

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET LA
RECHERCHE SCIENTIFIQUE

UNIVERSITE LARBI BEN M'HIDI-OUM EL BOUAGHI
FACULTE DES SCIENCES EXACTES ET DES SCIENCES DE LA NATURE ET DE
LA VIE

Département de Mathématique et Informatique

N°D'orde :

Série :

Mémoire

Présenté en vue de l'obtention du diplôme de

Magister en informatique

OPTION : Intelligence Artificielle et Imagerie

Thème

**Systeme Distribue D'aide à la
Décision (DDSS)**

Présenté par: Ahlem MEZIANI

Dirigé par: Pr. Sahnoun ZAIDI

Soutenance prévue / / devant le jury composé de :

Année universitaire 2013/2014

Remerciements

Avant tout, merci mon DIEU de m'avoir donné le courage, la volonté et la patience de mener ce travail à terme.

Les années d'une thèse constituent une période d'événements, parfois de doutes, de rencontres, d'échanges. Le travail qui est présenté dans ce mémoire est la synthèse de nombreuses idées qui ont des origines diverses et variées mais aussi la conséquence d'échanges et de relations plus personnelles. Ce travail n'aurait ainsi pu arriver à terme sans l'aide de nombreuses personnes.

Je ne serais pas arrivée jusque-là si Professeur Zaidi Sahnoun, Professeur à l'université de Constantine et l'encadreur de ma thèse, m'avait pas pris sous son aile bienveillant. Je le remercie pour toute l'attention qu'il m'a portée et son soutien inconditionnel au cours de ces dernières années. Je lui suis sincèrement reconnaissante pour la confiance qu'il m'a témoigné depuis le début et tout au long de la thèse, pour son encadrement, ses conseils et ses orientations. J'espère avoir été digne de cette confiance.

Je remercie également les membres du jury de me faire l'honneur d'examiner et de juger cette thèse.

J'adresse une pensée particulière à mon très cher époux Saidi, pour sa patience et son zen légendaire qui ont su créer un climat favorable sans lequel ce travail n'aurait jamais pu être achevé.

Mes sincères remerciements s'adressent à mes parents, mes frères et ma sœur pour leur soutien moral, leur encouragement inconditionnel, et surtout pour la confiance qu'ils m'accordent.

Je remercie également mes amis pour leur soutien moral considérable et pour m'avoir aidée à résister durant toutes ces années.

Durant les trois années j'ai rencontré beaucoup de personnes que je ne citerai pas de peur d'en oublier, je remercie donc tout simplement tous ceux et celles qui me sont chers, qui ont participé de près ou de loin à l'aboutissement de ce travail.

À mes parents...

À mon époux...

À mon bébé...

À mes frères et sœur.

Résumé : Dans ce mémoire, nous abordons le problème d'aide à la décision pour permettre la satisfaction des demandes des clients par une entreprise. Ces demandes sont particulièrement celles qui divergent fortement des prévisions préalablement établies. Dans cette situation, le processus décisionnel peut être complexe. En plus, chaque entreprise appartient à un réseau d'entreprises plus simplement appelé chaîne logistique. Chaque acteur qui compose ce réseau prend en charge la mise en œuvre d'actions qui profitent à une partie ou à la totalité du réseau. Cette vision naturellement distribuée d'une chaîne logistique se prête bien à une démarche d'analyse orientée agents.

L'implémentation du système multi-agent (SMA) nécessite l'utilisation d'une plateforme multi-agent. Cependant, dans le cadre de ce mémoire, la plateforme JADE fondé sur le langage JAVA est la plus adéquate, en raison que cette dernière est la plus adaptée pour identifier les trois agents cognitifs de notre modèle en même temps elle fournit un ensemble complet de services conformes aux spécifications FIPA.

Mots clés : Systèmes Multi-Agent SMA, Chaîne Logistique, Agent UML, Aide à la décision, Système distribué.

ملخص: في هذه المذكرة، نتناول مسألة المساعدة في اتخاذ القرارات بما يسمح بتلبية طلبات عملاء الشركة، يمكن لعملية اتخاذ القرار أن تكون معقدة وهذا في الحالة عندما تكون الطلبات تختلف كثيرا عن التوقعات أو التقديرات التي أنشئت سابقا على مستوى الشركة بالإضافة. إلى ذلك، كل مؤسسة تنتمي الى شبكة من الشركات تسمى ببساطة سلسلة الامدادات أو سلسلة التوريد، كل طرف من الاطراف الفاعلة التي تدخل في تشكيل هذه الشبكة تدعم تنفيذ الإجراءات التي تعود بالنفع على جزء من الشبكة أو على الشبكة بأكملها و. بالتالي هناك حاجة لاستعمال تحليلي يعتمد على الأعوان نظر الطبيعة سلسلة التوريد الموزعة. يتطلب تجسيد النظام المتعدد الأعوان SMA استعمال قاعدة متعددة الأعوان. في هذه المذكرة القاعدة JADE المبنية على لغة JAVA هي الأنسب لتحقيق الأعوان المعرفية الثلاثة التي تمثل نموذجنا و في الوقت نفسه توفر مجموعة شاملة من الخدمات المتفقة مع المواصفات FIPA.

كلمات دلالية: نظام: موزع، سلسلة التوريد، دعم اتخاذ القرار، النظام المتعدد الأعوان SMA، AUML.

Abstract: In this paper, we address the problem of decision support to the satisfaction of the demands of customers by a company, These requests are particularly those that strongly differ from the forecasts established in advance. In this situation, the decision-making process can be complex. In addition, each company belongs to a network of companies called supply chain. Each actor who composed this network supports the implementation of actions that benefit to a part or the whole of the network. This vision naturally distributed in a supply chain is well suited to an analysis oriented agents approach. The implementation of multi-agent (SMA) system requires the use of a multi-agent platform. However, in the context of this thesis, the JADE platform based on the Java language is most appropriate, because the latter is more appropriate to identify the three cognitive agents in our model at the same time it provides a comprehensive set of services which are conform to FIPA specifications.

Keywords: Multi- Agents Systems (SMA), AUML, Supply Chain, Decision support, Distributed system.

Table Des Matières

Introduction Générale

1.1 Contexte de recherche.....	11
1.2 Problématique.....	12
1.3 Objectifs de mémoire.....	13
1.4 Structure de mémoire.....	14

Chapitre1 :Etat De L'art Et Contexte Générale De L'étude.

1 Introduction.....	19
2 Chaîne logistique.....	19
2.1 Chaîne logistique et organisation en réseau des entreprises.....	20
2.1.1 Système fragmenté.....	20
2.1.2 Système centralisé.....	21
2.1.3 Système hiérarchisé.....	21
2.1.4 L'organisation centralisé/ hiérarchisé.....	22
2.1.5 Système distribué.....	23
2.2 Les flux de la chaîne logistique.....	24
2.2.1 Le flux d'information.....	24
2.2.2 Flux physique.....	24
2.2.3 Flux financier.....	25
2.3 Gestion de la chaîne logistique.....	25
3 Le rôle de la chaîne logistique.....	25
3.1 Le processus Approvisionnement.....	26
3.2 Le processus Production.....	27
3.3 Le processus Distribution.....	27
3.4 Le processus Vente.....	27
4 Les déferents types de relations industrielles.....	28
5 Modèles de pilotage des chaînes logistiques.....	28
5.1 La modélisation SCOR.....	28
5.2 Problème liée au pilotage des chaînes logistiques distribuées.....	29
5.2.1 Problème de l'incertitude des informations et robustesse des décisions.....	30
5.2.2 L'effet bullwhip (coup de fouet).....	31
6 Les instruments de pilotage de telles structures.....	32
7 Les outils d'aide à la décision dans les SCM.....	33
7.1 La décision.....	33
7.1.1 Prendre une décision.....	34
7.1.2 L'aide à la décision.....	34
7.1.3 Une définition des systèmes d'aide à la décision.....	35

7.2 Pilotage des chaînes logistiques.....	36
8 La problématique de notre étude.....	38
8.1 Contexte de l'étude.....	39
8.2 Le but de travail.....	40
9 Conclusion.....	40

Chapitre 2 : Modèles D'aide A La Prise De Décision Pour Une Demande Imprévue Dans Une Chaîne Logistique.

1 Introduction.....	45
2 L'entreprise.....	45
2.1 Niveaux décisionnels.....	45
2.1.1 Niveau stratégique.....	46
2.1.2 Niveau tactique.....	46
2.1.3 Niveau Opérationnel.....	46
3 Problématique de prise d'une commande.....	46
3.1 Mécanismes de prise de décisions.....	47
3.1.1 Acteurs de la prise de décisions.....	47
3.1.2 Processus de la prise de décision.....	47
3.2 Impacts des décisions opérationnelles dans le fonctionnement interne de l'entreprise.....	49
3.3 Impacts des décisions opérationnelles sur la chaîne logistique.....	49
4 Problèmes liés à la prise de commandes.....	50
5 Processus de prise de commande et présentation globale de notre approche.....	51
5.1 Mécanisme de traitement d'une nouvelle demande.....	52
5.2 Arrivée d'une nouvelle demande.....	53
5.3 Cas n°1 – Satisfaction de la demande en utilisant les produits disponibles dans les stocks	53
5.4 Cas n°2 et 3 – Satisfaction de la demande par production.....	54
5.5 Cas n°4 – Disponibilité de la matière première.....	56
5.5 Cas n°5 – Condition insuffisantes pour la réalisation de la demande.....	57
6 Problèmes de manque de composants.....	59
6.1 Positionnement du problème.....	59
6.2 Démarche de résolution.....	60
6.2.1 Création des scénarios.....	60
6.2.2 Recherche de fournisseurs.....	60
6.2.3 Envoi d'une demande aux fournisseurs.....	61
6.2.4 Réception des réponses.....	61
6.2.5 Sélection des propositions.....	61
6.3 Modèles d'aide à la décision	62
6.3.1 Faisabilité d'un scénario d'approvisionnement	63
6.3.2 Modèle d'optimisation de la sélection.....	63
7 Problème d'insuffisance des conditions de production.....	66
7.1 Démarche de résolution.....	66
8 Synthèse de l'approche proposée.....	67
9 Conclusion.....	71

Chapitre3 :Systèmes Multi-Agent : Une Approche Pour Les Architectures Distribuées D'aide A La Décision.

1 Introduction.....	74
2 La notion d'agent.....	74
3 Les Systèmes Multi-Agents (SMA).....	75
3.1 Définition.....	75
3.2 Interactions.....	76
3.3 Langages de communication entre agents.....	77
3.4 Typologie des agents.....	78
3.4 Architectures d'intégration des agents.....	81
4 Applications des systèmes multi agents.....	82
5 Méthodologies de développement de SMA.....	86
6 Quelques Plates-formes de développement des SMA	90
7 Conclusion	93

Chapitre4 :Architecture Distribuée A Base D'agent Pour L'aide A La Décision Dans La Chaîne Logistique.

