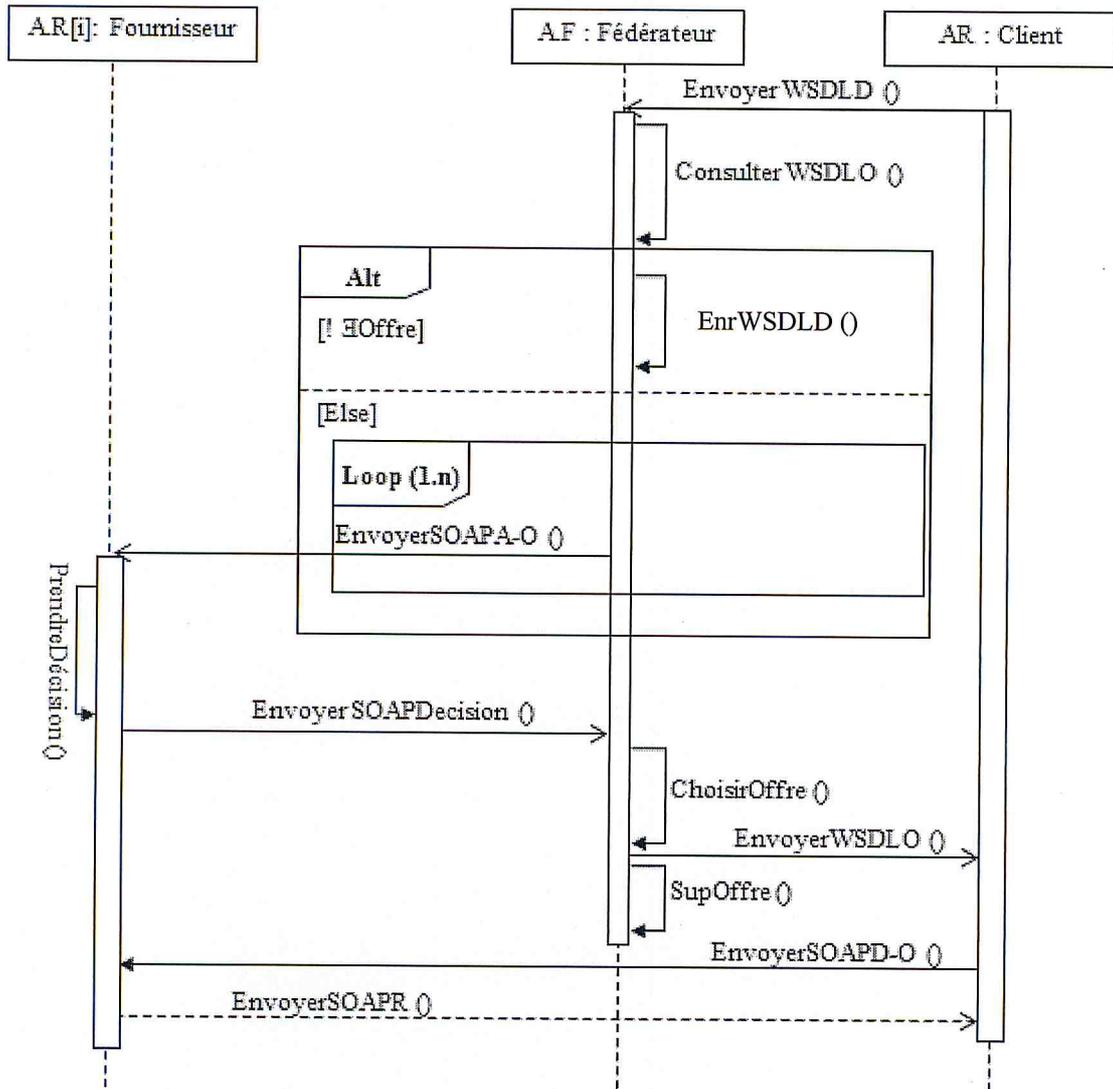


Fig.40 : Diagramme de séquence «Recherche et invocation d'une offre»