1 Introduction.....	96
2 Justification du choix.....	96
3 Notre approche multi agents.....	97
3.1 Identification des agents et leurs rôles.....	97
3.2 Structure statique du système.....	99
3.2.1 Structure de l'agent « App ».....	99
3.2.2 Agent « Dist »	102
3.2.3 Agent « Fab »	105
4 Protocoles de négociation.....	106
4.1 Diagramme de séquence.....	107
4.2 Négociation heuristique.....	108
5 Modélisation des comportements des agents.....	109
5.1 Comportement des agents.....	109
5.1.1 Comportement de l'agent « Dist ».....	110
5.1.2 Comportement de l'agent « App »	112
5.1.3 Comportement de l'agent « Fab ».....	114
6 Exemple d'interaction entre les agents pour la prise de commande ferme....	115
7 Conclusion.....	117

Chapitre5 : Implémentation Et Validation

1 Introduction.....	120
2 Implémentation.....	120
2.1 La norme FIPA pour les systèmes multi-agents.....	120
2.2 Modèle d'agents JADE et notion de comportement	122
2.2.1 Modèle d'agents.....	122
2.2.2 Critères de choix.....	125
2.3 Des diagrammes d'états-transitions vers l'implémentation des agents.....	127
2.4 Architecture fonctionnelle de JADE et cycle de vie d'un agent.....	130
2.4.1 Architecture de la plate-forme JADE.....	130
2.4.2 Cycle de vie d'un agent.....	131
3 Validation de notre approche.....	133
3.1 Évaluation des performances dans les chaînes logistiques par simulation d'un système multi agent.....	133
3.2 Données prises en compte dans le modèle	134
3.3 Illustration de notre approche de prise d'une demande imprévue	135
3.4 Simulation.....	141
3.5 Résultats des simulations.....	142
4 Conclusion.....	143
Conclusion Générale Et Perspectives.....	144
Références Bibliographiques.....	147

Liste de Figures et Tables

Chapitre 1

Figures

<i>Figure 1.1 : Exemple de chaîne logistique.....</i>	20
<i>Figure 1.2 : Représentation du système fragmenté.....</i>	21
<i>Figure 1.3 : Représentation du système centralisé.....</i>	21
<i>Figure 1.4 : Représentation du système hiérarchisé.....</i>	21
<i>Figure 1.5 : organisation centralisée / hiérarchisée.....</i>	22
<i>Figure 1.6 : Représentation du système intégré.....</i>	23
<i>Figure 1.7 : Représentation du système hiérarchisé.....</i>	23
<i>Figure 1.8 : réseau d'entreprises distribuée.....</i>	23
<i>Figure 1.9 : Niveaux stratégique, tactique et opérationnel dans une SC.....</i>	26
<i>Figure 1.10 : le modèle SCOR.....</i>	29
<i>Figure 1.11 : Sources d'incertitudes dans la CL.....</i>	30
<i>Figure 1.12: exemple de l'effet bullwhip.....</i>	32
<i>Figure 1.13 : Modèle d'aide à la décision.....</i>	34
<i>Figure 1.14 : L'offre logicielle du SCM.....</i>	36
<i>Figure 1.15 : Le positionnement relatif des ERP et des APS.....</i>	37
<i>Figure 1.16 : Structure de la chaîne logistique étudiée.....</i>	39

Chapitre 2

Figures

<i>Figure 2.1 : Exemple de cheminement interne des décisions.....</i>	48
<i>Figure 2.2 : propagation externe de la prise de décision.....</i>	48
<i>Figure 2.3 : Exemple de re-planification.....</i>	55
<i>Figure 2.4 : Diagramme de l'approche pour la résolution du problème de disponibilité des matières premières.....</i>	57
<i>Figure 2.5 : Diagramme de l'approche pour la résolution du problème de recherche d'entreprise de produit finit.....</i>	58
<i>Figure 2.6 : Algorithme de recherche de la quantité.....</i>	69
<i>Figure 2.7 : Synthèse de l'approche globale</i>	70

Chapitre3

Figures

<i>Figure 3.1: Représentation imagée d'un agent en interaction avec son environnement et les autres agents.....</i>	76
<i>Figure 3.2 : Composition d'un agent simple reflex.....</i>	78
<i>Figure 3.3 : composition d'un Agent à modèle basé sur le réflexe.....</i>	79
<i>Figure 3.4 : Composition d'un agent basé sur le but.....</i>	79
<i>Figure 3.5 : Composition d'un agent basé sur le service.....</i>	80
<i>Figure 3.6 : Composition d'un agent basé sur la connaissance.....</i>	80
<i>Figure 3.7 : Les relations entre les modèles de la méthodologie GAIA.....</i>	87
<i>Figure 3.8 : Les trois concepts centraux d'AALAADIN.....</i>	87
<i>Figure 3.9 : L'interface de la plate-forme JADE.....</i>	92

Chapitre4

Figures

<i>Figure 4.1 : Agents cognitifs de l'entreprise.....</i>	98
<i>Figure 4.2 : Architecture distribuée d'aide à la décision dans une chaîne logistique.....</i>	98
<i>Figure 4.3 : Structure de l'agent acheteur.....</i>	99
<i>Figure 4.4 : Le niveau implémentation pour l'agent « App ».....</i>	102
<i>Figure 4.5 : Structure de l'agent Distribution « Dist ».....</i>	103
<i>Figure 4.6 : Le niveau implémentation pour l'agent « Dist ».....</i>	104
<i>Figure 4.7 : Le niveau implémentation pour l'agent « Fab ».....</i>	106
<i>Figure 4.8 : Négociation heuristique.....</i>	108
<i>Figure 4.9 : Négociation heuristique.....</i>	109
<i>Figure 4.10 : Diagramme d'états-transitions de l'agent « Dist ».....</i>	110
<i>Figure 4.11 : Diagramme d'états-transitions de l'agent « App ».....</i>	111
<i>Figure 4.12 : Diagramme d'états transitions de l'agent « Fab ».....</i>	114

Chapitre5

Figures

<i>Figure 5.1 : Services fournis par la Plate-forme FIPA.....</i>	121
<i>Figure 5.2 : Etapes d'exécution d'un thread d'agent.....</i>	124
<i>Figure 5.3 : Hiérarchie de quelques classes Behaviour de la plateforme JADE.....</i>	125
<i>Figure 5.4 : Protocole d'interaction FIPA Propose.....</i>	126
<i>Figure 5.5 : Transformation partielle du modèle de comportement de l'agent « Dist » vers le code Java (Agent JADE).....</i>	129
<i>Figure 5.6 : Plateforme et récipients (container).....</i>	131
<i>Figure 5.7 : Gestion des messages envoyés et reçus par les agents JADE.....</i>	131
<i>Figure 5.8 : Le cycle de vie d'un agent défini par FIPA.....</i>	132
<i>Figure 5.9 : Fenêtre principale de l'Agent Distribution (Dist).....</i>	138
<i>Figure 5.10 : Fenêtre de l'Agent Approvisionnement.....</i>	139

Liste des Abréviations

<i>ACL</i>	<i>Agent Communication Language</i>
<i>APS</i>	<i>Advanced Planning Systems</i>
<i>AUML</i>	<i>Agent Unified Modeling Language</i>
<i>BDI</i>	<i>Belief, Desire and Intention</i>
<i>CL</i>	<i>Chaîne Logistique globale</i>
<i>ERP</i>	<i>Enterprise Resource Planning</i>
<i>FIPA</i>	<i>Foundation for Intelligent Physical Agents</i>
<i>JESS</i>	<i>Java Expert System Shell</i>
<i>JADE</i>	<i>Java Agent Development Framework</i>
<i>KIF</i>	<i>Knowledge Interchange Format</i>
<i>KQML</i>	<i>Knowledge Query and Manipulation Language</i>
<i>RMA</i>	<i>Remote Management Agent</i>
<i>SMA</i>	<i>Système Multi-Agent</i>
<i>SCOR</i>	<i>Supply-Chain Operations Reference-model</i>
<i>SCC</i>	<i>Supply Chain Council</i>
<i>SCM</i>	<i>Supply Chain Management</i>

Introduction Générale

1.1 Contexte de recherche

De nos jours, les entreprises évoluent dans un environnement qui confronte plusieurs problèmes : des marchés saturés, une compétitivité accrue, des clients plus exigeants et moins fidèles, etc. Dans un tel environnement, la compétitivité des entreprises dépend de plus en plus de leur flexibilité et de leur capacité d'innover, tant dans leur structure organisationnelle, leur mode de production que dans leur mode d'échange avec les clients et les fournisseurs.

La nouvelle source d'avantage compétitif devient la capacité à satisfaire au mieux les demandes des clients en termes de livraison [152]. F. Doche confirme dans [153] que l'acquisition d'un nouveau client peut coûter jusqu'à vingt fois plus cher que la fidélisation d'un client existant. Une réduction des défections de 5% génère une hausse des bénéfices de 25% à 85%. Il ne sert donc à rien de dépenser des fortunes pour recruter de nouveaux clients si l'on n'est pas capable, avant tout, de fidéliser sa propre clientèle. Aujourd'hui, le comportement client a évolué, les clients sont de plus en plus exigeants et les industriels doivent répondre au mieux à leurs besoins en matière de personnalisation, de rapidité de traitement des commandes et de livraison.

La recherche de compétitivité s'est de plus traduite par une spécialisation et un recentrage des activités des entreprises sur leur cœur de métier. Ainsi, dans un contexte de globalisation des échanges, cette spécialisation a eu pour effet de multiplier les relations interentreprises. Par conséquent, la coordination des différents acteurs extérieurs, développant des savoir-faire spécialisés, est devenue un défi important pour l'amélioration de la gestion de production des entreprises. Ainsi, les entreprises se sont structurées en réseaux distribués dans le but de mieux prendre en compte les besoins du client final à moindre coût en essayant d'appréhender l'ensemble du processus d'élaboration d'un produit depuis la matière première. Au sein de ces réseaux, appelés chaînes logistiques, des mécanismes de

collaboration sont apparus pour faciliter les relations entre un donneur d'ordres et ses sous-traitants ou fournisseurs. Ces pratiques reposent notamment sur une bonne utilisation des informations échangées.

La prise de décision dans ce genre de structure en réseau d'entreprises est un processus complexe qui demande l'interaction entre une partie ou la totalité des partenaires qui le compose. L'une des difficultés dans ce domaine est de garantir la cohérence des décisions prises dans le réseau. L'un des phénomènes résultant de l'incohérence des décisions prises le plus connu en gestion des réseaux d'entreprises est l'effet coup de fouet (*Bullwhip*). Pour parvenir à limiter au maximum ces incohérences, la coordination des actions et des décisions prises semble être le meilleur moyen.

L'ensemble de ce mémoire s'intéresse au problème de prise de décision pour la satisfaction de commandes imprévues. Ce problème est majeur, car comme nous l'avons dit préalablement, la satisfaction de commandes est le but essentiel de toute entreprise en réseau. Dans ce contexte, la prise de décision est opérationnelle, car les délais impartis pour répondre à une commande sont courts. En outre, nous nous plaçons volontairement dans la situation pour laquelle les demandes ne correspondent pas aux prévisions préalablement élaborées. Enfin, les entreprises qui forment le réseau considéré sont de natures autonomes dans leurs prises de décisions et dans leur système d'information. Il s'agit alors, d'une architecture distribuée.

1.2 Problématique

La problématique de la fidélisation et la satisfaction des clients en matière de rapidité de traitement des commandes et de livraison a suscité les intérêts d'un nombre important de chercheurs. Plusieurs techniques et méthodes ainsi que des outils informatiques d'aide à la décision sont développées et mises en œuvre entre les acteurs d'une chaîne logistique en vue de coordonner leurs décisions et répondre ainsi à ce besoin. C'est dans le cadre de ces travaux que s'inscrit la principale contribution présentée dans ce mémoire.

Nous visons à proposer des solutions liées aux chaînes logistiques au moyen des agents cognitifs qui soient capables d'aider à la prise de décision, Ceci, en vue de permettre aux différentes entreprises d'avoir la capacité à satisfaire au mieux les demandes fermes des clients en termes de livraison, plus particulièrement, lors de la présence :

- des commandes imprévues : représentent l'écart entre la demande prévisionnelle et la demande initiale du client.

- des exceptions (problème de production, erreur sur prévisions, retard de livraison, etc.) pouvant altérer le bon déroulement d'un processus de fabrication, d'approvisionnement ou de livraison ce qui implique la non fidélisation de client.

Le but de notre recherche est de mettre en place les fondements d'une démarche globale de collaboration entre les partenaires existants dans la chaîne logistique distribuée. Cela en vue de la réalisation des opérations de prise de commandes fermes dans le contexte où les délais de réponses sont courts car nous focalisant sur la prise de décision au niveau opérationnel.

1.3 Objectifs de mémoire

Les objectifs de ce mémoire sont :

- développer une architecture distribuée à base de système multi-agent (SMA) pour l'aide à la prise de décision dans le contexte de la chaîne logistique, nous modélisons l'entreprise par trois agents cognitifs « Agent Distribution, agent fabrication et agent approvisionnement ». Chaque entité travaille conjointement avec les autres afin de réaliser les buts de l'entreprise.
- Nous proposons aussi un processus de prise de décisions pour la satisfaction d'une commande ferme et la résolution des problèmes liés à la prise de commande causé par des événements imprévues.
- Afin de donner un cadre méthodologique au processus de modélisation, nous avons adopté le langage AUML (Agent Unified Modeling Language). Nous donnons une grande importance au diagramme d'états-transitions (statechart diagram) [138] sera utilisé pour modéliser les comportements des agents.
- Vu la diversité des plates-formes utilisées par les acteurs industriels et la nature distribuée de la chaîne logistique, nous avons choisi la plateforme de développement JADE(Java Agent Development Framework) [124] pour implémenter notre architecture. Elle est gratuite et est basée sur le langage JAVA.

1.4 Structure de mémoire

Après la présentation des principaux objectifs de notre travail, la suite du présent mémoire est organisée comme suit :

Chapitre 1

Dresse globalement un état de l'art sur la chaîne logistique. Nous définissons tout d'abord le pilotage distribué des chaînes logistiques et nous montrons les problèmes liés aux flux d'informations et de produits. Nous citons ensuite les principaux outils et approches utilisés dans le cadre de la gestion de la chaîne logistique afin de pallier aux problèmes ayant une influence directe ou indirecte sur la satisfaction des clients enfin nous présentons la problématique générale de ce mémoire qui concerne la prise de décision pour la satisfaction de commandes imprévues.

Chapitre 2

Aborde le problème de la prise de commandes imprévues dans un contexte de décisions opérationnel. Ensuite, nous décrivons le processus induit par cette prise de décision qui nous permet de mettre en avant les sous problèmes à aborder à savoir : le problème de disponibilité des quantités de composants nécessaires à la réalisation de telles commandes, nous proposons une démarche de recherche de composants ainsi qu'un modèle de sélection des propositions des fournisseurs potentiels des matières premières et des fournisseurs des produits finis. Ce modèle comporte un outil mathématique d'aide à la décision pour la sélection des fournisseurs de matière première et un algorithme pour la sélection des fournisseurs de produit fini tout en garantissant un résultat optimal au sens du coût total d'acquisition.

Chapitre 3

Nous introduisons dans le chapitre 3 les définitions de base concernant le concept d'agent et de système multi-agent. Nous présentons ensuite quelques recherches dans le domaine des agents et chaîne logistique puis nous donnons une synthèse sur les méthodologies d'analyse et conception ainsi que les plates-formes de développement des systèmes multi-agent. Puis nous montrons les grandes lignes et critères qui ont guidé nos choix concernant le langage de modélisation et la plate-forme utilisés pour le développement de notre architecture. Enfin, Nous terminons ce chapitre par une conclusion.

Chapitre 4

En s'appuyant sur le langage AUML, le chapitre 4 présente la modélisation d'une chaîne logistique à base de SMA. Nous identifions les agents nécessaires et nous proposons des protocoles de négociation. Les comportements des agents et leurs interactions seront aussi modélisés à l'aide des diagrammes d'états-transitions d'AUML. Enfin, nous terminons ce chapitre par une conclusion récapitulant l'originalité de notre contribution.

Chapitre 5

Est consacré à la phase d'implémentation de l'architecture proposée sous l'environnement JADE. Après une présentation du modèle d'agent JADE, nous proposons des règles de transformation permettant l'obtention de squelette d'une classe d'agent dans JADE à partir de son modèle de comportement (statechart), en particulier, l'établissement d'une correspondance entre les diagrammes d'états-transitions AUML et les comportements de JADE et la validation de notre approche.

Chapitre 1

Etat De L'art Et Contexte Générale De L'étude

1 Introduction.....	19
2 Chaîne logistique.....	19
2.1 Chaîne logistique et organisation en réseau des entreprises.....	20
2.1.1 Système fragmenté.....	20
2.1.2 Système centralisé.....	21
2.1.3 Système hiérarchisé.....	21
2.1.4 L'organisation centralisé/ hiérarchisé.....	22
2.1.5 Système distribué.....	23
2.2 Les flux de la chaîne logistique.....	24
2.2.1 Le flux d'information.....	24
2.2.2 Flux physique.....	24
2.2.3 Flux financier.....	25
2.3 Gestion de la chaîne logistique.....	25
3 Le rôle de la chaîne logistique.....	25
3.1 Le processus Approvisionnement.....	26
3.2 Le processus Production.....	27
3.3 Le processus Distribution.....	27
3.4 Le processus Vente.....	27
4 Les différents types de relations industrielles.....	28
5 Modèles de pilotage des chaînes logistiques.....	28
5.1 La modélisation SCOR.....	28
5.2 Problème liée au pilotage des chaînes logistiques distribuées.....	29
5.2.1 Problème de l'incertitude des informations et robustesse des décisions.....	30
5.2.2 L'effet bullwhip (coup de fouet).....	31
6 Les instruments de pilotage de telles structures.....	32
7 Les outils d'aide à la décision dans les SCM.....	33
7.1 La décision.....	33

7.1.1 Prendre une décision.....	34
7.1.2 L'aide à la décision.....	34
7.1.3 Une définition des systèmes d'aide à la décision.....	35
7.2 Pilotage des chaînes logistiques.....	36
8 La problématique de notre étude.....	38
8.1 Contexte de l'étude.....	39
8.2 Le but de travail.....	40
9 Conclusion.....	40

1 Introduction

La compétition globale et les changements rapides des besoins des clients conduisent à des mutations majeures dans les styles de production ainsi que dans la configuration des entreprises. Ainsi, les flux d'informations et de produits circulant dans la logistique doivent être bien maîtrisés et gérés. La problématique des événements imprévus présente un challenge important à surmonter dans les chaînes logistiques distribuées, car c'est l'un des facteurs pouvant causer des ruptures de stock. Ce qui se traduira par un mauvais niveau de service dans l'optique de la fidélisation du client final en matière de rapidité de traitement et de livraison des commandes, problématique à laquelle nous portons un grand intérêt dans ce mémoire.

Nous organisons ce chapitre de la manière suivante : dans la première partie, nous détaillons l'ensemble des caractéristiques et définitions des chaînes logistiques et nous donnons un aperçu des travaux réalisés dans ce domaine. Cela nous permet d'avoir une vision globale sur le domaine étudié. Dans le second volet de ce chapitre, nous abordons la problématique ainsi que le contexte général de l'étude que nous traitons tout au long de ce mémoire.

2 Chaîne Logistique

C'est un concept relativement récent. On définit assez souvent la chaîne logistique comme « *La suite des étapes de production et de distribution d'un produit depuis les fournisseurs des fournisseurs des producteurs, jusqu'aux clients de ses clients* » (Supply Chain Council).

Afin de mieux comprendre le concept de Chaîne Logistique CL (Supply Chain SC)(*Figure 1.1*), nous proposons d'effectuer une revue des définitions de ce terme, utilisées dans la littérature.

Christopher [5] définit la chaîne logistique comme étant « *le réseau d'entreprises qui participent, en amont et en aval, aux différents processus et activités qui créent de la valeur sous forme de produits et de services apportés au consommateur final. En d'autres termes, une chaîne logistique est composée de plusieurs entreprises, en amont (fourniture de matières et composants) et en aval (distribution), et du client final* ».

Lummus [6] a, quant à lui, définit la chaîne logistique comme étant « *le réseau d'entités par lequel le flux matériel passe. Ces entités incluent fournisseurs, transporteurs, sites d'assemblages, centres de distribution, détaillants et clients* ».

Une définition plus générale est celle proposée par Poirier [7] : « Une chaîne logistique est le système grâce auquel les entreprises amènent leurs produits et leurs services jusqu'à leurs clients ».

Pour notre travail, nous retenons la définition de [8] qui définissent la chaîne logistique ainsi : « une chaîne logistique consiste en toutes les étapes impliquées directement ou indirectement dans la satisfaction de la requête d'un client. La chaîne logistique inclut non seulement le fabricant et ses fournisseurs, mais aussi les transporteurs, les centres d'entreposage, les détaillants et les clients eux-mêmes ».

Ainsi, dans ce travail, nous considérons que la chaîne logistique est composée de tous les partenaires qui interviennent dans la réalisation d'un produit ou d'une famille de produits donnés. Les relations entre les partenaires se basent sur la relation client/fournisseur. Chacun des partenaires est libre dans sa prise de décision, il peut accepter ou refuser une demande d'un de ses partenaires

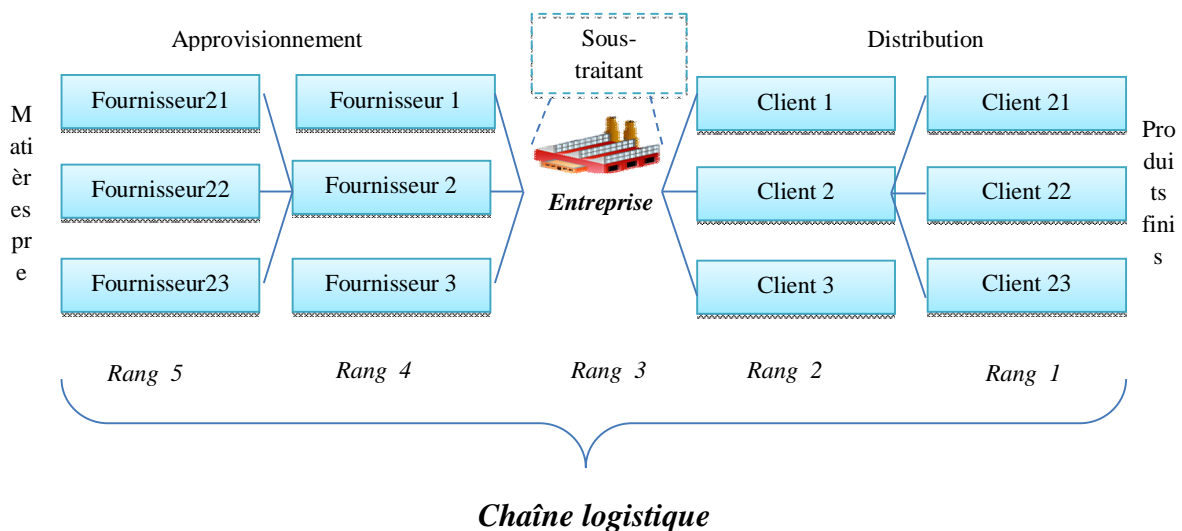


Figure 1.1 : Exemple de chaîne logistique

2.1 Chaînes logistiques et organisation en réseau des entreprises

Les entreprises ont tendance à se regrouper en structure homogène ou non (organisation en réseau d'entreprises) qui leur permet d'atteindre leurs objectifs respectifs et de faire face à la concurrence [140]. Les entreprises sont alors organisées en réseaux. La littérature distingue différents types d'organisation en réseaus nous appuyons pour ce point sur les travaux réalisés dans la thèse de doctorat de Monteiro T. [141].