D. Recherche et sélection d'une demande :

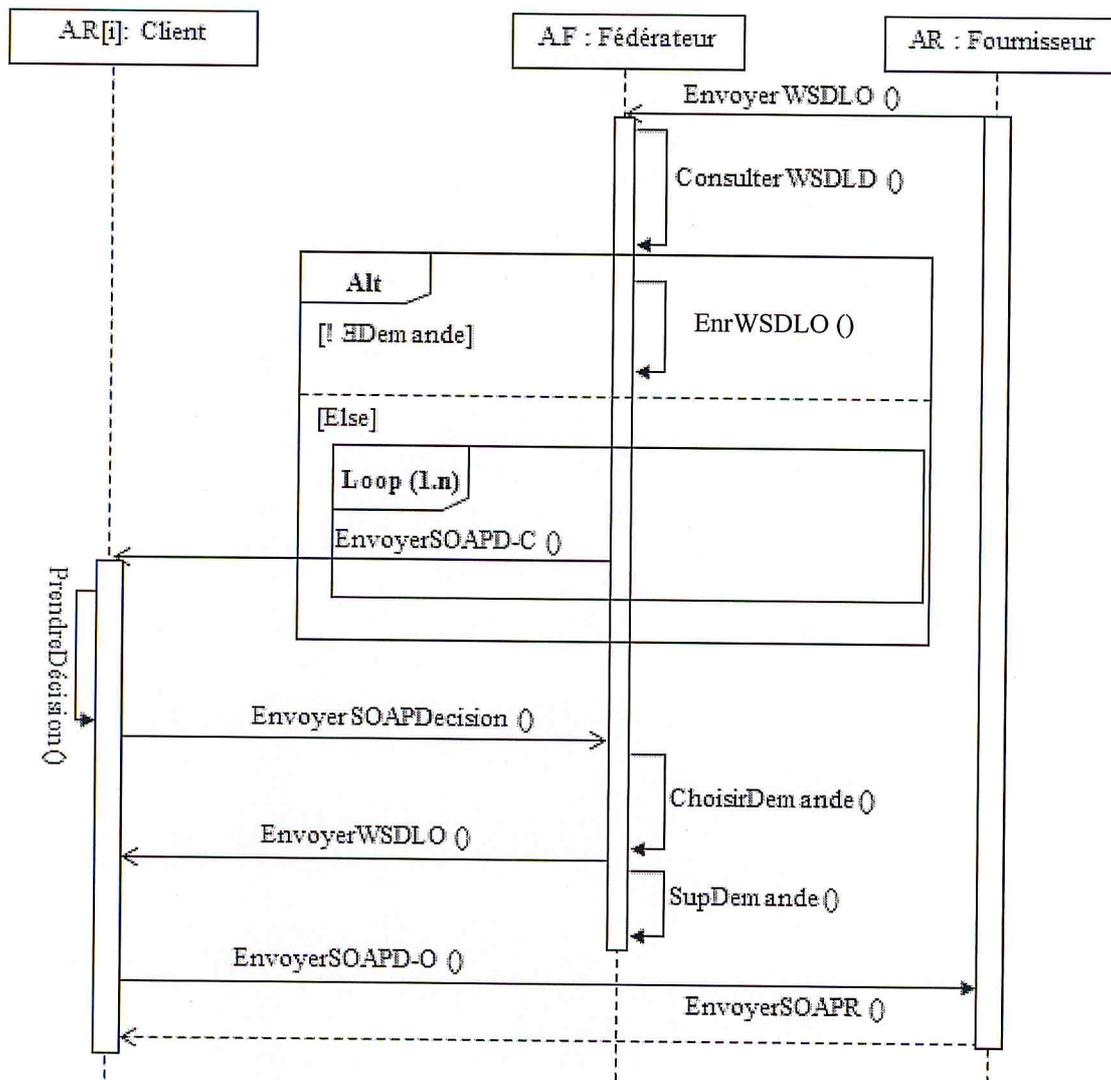


Fig.41 : Diagramme de séquence «Recherche et sélection d'une demande»

CHAPITRE II :

La Réalisation du Système

II.1.outils de développements :

Pour la réalisation de notre produit logiciel nous avons opté pour la Plate-forme « Jade 3.6 » pour l'implémentation de notre système Multi Agents, « Eclipse j2ee » pour implémenter les services web et les intégrer avec notre système Multi Agents, pour les serveurs nous avons utilisés « Apache Tomcat 6.0 » comme serveur d'application, et « MySQL » comme serveur de base de données.

La plate-forme JADE:

JADE est une plate-forme multi-agents créée par le laboratoire TILAB. JADE permet le développement des systèmes multi-agents et d'applications distribuées conformes aux normes FIPA. Aucune méthodologie n'est spécifiée. Elle est entièrement implémentée en JAVA et fournit des classes qui implémentent « JESS » pour la définition du comportement des agents.

JADE possède trois modules principaux (nécessaire aux normes FIPA) :

- DF : « DirectorFacilitator » fournit un service de « pages jaunes » à la plate-forme.
- ACC : « Agent Communication Channel » gère la communication entre les agents.
- AMS : « Agent Management System » supervise l'enregistrement des agents, leur authentification, leur accès et l'utilisation du système.

Le langage de Communication de la plate-forme JADE est FIPA-ACL (Agent Communication Language). [Jade].

Eclipse JEE

Java Platform, Enterprise Edition ou Java EE est une plate-forme largement utilisée pour la programmation serveur dans le langage de programmation Java. La plate-forme Java (Enterprise Edition) est différente de la plate-forme Java Standard Edition (Java SE) en ce qu'il ajoute les bibliothèques qui fournissent des fonctionnalités pour déployer à tolérance de pannes, distribués, à plusieurs niveaux du logiciel Java, largement basée sur des composants modulaires fonctionnant sur un serveur d'application [Wikipédia 2].

Apache Tomcat:

Est un conteneur libre de servlets Java. Issu du projet Jakarta, Tomcat est un projet principal de la fondation Apache. Tomcat implémente les spécifications des servlets et des JSP du Java Community Process . Il est paramétrable par des fichiers XML et de propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP.

Tomcat est un serveur Web qui gère les servlets et les JSP. C'est le compilateur Jasper qui compile les pages JSP pour en faire des servlets.

Tomcat a été écrit en langage Java. Il peut donc s'exécuter via la machine virtuelle Java sur n'importe quel système d'exploitation la supportant. [Tomcat]

MySQL :

Selon [Wikipédia 1] MySQL est un système de gestion de base de données (SGBD). Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle et Microsoft SQL Server.

Les bases de données sont accessibles en utilisant les langages de programmation C, C++, VB, VB .NET, C#, Delphi/Kylix, Eiffel, Java, Perl, PHP, Python, Ruby et Tcl ; une API spécifique est disponible pour chacun d'entre eux. Une interface ODBC appelée MyODBC est aussi disponible.

II.2. Architecture générale de logiciel :

Notre logiciel est composé de deux modules :

- Un module local au niveau de chaque pharmacie de la fédération.
- Un module global destiné à l'autorité qui gère cette fédération.

Le Module local :

Nous mettons en place ce module de notre logiciel au niveau de chaque site pharmaceutique, ce module à une relation avec l'application de gestion de stock de la pharmacie, au lancement de ce module :

- Un agent gestionnaire de stock est créé, qui a les objectifs suivants :
 - ✓ Surveillance de quotidien de la pharmacie en terme d'entres et sorties des médicaments.
 - ✓ Guette les informations sur les médicaments et les envoyer aux agents de ressources.
- La consultation de base de données de gestion de stock, guette tous les catégories de médicament et créer pour chaque catégorie un agent de ressource
 - ✓ Calculer l'état de stock de chaque ressource.
 - ✓ Déclenche l'envoi d'une offre ou une demande selon l'état détecté
 - ✓ Répondre aux messages de confirmation d'une demande ou une offre.
 - ✓ Réponde aux messages de demande d'offre.

Le module global :

Ce module contient :

- Un agent fédérateur pour chaque catégorie de médicaments existante dans la fédération.
 - ✓ Accueillir les messages des agents de ressources « messages des demandes, offres, décisions, mis à jour ».
 - ✓ Chercher les meilleures offres pour les demandes existantes.
 - ✓ Chercher les meilleures demandes pour les offres existantes.
 - ✓ Envoyer des messages de demande de confirmation aux agents de ressources. « messages de demandes de confirmation des offres ou demandes ».
 - ✓ Envoyer des messages de description d'offre à un agent de ressource client.

Ces deux modules coopèrent pour atteindre l'objectif finale « équilibrage de stock » cette coopération est assurée par les outils réseaux .La figure suivante montre clairement cette architecture.

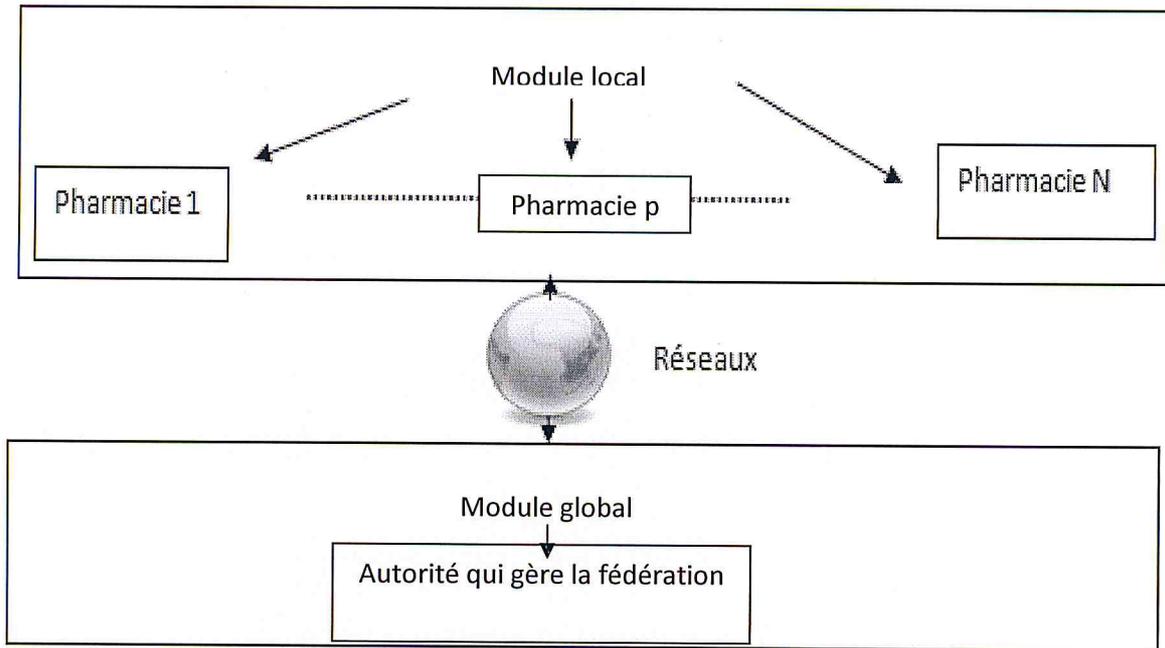


Fig.42 : Architecture générale de logiciel

II.3. Architecture fonctionnelle et comportement de logiciel:

- **Module locale (Partie pharmacie) :**

Le menu principal de ce module contient trois boutons lancer, option et fermer.

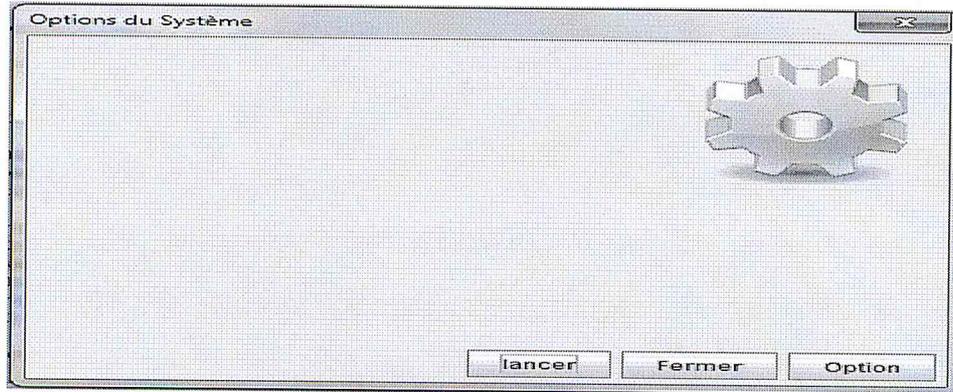


Fig.43 : menu principale de module locale

Le Bouton lancer :

Permet le lancement de système local :

- Lancer l'agent gestionnaire de stock.
- Lancer les agents de ressources.

Le Bouton option :

Permet de paramétrer le système, les options disponibles sont illustrées sur le schéma suivant.

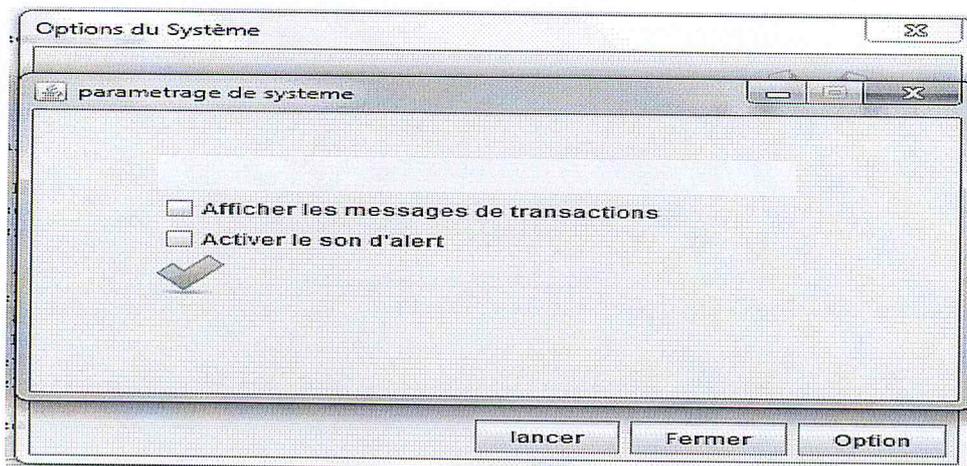


Fig.44: Fenêtre paramétrage de système

Un message d'alerte s'affiche lorsqu'un agent de ressource envoie une demande ou une offre quand il détecte un état d'anomalie « sur stockage, sous stockage ».