2.1.1 Système fragmenté

Structurellement le plus simple, ce système d'organisation (figure 1.2) est caractérisé

par l'absence de relations physiques ou virtuelles entre chaque entité. Il est archaïque et n'a pas pour vocation une quelconque coordination entre les entités qui le compose.

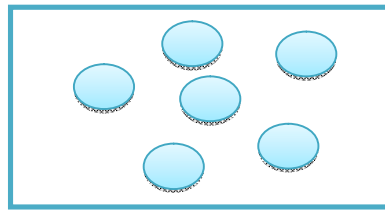


Figure 1.2 : Représentation du système fragmenté.

2.1.2 Système centralisé

Ce type de système est caractérisé par le contrôle d'une entité de l'ensemble de l'organisation, la coordination entre les entités se fait par l'intervention de l'entité de contrôle.

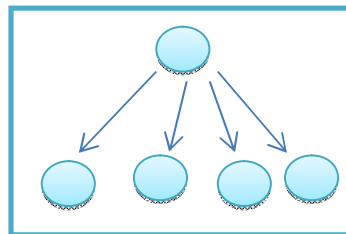


Figure 1.3 : Représentation du système centralisé.

2.1.3 Système hiérarchisé

Il est caractérisé par une structure (*figure 1.4*) dans laquelle les niveaux supérieurs définissent les contraintes et les objectifs à atteindre. On peut difficilement parler ici de coopération. La synchronisation est réalisée par des décisions unilatérales qui sont répercutées en fonction d'une hiérarchie décisionnelle.

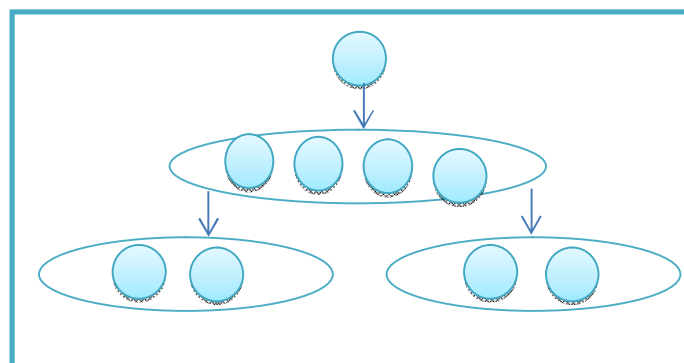


Figure 1.4 : Représentation du système hiérarchisé.

2.1.4 L'organisation centralisé / hiérarchisé

Ce type d'organisation se caractérise par le fait que toute entité possède un centre de décision qui est autonome dans ses prises de décisions à condition de respecter des contraintes globales fournies par le système de décision coordinateur. La hiérarchie est ainsi plus ou moins souple selon la marge de décision laissée aux niveaux inférieurs. Dans ces organisations il existe un ou plusieurs (super) centres de décision qui gèrent ensemble et hiérarchiquement l'intégralité du réseau (*figure 1.5*). Cela implique que chacun des autres centres est soumis non seulement au contrôle du centre coordinateur mais aussi doit appliquer les décisions de ce dernier.

L'avantage de ce type d'organisation est que toutes les phases du processus de fabrication sont sous contrôle du centre dirigeant. Cela implique l'assurance de la cohérence de l'ensemble des décisions prises par les autres centres de décision.

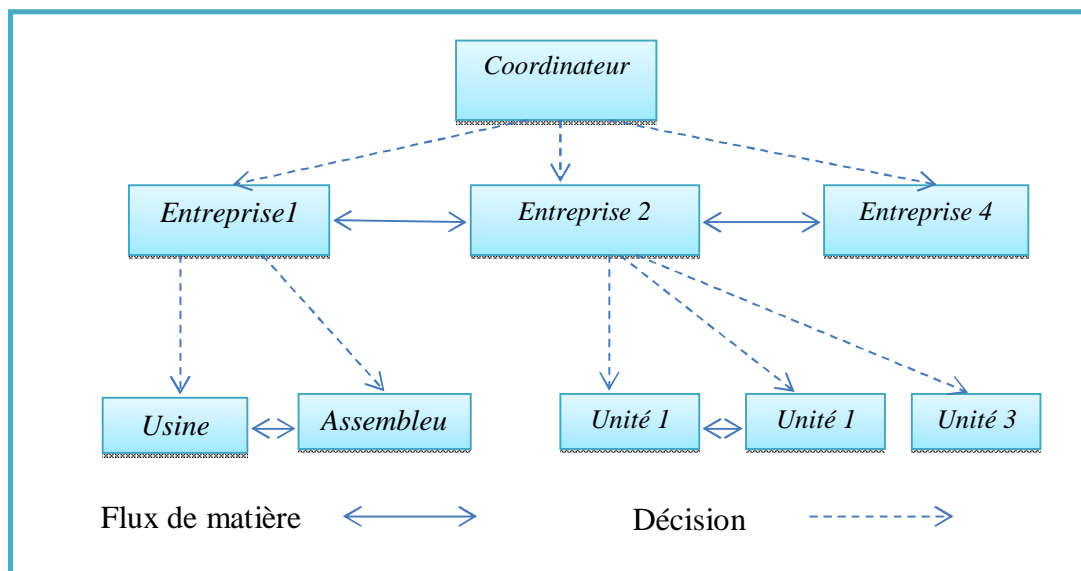


Figure 1.5 : organisation centralisée / hiérarchisée.

L'inconvénient réside dans le fait que ce type d'organisation est mal adapté à l'environnement actuel des entreprises. Cela est dû à la difficulté ou à l'impossibilité de réunir l'ensemble des acteurs dans de telles structures en raison de leur appartenance à des organismes partenaires voire concurrents.

Ces organisations sont aussi identifiées sous le nom de systèmes hiérarchiques intégrés. Par rapport aux architectures hiérarchiques simples, ces architectures (*figure 1.6*) sont coordonnées. Les îlots peuvent s'informer. Ces échanges d'informations peuvent donner lieu à une coopération sur le même niveau hiérarchique. C'est le cas par exemple de la co-traitance qui permet à un groupement de fournisseurs de répondre ensemble à la demande d'un donneur d'ordres commun.

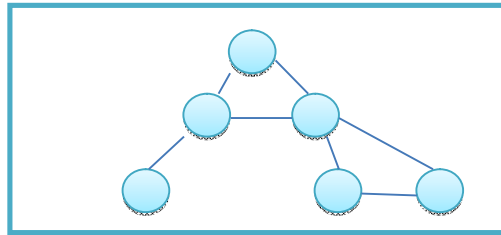


Figure 1.6 : Représentation du système intégré

2.1.5 Système distribué

Il constitue des hétérahies dans lesquelles les îlots ne sont plus subordonnés par des îlots hiérarchiquement supérieurs. Dans ces architectures (figure 1.7), la coopération de ces îlots dépend de leur interopérabilité (coopération de leurs services, ressources).

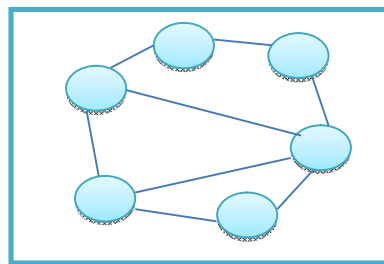


Figure 1.7 : Représentation du système hiérarchisé.

Ce type d'organisation (figure 1.8) se caractérise par le fait que chacun des centres de décision, qui représente un partenaire, est totalement autonome dans ses prises de décisions. Il n'existe pas de structure dirigeante de l'ensemble de l'organisation.

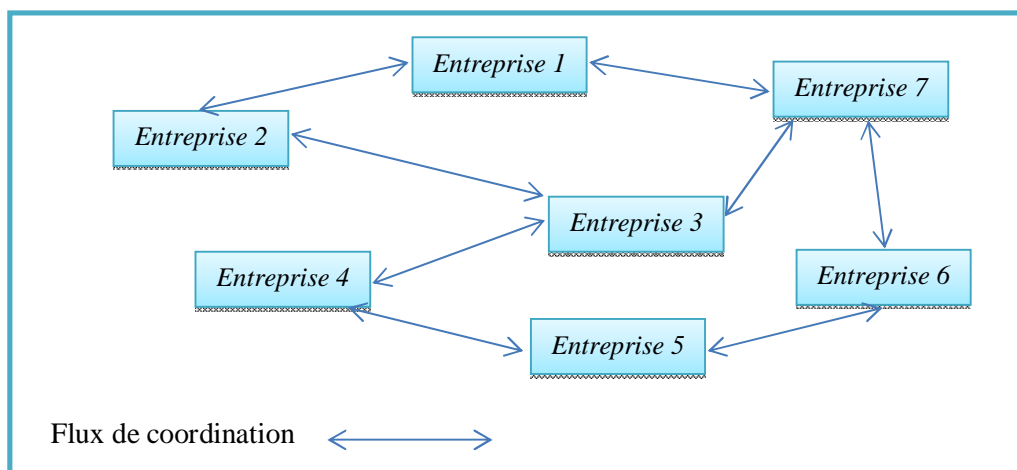


Figure 1.8 : réseau d'entreprises distribuée.

L'avantage de ce type d'organisation réside dans le fait que toute entité peut y intervenir sans prise de contrôle d'une autre entité.

Par contre il est difficile de mesurer dans ce type d'organisation la cohérence et l'influence que peuvent avoir les décisions prises localement sur le reste du système.

Dans notre recherche, nous nous intéressons plus particulièrement à ces organisations en réseaux d'entreprises dits distribués, pour lesquels une grande partie des outils de gestion des négociations et des prises de décision restent à inventer et à valider.

2.2 Les flux de la chaîne logistique

Les entreprises appartenant à une même chaîne logistique sont reliées classiquement par des flux de produits, des flux d'informations et des flux financiers [13], [14], [15].

2.2.1 Le flux d'information

Les informations remontent la CL puisque ce flux part des clients et, de proche en proche, traverse l'ensemble de la chaîne jusqu'au dernier des fournisseurs. Le flux d'information se compose au minimum des commandes émises par les clients à leurs fournisseurs. On distingue, en général, deux types d'informations :

- celles concernant les commandes fermes, c'est-à-dire, les commandes réelles, confirmées qui considèrent une quantité de produit à livrer à une date et dans un lieu donnés,
- celles concernant les commandes prévisionnelles. Elles correspondent à des prévisions sur les demandes futures à partir des historiques des ventes et des prévisions sur l'évolution des marchés.