Fig.45 : Message d'alerte détection d'anomalie

Le Bouton consulter :

Permet de voir le détail de ces messages, ce détail concerne le type de message, offre ou demande, la quantité et la date d'envoi.

Lors de la réception des messages provenant de l'extérieur (demande d'offre) le message suivant s'affiche

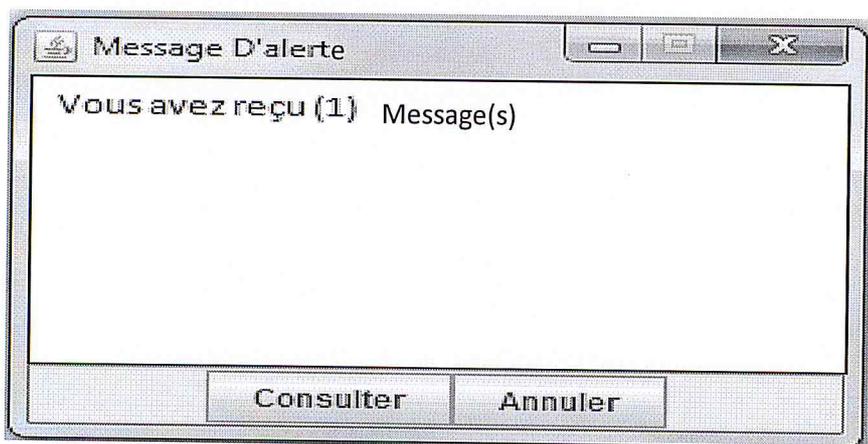


Fig.46 : Message d'alerte recevoir un message

Le Bouton consulter :

Permet de visualiser les détails des messages reçues (La ressource demandée, la quantité et le site demandeur).

- **Module globale (Partie fédération) :**

Le menu principal de ce module offre les fonctionnalités suivantes :

Gestion des agents fédérateurs :

Permet la création, modification, suppression des agents fédérateurs.

Actualité de fédération :

Elle contient les sous menus suivants :

Afficher offre : il affiche tous les offres existantes dans la fédération.

Afficher demande : il affiche tous les demandes existantes dans la fédération.

Offres /Demandes : il affiche la quantité de médicament en offre et en demande des la fédération.

Affectation offres/demandes : il affiche les choix d'équilibrage fait par les agents fédérateurs.

Transaction entre Agents : il affiche les interactions entre les agents de ressources et l'agent fédérateur qui gèrent la même ressource (médicament).

Lancer Système :

Permet de lancer le système d'équilibrage :

- Lancement des agents fédérateurs.

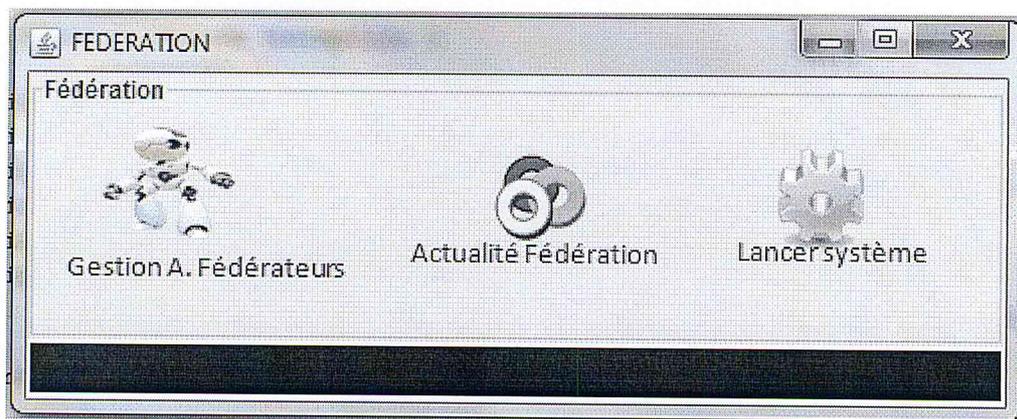


Fig.47 : Menu principal du module global



Fig.48 : Fenêtre gestion des agents fédérateurs

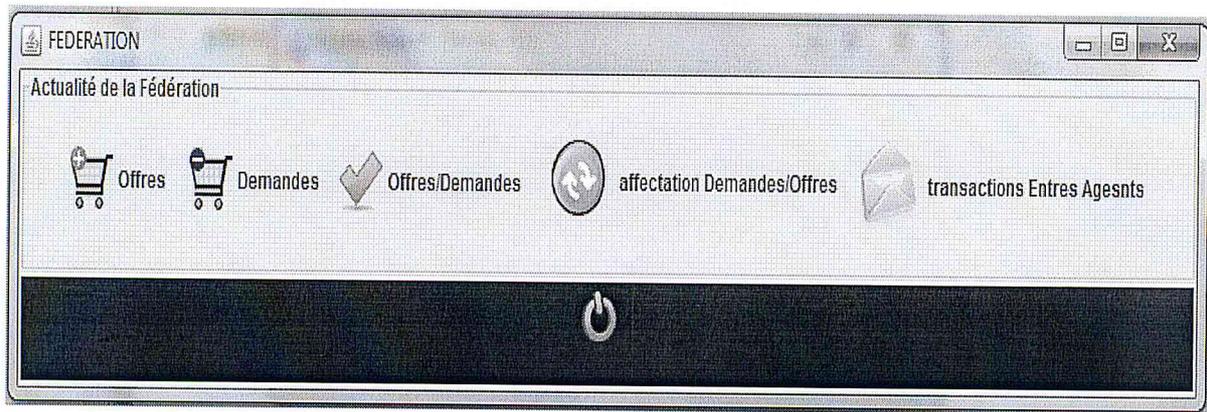


Fig.49 : Fenêtre d'actualité de la fédération



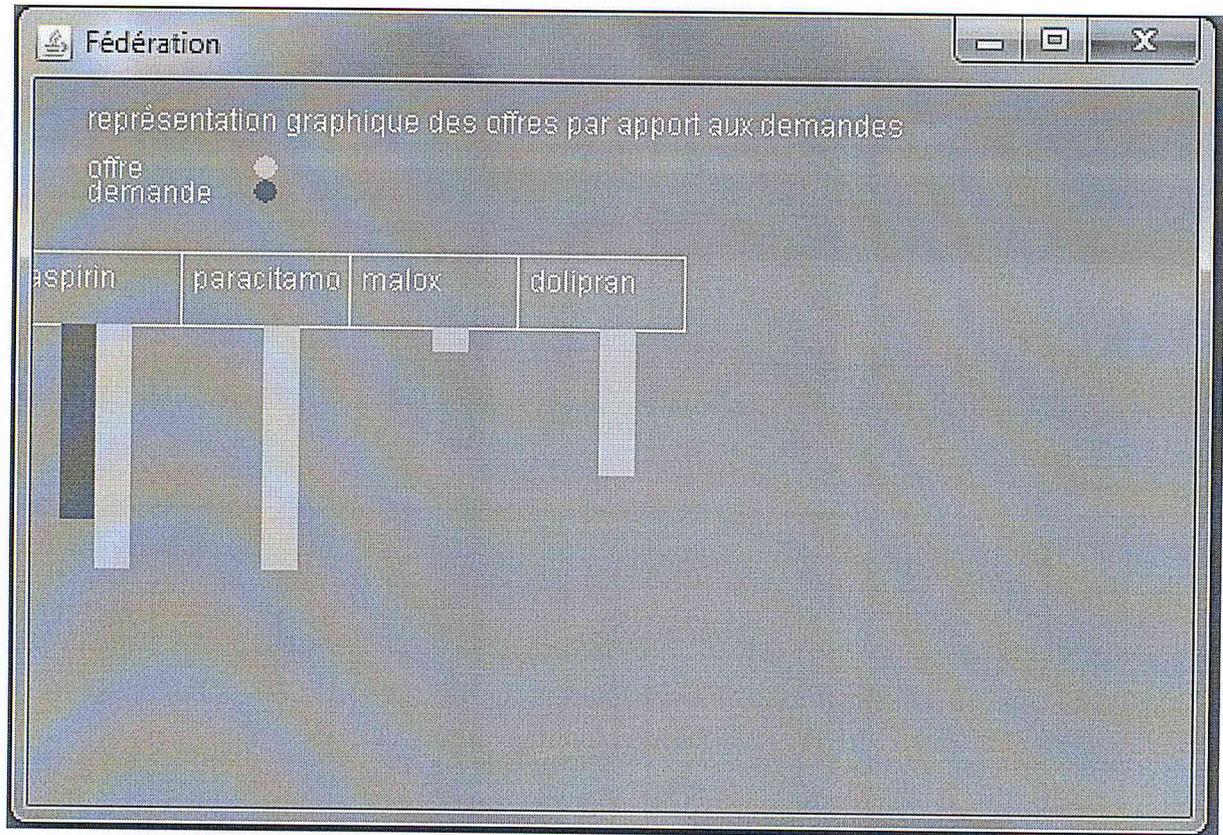


Fig.58 : Présentation graphique des offres par apport aux demandes

Les interactions entre les agents :

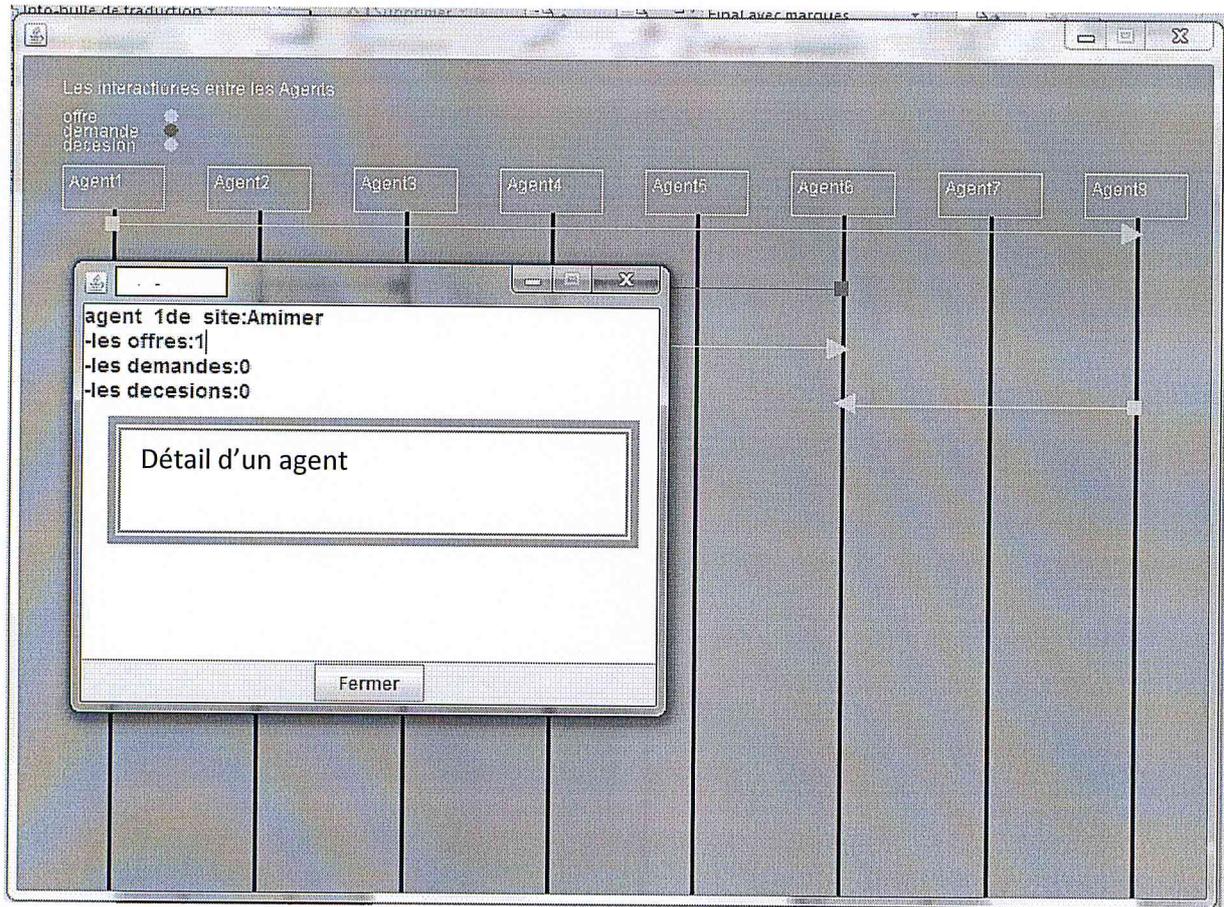


Fig.60 : Les interactions entre agents

Affectation demandes/offres : c'est une liste qui affiche les affectations fait par les agents fédérateurs