Les commandes prévisionnelles sont analogues à des commandes fermes sauf qu'elles ne sont pas définitives et peuvent être modifiées.

Il peut y avoir d'autres types d'informations qui circulent dans le but d'aider les acteurs de la relation client/fournisseur dans leur prise de décision. Ce flux peut être bidirectionnel, le fournisseur pouvant informer son client sur les retards à la livraison, les quantités à livrer, etc.

2.2.2 Flux physique

Le flux de produit traverse la CL dans le sens orienté des fournisseurs aux clients. Il correspond aux flux de matières premières, composants de produits, produits semi-finis et produits finis entre les entreprises. Les équipements d'usinage (ex. machines, convoyeurs), de transport (ex. véhicules de transport de marchandises) et le matériel d'investissement (ex. outillage) ne font pas partis de ce type de flux.

Le transfert de produit peut être bidirectionnel : il circule évidemment du fournisseur vers les clients, mais il peut aussi, dans le cas d'une relation de sous-traitance, circuler du donneur d'ordre vers le sous-traitant [16]. Par exemple, l'entreprise fournit à son sous-traitant la matière première et récupère le produit par la suite.

2.2.3 Flux financier

Le flux financier regroupe les paiements et les arrangements financiers divers (crédits, mensuralisation des paiements, etc.) entre les clients et les fournisseurs. Il est donc lui aussi bidirectionnel.

Dans ce mémoire, nous allons prendre en compte uniquement les flux de matières ainsi que les flux d'informations. Nous considérons que les flux financiers sont la conséquence directe d'échange de matières ou de services (incluant des informations) entre deux partenaires.

2.3 Gestion de la chaîne logistique

Il existe une distinction entre la « *chaîne logistique* » et la « *gestion de la chaîne logistique* ». En effet, la gestion de chaîne logistique regroupe les approches, processus et fonctions indispensables pour la réduction des coûts d'une chaîne logistique et l'augmentation de sa flexibilité en vue d'optimiser sa performance.

Ici encore, on relève plusieurs définitions de la gestion de la chaîne logistique [9]. Beaucoup d'auteurs soulignent la difficulté de définir le SCM. Voici quelques définitions, issues notamment de [10] :

Vakharia [11] définit la SCM comme étant « *l'art et la science de créer et d'accentuer les rapports synergiques entre les partenaires d'une même chaîne logistique ayant comme objectif commun de livrer, juste à temps, les bons produits et les bons services au bon client, avec la meilleure quantité* ».

Simchi-Levi propose dans son ouvrage la définition [12] « *Le SCM est une stratégie qui vise à la fois la réduction des frais globaux, permettant une position plus concurrentielle à toutes les différentes parties de la chaîne logistique, et l'optimisation de la satisfaction du client final par une plus grande adaptabilité des systèmes de production et de distribution* ».

3 le rôle de la chaîne logistique

Un processus opérationnel (ou *Business Process*, en anglais) est un ensemble d'activités qui définit des rôles et des relations (Figure 1.9), et qui systématise l'organisation et la politique d'une entreprise dans le but d'atteindre certains des objectifs de cette entreprise. Nous rappelons ici les quatre processus principaux d'une entreprise, qui sont l'approvisionnement, la production, la distribution et la vente. Stadtler et Kilger [17] se réfèrent à ces quatre processus-clés pour proposer la classification SCP-Matrix des tâches de planification en fonction des niveaux décisionnels.

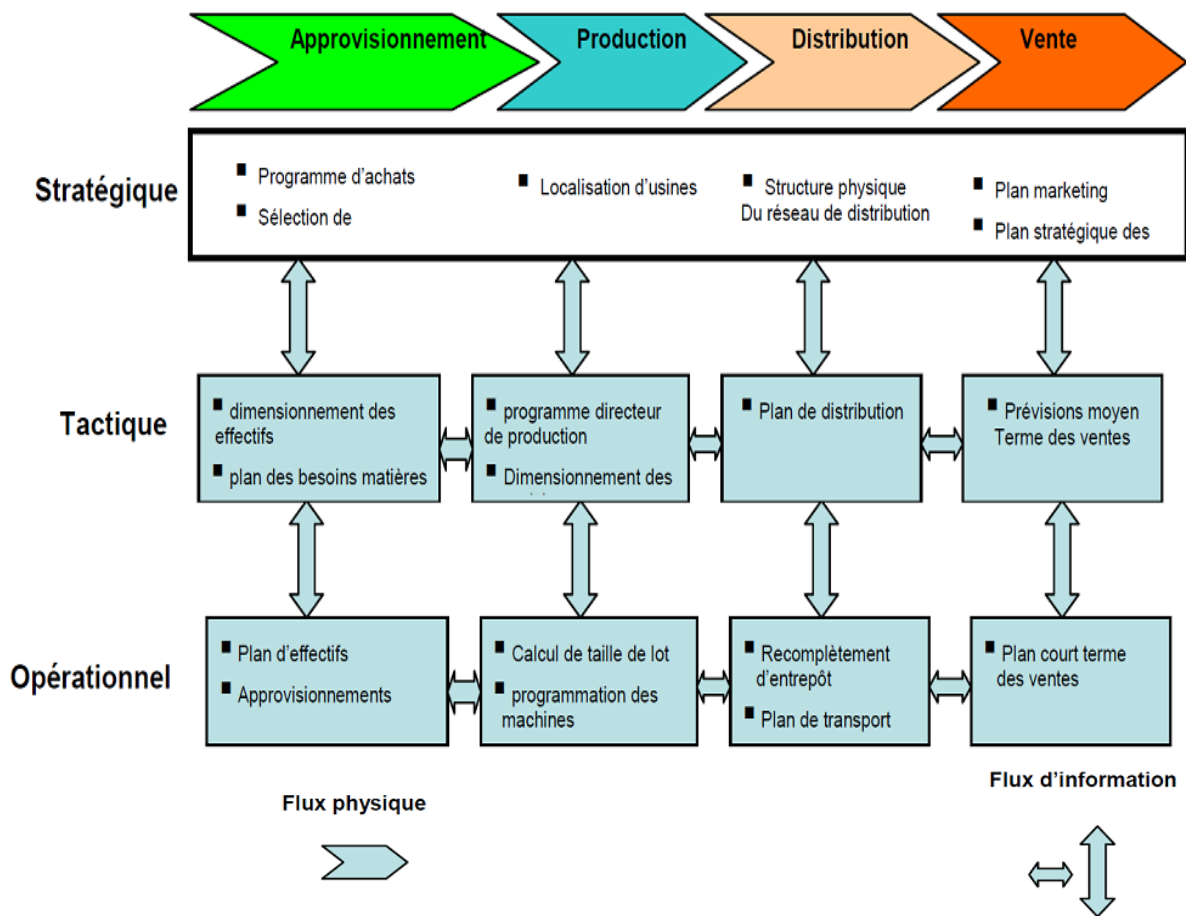


Figure 1.9 : Niveaux stratégique, tactique et opérationnel dans une SC [44]

3.1 Le processus Approvisionnement

Le processus Approvisionnement se concentre sur la fourniture de tous les composants nécessaires à la fabrication. Deux grandes phases sont ici à distinguer. La première phase consiste à sélectionner les fournisseurs de l'entreprise. Le choix des fournisseurs peut se faire sur différents critères comme la qualité, le prix, les délais de réapprovisionnement des matières premières ou composants, mais aussi leur capacité de production, leur facilité à accepter une demande très variable, leur possibilité de faire évoluer techniquement les composants... Il est possible de sélectionner un fournisseur unique par produit ou, au contraire, des sources multiples qui se partagent la demande, en

minimisant ainsi le risque de rupture de livraison.

Les fournisseurs étant déterminés, la seconde phase du processus Approvisionnement consiste à passer les commandes des composants à ces fournisseurs en fonction de la production à réaliser. Il s'agit aussi de vérifier que ces composants sont livrés dans de bonnes conditions, c'est-à-dire de vérifier que la livraison comporte les bons composants, de qualité requise, en quantité conforme et au bon moment.

Le processus Approvisionnement regroupe ainsi toutes les relations avec les fournisseurs pour assurer les niveaux de stocks en composants nécessaires et suffisants pour la fabrication.

3.2 Le processus Production

Le processus Production concerne l'ensemble des transformations que vont subir les composants pour réaliser les produits finis de l'entreprise. L'objectif du processus Production est de fabriquer les produits requis tout en assurant la productivité du système (notamment par un taux élevé d'utilisation des ressources mobilisées).

Les méthodes utilisées pour la gestion de la production cherchent à améliorer le flux des produits dans les ateliers de fabrication à travers la planification et l'ordonnancement, la détermination de la taille optimale des lots de production, la détermination des séries économiques. Des approches multi-niveaux ont été développées dans le but de réduire la complexité des problèmes d'optimisation : planification multi-niveaux [18],[19] ordonnancement hiérarchisé [20],[21]. La principale difficulté dans l'exercice de la planification est la gestion de l'incertitude des informations [17]. Les données utilisées, telles que les plans de demandes, sont en effet issues de modèles de prévisions hypothétiques. De ce fait, la disponibilité des produits et des ressources de production en situation réelle, et par suite, le niveau de service ne sont pas toujours « au rendez-vous ».

3.3 Le processus Distribution

Le processus Distribution concerne la livraison des produits finis aux clients et reprend les questions d'optimisation des réseaux de distribution : l'organisation et le choix des moyens de transport, le choix du nombre d'étages (ou d'intermédiaires) dans le réseau de distribution ainsi que le positionnement des entrepôts et leur mode de gestion. Par exemple, les produits peuvent être acheminés en nombre par train et regroupés dans un entrepôt pour être livrés ensuite par camion aux clients d'une même zone géographique, en vue du meilleur compromis entre qualité de service et coût économique.

3.4 Le processus Vente

Le processus Vente, mis en œuvre par le service commercial, développe les relations envers le client (négociation des prix et des délais, enregistrement des commandes, ...) et par extension, recherche une meilleure connaissance du marché. Ce processus de l'entreprise est

également chargé de définir la demande prévisionnelle et d'intégrer des aspects commerciaux comme la durée de vie du produit pour anticiper l'évolution de ses ventes. Les aspects marketing (analyse de marché, publicité, promotions, ...) sont aussi gérés dans ce processus.

4 Les différents types de relations industrielles

Une entreprise ne fabrique pas l'intégralité des produits qu'elle consomme, elle doit en acheter ou en sous-traiter la production à d'autres entreprises.

La relation élémentaire entre deux entreprises correspond à une relation client/fournisseur dans le cas où une entreprise (le client) s'approvisionne en composants auprès d'une autre entreprise (le fournisseur).

Elle correspond à une relation donneur-d'ordres/sous-traitant, lorsqu'une entreprise définit les spécificités d'un composant qu'elle fait réaliser par une autre entreprise, soit par manque de ressource (sous-traitance de capacité), soit par manque de compétence (sous-traitance de spécialité).