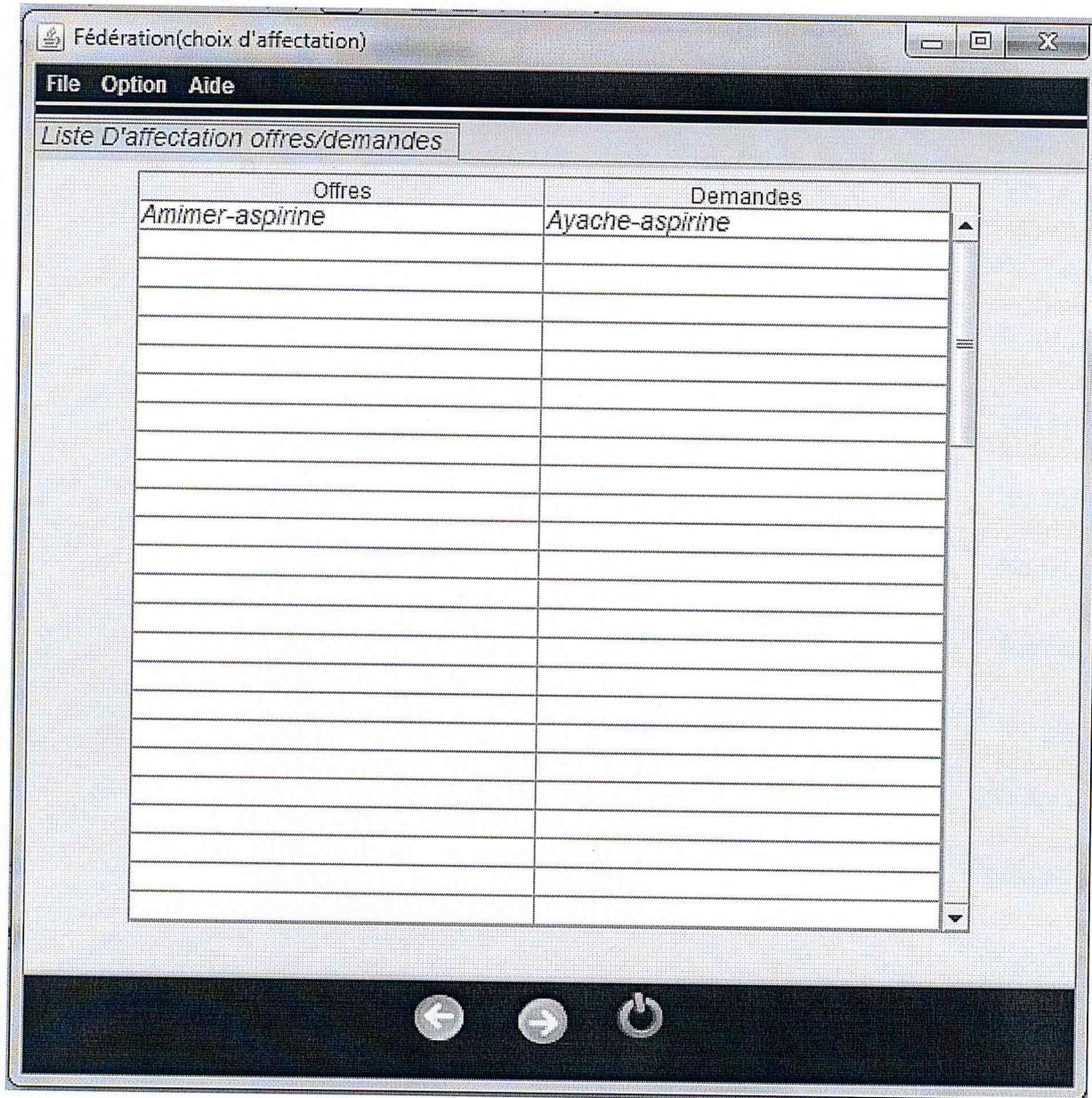


Fig.59 : liste d'affectation offres/demandes

BIBLIOGRAPHIE

Bibliographie:

*[Bar, 10]: Mikael Baron, «SOA – Web Services Introduction SOA», 2010, PDF.

* [Ben, 09] : Riadh Ben Halima, Thèse Conception pour obtention du doctorat, « implantation et expérimentation d'une architecture en bus pour l'auto réparation des applications distribuées à base de services web », Toulouse III- Paul Sabatier et la Faculté des Sciences Économiques et de Gestion – Sfax, 2009 , PDF.

*[Car, 03] : Caroline Thierry « Mémoire d'habilitation à diriger des recherches Gestion de chaînes logistiques Modèles et mise en œuvre pour l'aide à la décision à moyen terme », 2003,PDF.

*[Cha et al, 01] : B. Chaib-Draa, B. Moulin, I. Jarras. « Agent et Systèmes Multi agents», In Principes et architecture des systèmes multi-agents, JP. Briot et Y Demazeau éditions Hermes, Lavoisier, 2001.

*[Che, 09] : Cherouati walid, Exposé « Langages de communication & d'échange d'information sémantisée entre agents », 2009.

*[Deg, 04] : Patrice Degoulet « Systèmes d'information hospitaliers Dossier patient et prescription électronique » ,PDF.

*[Des, 08] : Desmazes Jean, Thèse pour obtenir le grade de : Docteur de l'université François – Rabelais, « étude et comparaison des factures décisionnels de l'externalisation informatique dans les établissements hospitaliers publics et privés », 2008.PDF.

*[Fer, 95] : J. Ferber, « Les systèmes multi agents vers une intelligence collective », Inter Edition, Paris, France, 1995.

*[Gue et Gue, 10] : Guenineche Zakaria et Guezil Mohamed, Mémoire de fin d'étude, « Conception et réalisation d'un système d'information de gestion de stock et d'approvisionnement interfacé avec le système commercial et le système de production de l'ONAAPH », L'institut National d'Informatique,2010 , PDF.

Conclusion générale :

Le thème traité par ce mémoire est nouveau en Algérie. L'idée de l'introduire vient des problèmes liés au secteur sanitaire qui sont de deux types :

Des problèmes qui menacent la vie patients « disponibilité des médicaments, le sous stockage des médicaments qui provoque des ruptures de stock »

Des problèmes économiques des couts supplémentaires « stockage des médicaments périmes » et des pertes.

A travers cette étude nous avons pu atteindre les objectifs tracés grâce à une organisation logique de travail. Dans un premier temps et après la présentation de sujet, la problématique et les objectifs voulus, nous avons étudié les différents aspects liées à ce thème à savoir :

Les systèmes d'information de santé, les systèmes d'information hospitalier, la gestion du stock pharmaceutique, la problématique de la planification des ressources qui constituent l'espace métier et les architectures oriente services, services web et les systèmes multi-agents qui constituent l'espace solution. Nous avons conclu cette partie en mettant en évidence notre approche « couplage système multi-agents/ services web ».

Dans la deuxième partie nous avons présenté notre solution de gestion coopérative ; sa conception et sa modélisation avec le langage AUML

Bilan du travail :

Comme n'importe quel projet, notre travail présente des points positifs, des limites et surtout des perspectives.

Notre travail englobe plusieurs points positifs liés aux aspects conception, modélisation et applicatifs parmi lesquels :

❖ Conception, modélisation :

Grâce à l'architecture optée qui se base sur une organisation modulaire d'agents notre travail peut être réutilisé ou ajusté pour des problèmes liés à la répartition et à la planification de ressources de manière générale. Bien que le modèle choisi est simple mais il donne des bons résultats qui représente fidèlement l'actualité et l'état de stock pharmaceutique.

❖ Applicatifs :

Le produit logiciel développé est simple à utiliser il donne une vision exacte sur l'état de la fédération en temps réel assurant l'équilibrage de stock pharmaceutique distribué.

Sur le plan technique, notre logiciel est modulaire et ouvert envers l'ajout de nouveaux modules facilitant son évolution à des coûts de développement minimes.

Une autre caractéristique importante de notre logiciel c'est la portabilité grâce au langage choisi « java » et l'extensibilité grâce au langage « XML ».

La seule limite que nous pouvons citer concerne le prototype :

- le prototype que nous avons développé fonctionne parfaitement dans les situations d'approvisionnement et de consommations normales (absence d'urgence ou d'épidémie par exemple) mais dès qu'une catastrophe survient il y a des plans d'urgences qui sont prévus pour ce genre des problèmes dans le domaine de la santé.

❖ Perspectives :

Parmi les perspectives que nous avons dégagé suite à notre travail dans ce domaine et qui peuvent être une suite logique de notre thème nous citons :

- Rendre notre système comme un module ou bien un composant paramétrable que nous pouvons appliquer avec des problèmes pareils « équilibrage des ressources », en d'autre terme rendre le système comme un service que nous pouvons l'invoquer en précisent les besoins spécifique. Dans ce cas un langage formel de spécification des ressources devient obligatoire.
- Améliorer le modèle mathématique adopté en utilisant des modèles qui se basent sur la notion des phénomènes stochastiques « aléatoires ».
- Améliorer l'aspect sécurité dans les échanges des messages entre les différents agents par l'utilisation de la signature numérique et le protocole SSL « Secure Socket Layer » par exemple.
- D'un autre côté la prise en compte de la panne d'un agent fédérateur n'est pas encore prévue dans le prototype actuel; ce qui constituera une perspective supplémentaire au travail réalisé.

BIBLIOGRAPHIE

*[Izz, 06] : Saïd IZZA, Thèse pour obtenir le grade de Docteur en Informatique, « intégration des systèmes d'information industriels Une approche flexible basée sur les services sémantiques », l'Ecole Nationale Supérieure des Mines de Saint-Etienne France, 2006 ,PDF.

*[Jl, 04] : JL Michel, « La Gestion des Stocks », Chapitre 3, mars 2004 , PDF.

*[Lar, 08] : Petit Larousse, 2008.

*[Mar, 03]: Marjorie LE BARS, Thèse Pour l'obtention du titre de Docteur en Sciences Spécialité informatique, « Un Simulateur Multi-Agent pour l'Aide à la Décision d'un Collectif : Application à la Gestion d'une Ressource Limitée Agro-environnementale », p49, 2003.

*[Mon, 04] : Mongi Miled– Expert TIC – DG CIMSP, « le système d'information hospitalier des applications de gestion vers le dossier médical informatisé », 2004 , PDF.

*[Mos, 04] : Anne-Marie Mossière, 5eme journées de formation continue GSASA, « Gestion de stock de la pharmacie», 3 et 22 juin2004 , PDF.

*[Ngu, 03] : NGUYEN Thi Truc Vien, Rapport de stage « Modélisation et implémentation d'un système distribué de pages blanche au sein d'une plate forme multi agent», p12, 2003.

*[Otm et Smi, 10] : Otmane Laid et Smiah Lotfi Mémoire fin d'étude : Conception et réalisation d'un système d'information de Gestion du dossier médical pour la médecine du travail. L'institut nationale d'informatique(INI) promotion 2010 , PDF.

*[Sal, 07] : Madjid SALMI, « L'intégration des nouvelles technologies de l'information et de la communication (NTIC) dans le système d'information sanitaire (SIS) en Algérie »,2007.

*[Woo et Jen, 95]: M. Wooldridge et N. R. Jennings. « Intelligent agents: Theory and practice». The Knowledge Engineering Review, vol. 10(2) pp.115–152, 1995.

*[Seg et al., 08] : Amal El Fallah Seghrouchni, Serge Haddad, Tarak Melitti, «Interopérabilité des systèmes multi-agents à l'aide des services Web», l'université Paris 6, 2008,PDF.

*[she,93] les bases de données reparties,PDF.