Notre travail s'intéresse plus particulièrement aux relations de type client/fournisseur. Cependant, la relation de type donneur-d'ordres/sous-traitant dans le cas d'une sous-traitance de spécialité peut s'apparenter à une relation de type client/fournisseur. Car dans ces deux relations, la sollicitation entre partenaires est toujours unidirectionnelle, le donneur-d'ordres peut solliciter le sous-traitant par contre, l'inverse n'est pas possible.

5 Modèles de pilotage des chaînes logistiques

Pour aider les décideurs du SCM, des outils ont été développés par la recherche et transférés par l'offre logicielle. Il s'agit de la modélisation d'entreprise pour comprendre le positionnement des activités dans les processus d'entreprise.

5.1 La modélisation SCOR

On trouve également dans la littérature d'autres définitions, centrées sur une entreprise donnée [46], [47]. Ces définitions cherchent à caractériser la gestion de la chaîne logistique par une analyse des processus principaux qui vont cadrer leur fonctionnement. Dans ce cas, il ne s'agit plus d'aborder la chaîne logistique d'un produit donné, mais d'adopter le point de vue d'une entreprise qui doit gérer un ensemble de fournisseurs et de clients.

Dans cette optique, des modèles de chaîne logistique sont proposés. Le modèle le plus largement répandu pour répondre à cette définition orientée entreprise est le « *Supply Chain*

Operations Reference model » (SCOR) (Figure 1.10)[48], développé par le Supply Chain Council (SCC), organisation à but non lucratif rassemblant des entreprises et organisations désireuses d'apporter des avancées et d'appliquer l'état de l'art sur la gestion de la chaîne logistique et ses pratiques.

Le modèle SCOR a pour but de prendre en compte l'ensemble des interactions avec le client, toutes les transactions liées au produit et à la compréhension du marché grâce à une bonne interprétation de la demande agrégée jusqu'à la bonne réalisation des commandes. Pour cela, un ensemble de processus de référence est proposé, auxquels sont associés des indicateurs de performances permettant à l'entreprise de s'évaluer.

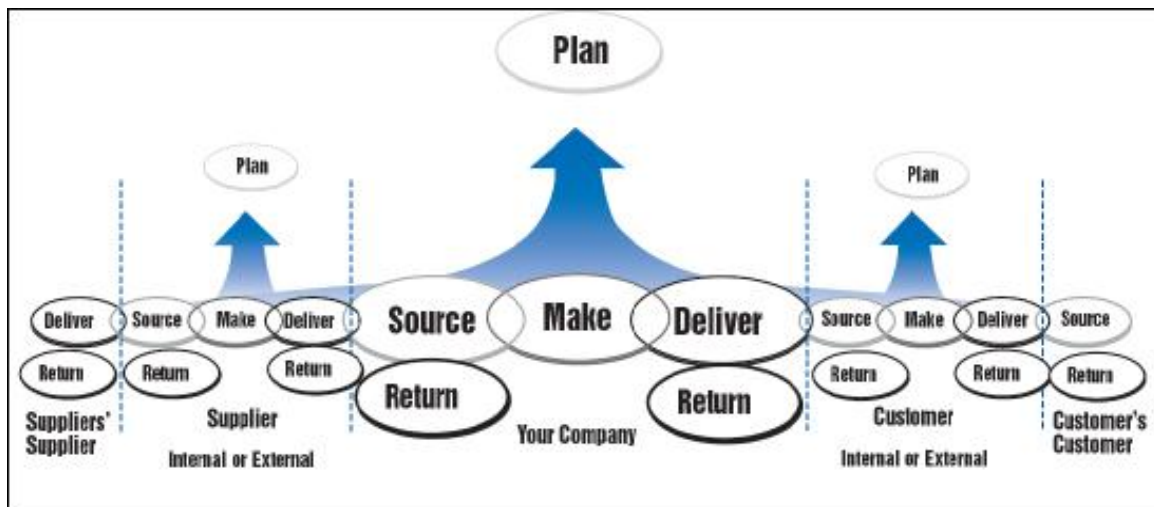


Figure 1.10 : le modèle SCOR [48]

Il faut noter que SCOR est spécifiquement établi pour les modèles de pilotage centralisés ou hiérarchisés sur des chaînes logistiques mono-entreprises. Ce modèle SCOR est donc peu adapté à la problématique du pilotage distribué que nous étudions dans ce mémoire.

5.2 Problème liée au pilotage des chaînes logistiques distribuées

Tous les modèles que nous avons présentés ci-dessus ont pour but d'améliorer les performances du management de la chaîne logistique. Cela dit, il nous semble important de mettre en lumière certains problèmes liés au fonctionnement de telles chaînes logistiques.

Dans sa thèse de doctorat, Latifa Ouzizi[49] reporte quatorze problèmes, pouvant apparaître, liés au fonctionnement même des chaînes logistiques et qui sont ; l'absence d'indicateurs de performance globaux de chaîne logistique, la définition insuffisante du service client, les données insuffisantes concernant la livraison, les systèmes d'information inefficaces, l'ignorance de l'impact des incertitudes, les politiques de stockage simplistes, la discrimination contre les clients internes, la coordination faible, l'analyse incomplète des moyens de transport, l'évaluation incorrecte des coûts de stock, la définition des limites des

organisations, la conception du Produit-Processus qui ne prend pas en considération la chaîne logistique, la séparation entre la conception de la chaîne logistique et des décisions opérationnelles et enfin la chaîne logistique incomplète. Tous ces problèmes sont toujours d'actualité et doivent être complétés par ce qui suit.

5.2.1 Problème de l'incertitude des informations et robustesse des décisions

Le problème de l'incertitude des informations au sein de la CL, concerne en particulier le traitement des commandes incertaines et imprécises. Nous appellerons 'commande incertaine' une commande dont l'occurrence n'est pas assurée et 'commande imprécise' une commande dont la quantité commandée n'est pas connue avec exactitude.

Le traitement des commandes incertaines, ou commandes imprécises, a été envisagé dans plusieurs travaux de recherche [26], [27], [28] et [29]. Mais le travail qui a englobé la majorité des études et a analysé simultanément l'imprécision et l'incertitude des informations a été proposé par [30]. Après avoir balayé l'apport des autres travaux, l'auteur propose une démarche similaire à MRP avec une approche basée sur la logique floue 'MRP flou'. Giannoccaro [31] a aussi utilisé la logique floue pour proposer une méthode de gestion de stock dans la chaîne logistique. Il a modélisé l'incertitude de la demande et les coûts du stock (coût de possession et coût des commandes en retard).

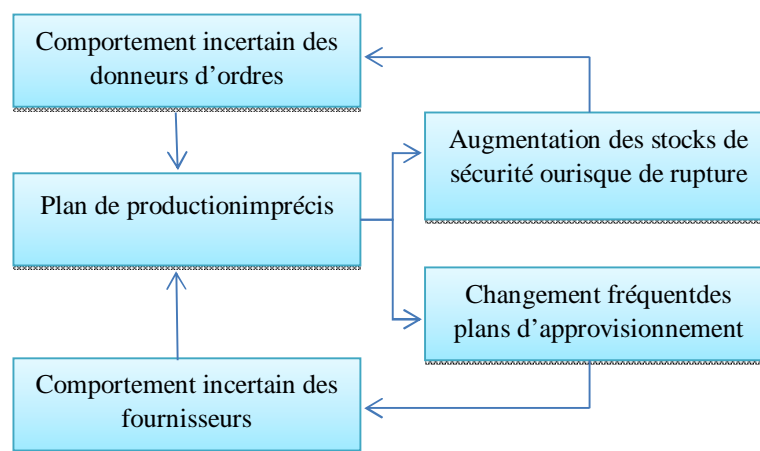


Figure 1.11 : Sources d'incertitudes dans la CL

Une des causes de la complexité des prises de décision est l'incertitude qui apparaît dans les fonctions d'approvisionnement, de production et de distribution. Différentes incertitudes, propres à chacune de ces fonctions, sont donc présentes dans la chaîne. Selon les travaux qui exposent cette problématique, la source la plus problématique concerne l'incertitude au niveau de la demande [40], [41] et [42]. Elle s'exprime en terme de quantité, de répartition entre produits, entre fournisseurs, en dates de besoin, etc. Si l'on considère des acteurs producteurs dans une CL, l'incertitude au niveau du processus de production est relative aux taux de pannes, de rebus, aux temps de répartition, etc. L'incertitude au niveau de la distribution se traduit par des retards à la livraison, des quantités supplémentaires livrées en

cas de mauvais transports, etc. La *figure 1.11* résume les sources d'incertitudes dans la CL [43].

L'une des solutions qui peut être apportée à ce genre de problèmes réside dans l'échange et le partage des informations dans le système. Ce qui permet à une entreprise de prédire, dans une certaine mesure, les effets potentiels de ces décisions sur son environnement.

Notre travail s'inscrit dans ce domaine de recherche des chaînes logistiques. L'enjeu principal des échanges d'informations est de permettre une meilleure cohérence dans les prises de décisions au niveau local de l'entreprise, mais aussi au niveau global, c'est-à-dire dans toute la chaîne logistique. La cohérence ainsi obtenue permet d'assurer la robustesse des décisions prises localement.

5.2.2 L'effet bullwhip (coup de fouet)

La performance de beaucoup de systèmes distribués est affectée par l'instabilité que confère la structure décisionnelle et informationnelle. Chaque entité de ce système agit de manière autonome et n'a aucune vision de l'impact de ses décisions sur le reste du système. Un des effets majeurs de la fluctuation des demandes d'un partenaire à un autre de la chaîne logistique est l'effet bullwhip [50].

La fluctuation du flux d'information, appelée aussi 'amplification de la demande' ou encore 'effet de Forrester' [53] reflète la déformation de l'information lorsque cette information remonte la CL du point de vente vers les fournisseurs de matières premières (*figure 1.12*). Concrètement, elle se propage de la façon suivante : le point de vente émet des commandes qui sont imprévisibles (pas forcément avec de fortes variations). Les conséquences négatives de ce phénomène sont l'accroissement des inventaires, la réduction du service aux clients due à des ruptures de stock, l'inefficacité des transports, l'échec des plans de production, etc.

La dynamique des flux d'information est difficile à cerner. Les entreprises sont autonomes et il n'y a pas de coordonnateur qui joue le rôle du centre de contrôle. Aussi, ces entreprises n'ont pas à transmettre l'intégralité de leurs informations.

L'effet coup de fouet est également la cause d'autres problèmes tels que les difficultés à ordonnancer, à gérer la main d'œuvre, à contrôler les stocks et les besoins en entrepôts qui en résultent, le service au client est faible (en particulier, en termes de retards des livraisons, de différences entre les quantités commandées et livrées et d'erreurs de facturation) ainsi que les efforts administratifs qui sont excessifs [52], [51].