WEB GRAPHIE

Web Graphie:

[Jade]: <http://jade.tilab.com/>

[Tomcat]: <http://tomcat.apache.org/>

[Wikipedia 1]: <http://fr.wikipedia.org/wiki/MYSQL>

[Wikipedia 2]: http://en.wikipedia.org/wiki/Java_Platform,_Enterprise_Edition/

Agent Unified Modeling Language

AUML

Introduction :

Avant d'aller à la phase d'implémentation « la réalisation effective de la solution » il y a une phase clé pour la réussite d'un projet informatique car dans cette phase ou nous présentons le comportement et les exigences de notre futur système .donc une bonne modélisation mène vers un produit logiciel fiable, robuste, facilement maintenable, évolutif. Nous avons opté le langage AUML pour la modélisation

Contrairement aux objets, les agents sont autonomes et réactifs. On dit à ce sujet qu'un objet fait ce qu'on lui dit de faire alors qu'un agent fait ce qu'il veut bien faire. On peut considérer les agents comme une extension des objets actifs. UML étant le langage de modélisation le plus utilisé dans le monde du déploiement de logiciel orienté-objets, il est naturel de se tourner vers une extension d'UML pour modéliser les agents

AUML (Agent Unified Modeling Language) est un langage de modélisation orienté agent proposé par Bauer, Odell et al en 1999, il a pour but d'étendre UML avec des moyens adaptés aux systèmes multi-agent.

AUML propose des extensions de notation d'UML adaptées au monde des agents, notamment :

- Diagramme de classes d'agents
- Diagramme de séquences.

Diagramme de class:

Le diagramme de classes agent doit supporter tous les concepts liés à l'agent, ce diagramme est illustré dans la figure suivante :

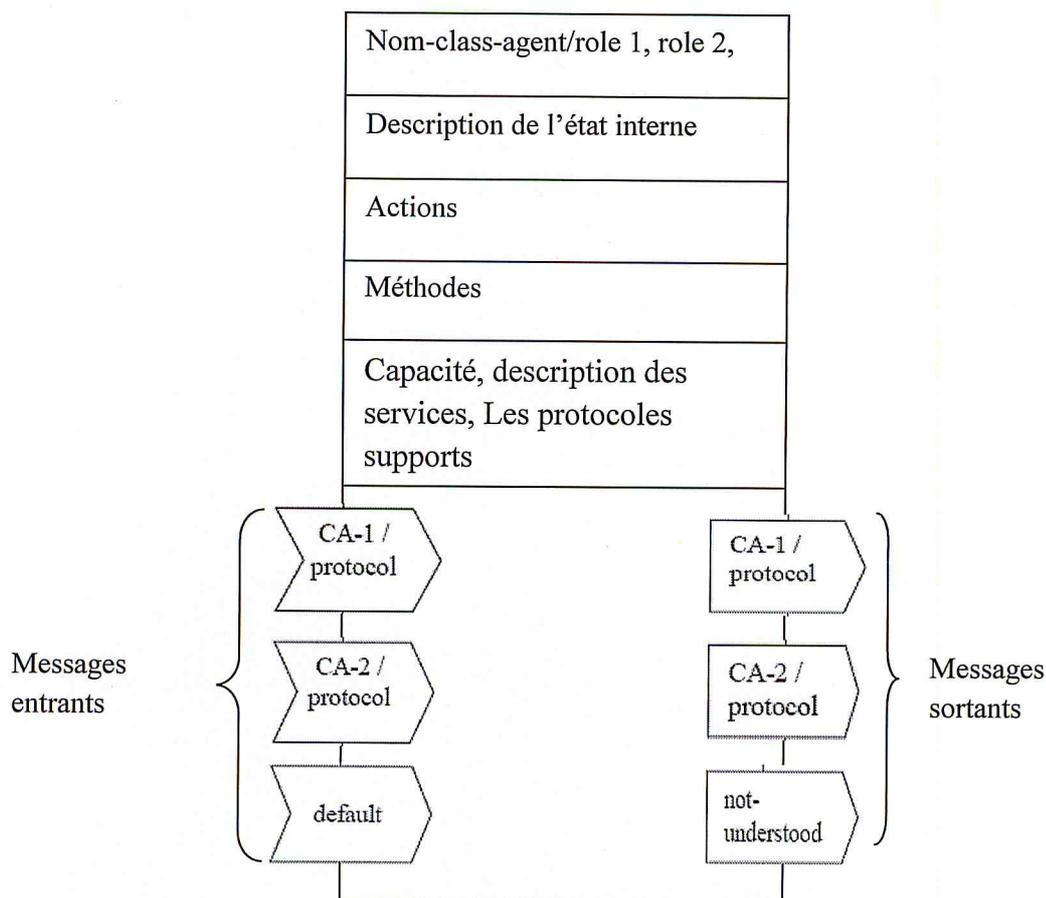


Fig.20: Diagramme de classe d'agent

- Nom de la classe agent/ rôle1, rôle2,...: Un agent d'une classe donnée peut avoir plusieurs rôles.
- Description des états: Définition de variables d'instance qui reflètent l'état de l'agent,
- Actions(Plans): deux types d'actions peuvent être spécifiés: action proactive exécutée par l'agent lui même si une précondition devient vraie, et réactive résultant d'un message reçu d'un autre agent.
- Méthodes: Elles sont définies comme dans UML, avec éventuellement des pré-conditions, post-conditions ou invariants.

ANNEX

- Envoi et réception de messages: l'interface entre les agents et leur environnement est la réception et l'envoi de messages.

Diagramme de séquence AUML :

Le diagramme de séquence AUML est très semblable au diagramme de séquence UML. Seul des opérations ont été ajoutées (types de branchements) afin de prendre en compte l'indéterminisme du comportement d'un agent, Ils permettent aussi d'exprimer clairement les processus de communication entre agents qui impliquent des structures conditionnelles assez lourdes à représenter avec UML.

On distingue trois types de branchement à savoir :

Branchement ET (a), Branchement, OU (b) et Branchement XOR/OU exclusif (c)

Les rôles du diagramme de séquences sont :

- De présenter les interactions entre agents.
- De présenter tous les agents/rôles impliqués.
- De présenter les messages échangés suivant un ordre chronologique.