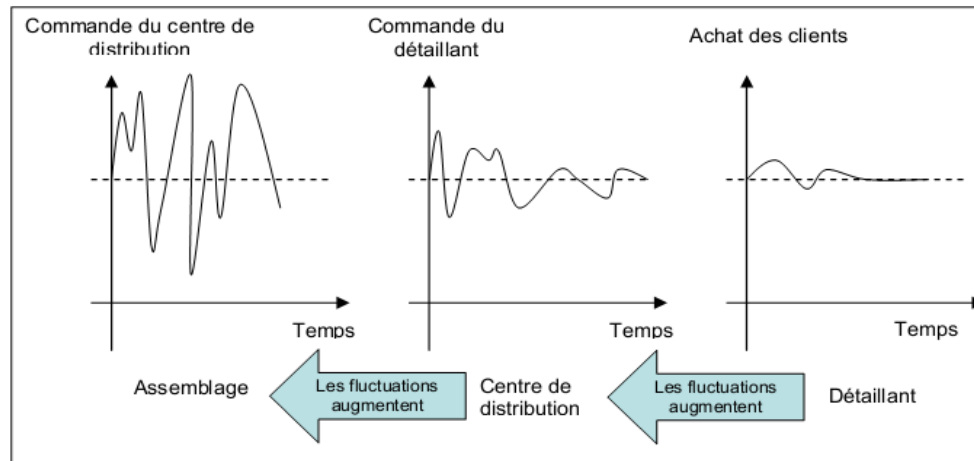


Figure 1.12: exemple de l'effet bullwhip [53].

6 Les instruments de pilotage de telles structures

La gestion de la chaîne logistique distribuée impose aux acteurs d'adopter une démarche coopérative. En effet, lorsque le pouvoir de décision est équitablement réparti, la coopération constitue un moyen efficace pour gérer de façon satisfaisante le processus de décision distribuée sous-jacent à la gestion de la chaîne logistique.

Selon [54], la coopération est le moyen de dépasser les limites de l'action individuelle. En ce sens, les entreprises optant pour un mode de fonctionnement coopératif attendent en retour une minimisation des risques et une réduction de l'incertitude, ainsi qu'un accroissement de la performance industrielle [55]. Plusieurs aspects composent la coopération interentreprises. Dans cette partie, nous nous intéressons particulièrement à trois aspects, en nous appuyant principalement sur les travaux de [56] et [57] : la coordination, la collaboration et la codécision.

- **La coordination** vise à synchroniser les actions dans le temps en exploitant un référentiel temporel commun, et à gérer la cohérence des actions individuelles par rapport à l'ensemble des activités. Selon [58], la coopération a pour objet de faciliter la coordination d'activités étroitement complémentaires pour la réalisation des processus.
- **La collaboration** signifie travailler ensemble à l'exécution d'une certaine action pour produire un résultat final. Selon [59], la collaboration implique le partage d'informations à l'intérieur d'un groupe donné, sans prise de décision collective. Le terme collaboration s'utilise à la place de coopération lorsque les actions individuelles ne sont pas différenciables.
- Si plusieurs acteurs collaborent en vue de prendre des décisions, on parlera de

codécision. Cette codécision peut être le résultat de mécanismes de négociation ou de renégociation. Ces mécanismes visent à trouver un compromis acceptable entre les objectifs locaux de chaque entité qui peuvent être contradictoires. Si aucune codécision n'a été prise au préalable, on parlera de négociation ; au contraire si l'objet de la collaboration entre partenaires est la remise en cause d'une décision passée et collective, on parlera de renégociation.

Dans notre approche, la coopération est assimilée à une action de prise de décision collective distribuée en vue de prise d'une commande ferme. Nous sommes donc dans un contexte de codécision que nous détaillons tout au long de ce mémoire.

7 Les outils d'aide à la décision dans les SCM

Nous avons vu dans les sections précédentes la complexité des organisations en chaînes logistiques. Celles-ci comprennent plusieurs partenaires et réalisent plusieurs fonctions. Nous avons vu aussi que le pilotage de telles structures est un processus complexe et difficile à mettre en œuvre. Tous les partenaires qui la composent coordonnent ou non leurs activités en vue de l'obtention du meilleur résultat pour le collectif ou individuellement. La taille importante d'un tel système, la diversité des données, et l'antagonisme des objectifs des différentes fonctions font sentir le besoin de mise en œuvre d'outils d'aide à la décision capables de prendre en compte tous ces paramètres.

Afin d'aborder le thème de l'aide à la décision dans le contexte de chaîne logistique, un retour sur la définition de la décision et sur la prise de décision s'impose.

7.1 La décision

Afin d'introduire les notions relatives à la décision, [142] propose un aperçu selon les trois axes :

- Selon [143], « La prise et l'exécution des décisions sont les buts fondamentaux de Toute organisation, de tout management. Toute organisation dépend, structurellement, de la nature des décisions qui sont prises en son sein et non par des décideurs, qu'ils soient individuels ou collectifs...etc ».
- Selon [144], une décision, qu'elle soit individuelle ou basé sur un travail de groupe, peut être définie comme « l'engagement dans une action, c'est-à-dire une intention explicite d'agir ».
- Selon [145], « Le but d'une décision est de résoudre un problème posé à une organisation ou à un individu.

Une décision peut donc s'interpréter comme le déclencheur d'un processus ayant pour objectif de répondre à un problème posé à une organisation. Cependant, la décision ne se résume pas à un simple choix à faire entre plusieurs alternatives. Pour la formuler, elle met en œuvre des mécanismes de prise de décision.

7.1.1 Prendre une décision

Simon.H[146] décrit un processus itératif qui découpe la décision en quatre activités fondamentales (*figure 1.13*) :

- L'identification, c'est-à-dire surveillé dans l'environnement les informations que l'on considère nécessaires à la décision.
- La conception, qui consiste à organiser et modéliser les informations retenues de façon à disposer des solutions possibles pour résoudre le problème.
- Le choix, c'est-à-dire effectuer une sélection parmi les solutions établies.
- L'évaluation, qui consiste à déterminer à l'issue du choix la qualité de la prise de décision.

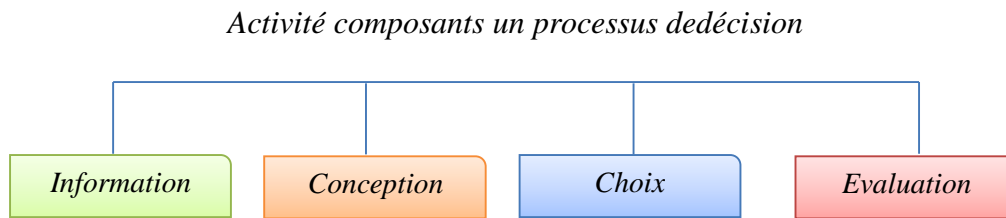


Figure 1.13 : Modèle de [146]

Toutefois, quel que soit le modèle considéré pour prendre une décision souvent difficile, il semble s'avérer nécessaire de disposer d'une quantité et d'une qualité d'informations adaptées, ainsi que le soutien d'une aide lors de la prise de décision.

7.1.2 L'aide à la décision

Selon Roy et Bouyssou[147], l'aide à la décision est définie comme étant l'activité de celui (homme d'étude) qui, prenant appui sur des modèles clairement explicites et plus ou moins complètement formalisés, cherche à obtenir des éléments de réponses aux questions que se pose un intervenant (décideur) dans un processus de décision, éléments concourant à éclairer la décision et normalement à prescrire un comportement de nature à accroître la cohérence entre l'évolution du processus d'une part, les objectifs et le système de valeurs au service duquel cet intervenant se trouve placé d'autre part.

Rapporté au domaine de la gestion de production, [148] conservent cette définition mais considèrent que l'aide à la décision est un terme impropre. Ils proposent de le décliner plus globalement en six fonctions en rapport aux activités élémentaires qu'ils proposent :

- L'aide au déclenchement du processus décisionnel ;
- L'aide au renseignement ;
- L'aide à la conception des alternatives ;
- L'aide à la décision ;
- L'aide à l'application de la décision ;
- L'aide à l'évaluation ;

7.1.3 Une définition des systèmes d'aide à la décision

Scott Morton a introduit pour la première fois, en 1971, l'aide à la décision en parlant de « système de décision et de gestion » couvrant le domaine qu'il définit comme suit[2] :

« [...] Ce travail, traitant de Décision et de Gestion, est basé sur une recherche qui combine à la fois des ordinateurs, des modèles analytiques et des terminaux utilisant des techniques de visualisation d'interactivité. Cette approche est basée sur l'analyse des décisions clés et fournit alors aux décideurs un support, une aide dans leur processus de prise de décision. Cette aide est possible dans des situations complexes et mal structurées et peut être utilisée par des décideurs en conjonction avec leur propre sentiment intuitif du problème et de sa solution. »

L'une des définitions des systèmes d'aide à la décision, qui fait référence, est celle de Keen et Morton [3] :

Les systèmes d'aide à la décision impliquent l'utilisation d'outils informatiques pour :

- assister les décideurs dans leur processus de décision dans des tâches semi-structurées
- aider plutôt que remplacer le jugement des décideurs ;
- améliorer la qualité de la prise de décision plutôt que l'efficacité.

Ainsi le concept d'aide à la décision est fondé sur l'équilibre entre le jugement humain et le traitement informatique. Elle constitue surtout une aide dans la formalisation d'un processus décisionnel et ne remplace en rien le décideur en vue de résoudre des problèmes complexes en s'appuyant sur des modèles et des données.

On peut remarquer que l'aide à la décision a repris le projet initial de la recherche opérationnelle : chercher à prendre appui sur la science pour éclairer les décisions de nature managériale et pour conduire les processus de décision dans les systèmes organisés[4].

7.2 Pilotage des chaines logistiques

Pour simplifier, Nous détaillons deux de ces systèmes, qui à notre sens sont les plus emblématiques : les Enterprise Resource Planning *ERP*, et les Advanced Planning and Scheduling *APS*.

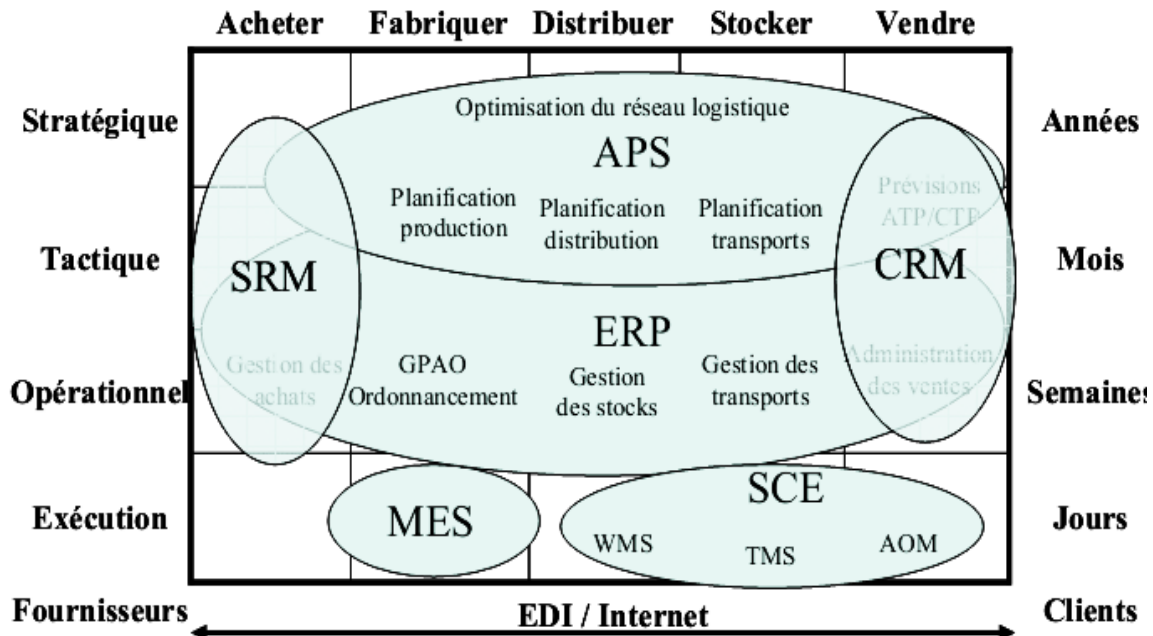


Figure 1.14 : L'offre logicielle du SCM[60]

Les *ERP* sont des outils transactionnels qui assurent la capture des données et leur stockage (commandes, gestion des stocks, gestion des achats, etc.). Les logiciels de type *APS* sont des outils spécialisés structurant la décision pour la planification de la demande, de la production et de la distribution. Ils optimisent la planification et synchronisent les flux de la chaîne logistique en tenant compte simultanément d'un grand nombre de contraintes (ressources, capacités, délais, coûts, profits, etc.) [149].

Les *ERP* se placent au niveau opérationnel en gérant les transactions. Les grands éditeurs d'*ERP* ont créé des progiciels qui couvrent littéralement toutes les fonctions de l'entreprise : gestion commerciale, gestion de production, achats, stocks, maintenance, qualité, comptabilités (générale, clients, fournisseurs), trésorerie, consolidation, gestion des ressources humaines, etc. Une étude récente sur les nouvelles recherches concernant les *ERP* est présentée dans [150].

Au-delà de toutes les fonctions de gestion interne de l'entreprise, les *ERP* offrent, grâce à une ouverture sur le monde extérieur au moyen des technologies de l'information, la possibilité de gérer efficacement l'ensemble de la chaîne logistique dans laquelle évolue l'entreprise. Ci-dessous une liste non exhaustive de fonctions :

- La planification inter-entreprise (Collaborative Planning) : permet de communiquer les plannings de fabrication et de livraison entre les partenaires (fournisseurs, clients, sous-traitants) pour aboutir à des solutions réalistes. Cela suppose des connexions entre les ERP des partenaires et donc d'avoir mis en place des accords de partenariat.
- La planification du réseau logistique (Supply Network Planning) : permet de faire correspondre la demande avec les processus d'achats, de fabrication et de transport, pour équilibrer et optimiser l'ensemble du réseau de logistique.
- Le pilotage du réseau logistique (Supply Chain Cockpit) : offre aux utilisateurs une vue générale de la chaîne logistique à l'aide d'une interface utilisateur graphique personnalisable.
- Le disponible à la vente globale (Global Available-to-Promise) : fait coïncider l'offre et la demande à une échelle vraiment internationale.
- Le e-procurement : permet de soumettre des appels d'offres via Internet sur des places de marché.

Un ERP permet normalement de réduire les ruptures de stock, d'abaisser le niveau moyen des stocks par une rotation plus élevée, d'améliorer le respect des délais de livraison promis aux clients et d'abaisser le coût de revient de la production par une meilleure régularité dans le fonctionnement des ateliers.

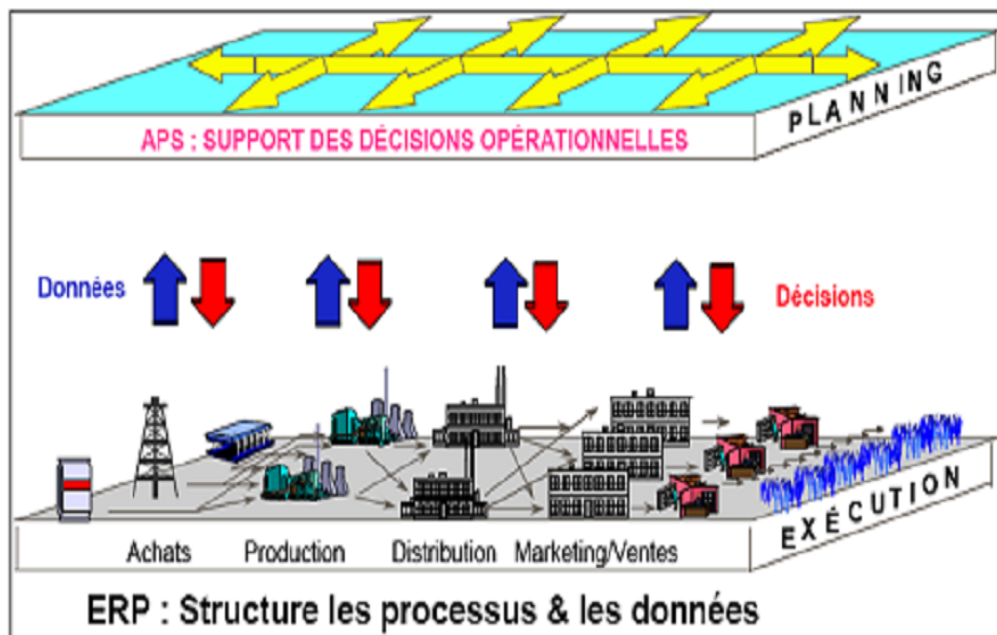


Figure 1.15 : Le positionnement relatif des ERP et des APS

Les données modélisées et traitées par l'APS sont issues de la base de données relationnelle de l'ERP. En effet, les APS ne disposent pas de leur propre base de données et doivent donc être « connectés » aux ERP ou au système d'information de l'entreprise.

Les APS interviennent pour la synchronisation de la chaîne logistique (*figure 1.15*) [151]. Les transactions liées aux déclarations de production, à la saisie d'une commande, etc. Seront exécutées dans l'ERP. Ces informations sont mises en forme dans l'APS pour que le gestionnaire puisse établir le plan. Les décisions de production, de transport, d'approvisionnement sont ensuite transmises à l'ERP sous forme d'ordres pour y être relancés.

Toutes les activités ayant un impact fort sur la qualité du service au client sont intégrées dans les APS : planification, remise de délai, gestion des stocks, expéditions, etc.

La cohabitation réussie entre l'APS et l'ERP s'avère essentielle. L'APS doit pouvoir déterminer les meilleures décisions en tenant compte des informations les plus récentes possibles contenues dans l'ERP, mais également assurer le suivi des plans pour contrôler que la situation ne dévie pas significativement. L'ERP doit recevoir les décisions de l'APS pour en tenir compte.

Les APS donnent une visibilité sur toute la chaîne et offrent des fonctionnalités d'interconnexion avec les systèmes d'information des clients et fournisseurs en s'appuyant sur le développement des technologies intranet et internet. Le gestionnaire de la chaîne logistique dispose donc aujourd'hui de l'architecture informatique qui lui permet de s'appuyer sur des données mises à jour à travers l'ERP pour prendre des décisions de planification avec le support de l'APS.

8 La problématique de notre étude

La problématique de la fidélisation et la satisfaction des clients en matière de rapidité de traitement des commandes et de livraison a suscité les intérêts d'un nombre important de chercheurs et supply chain managers. Plusieurs techniques et méthodes ainsi que des outils informatiques d'aide à la décision sont développées et mises en œuvre entre les acteurs d'une chaîne logistique en vue de coordonner leurs décisions et répondre ainsi à ce besoin. C'est dans le cadre de ces travaux que s'inscrit la principale contribution présentée dans ce mémoire.

Nous visons à proposer des solutions liées aux chaînes logistiques au moyen des agents cognitifs qui soient capables d'aider à la prise de décision, Ceci, en vue de permettre aux différentes entreprises d'avoir la capacité à satisfaire au mieux les demandes fermes des clients en termes de livraison, plus particulièrement, lors de la présence :

- des commandes imprévues : représentent l'écart entre la demande prévisionnelle et la demande initiale du client.
- des exceptions (problème de production, erreur sur prévisions, retard de livraison, etc.) pouvant altérer le bon déroulement d'un processus de fabrication, d'approvisionnement ou de livraison ce qui implique la non fidélisation de client.

Le but de notre recherche est de mettre en place les fondements d'une démarche globale de collaboration entre les partenaires existants dans la chaîne logistique distribuée. Cela en vue de la réalisation des opérations de prise de commandes fermes dans le contexte où les délais de réponses sont courts car nous focalisant sur la prise de décision au niveau opérationnel

8.1 Contexte de l'étude

Nous avons choisi de conduire notre étude sur La chaîne logistique qui se compose de plusieurs entreprises autonomes dans leurs prises de décision et représentées sous forme d'un graphe orienté sans circuits. Pour cette structure, nous développons les hypothèses de base à partir desquelles nous avons construit la modélisation également présentée par la suite.

La (Figure 1.16) illustre la structure de la chaîne logistique considérée. Cette chaîne est limitée à quatre acteurs :

- une entreprise productrice principale de produits finis,
- un ensemble des clients,
- des fournisseurs de produit fini,
- des fournisseurs de composants.

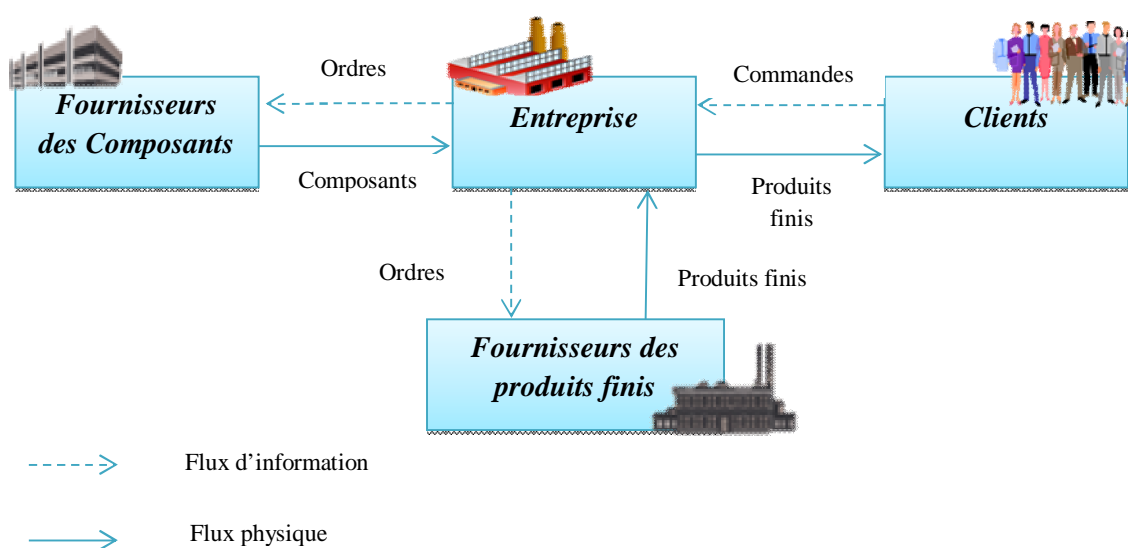


Figure 1.16 : Structure de la chaîne logistique étudiée

Nous nous plaçons dans le cas d'une entreprise qui assure la production d'une variété de produits finis en utilisant différents composants. Pour ce faire, elle gère un stock pour chaque produit fini ainsi qu'un stock pour chaque composant. Les capacités de stockage sont considérées infinies. La notion d'arriéré de production est prise en compte. Les arriérés sont assimilés à des retards dans la satisfaction des demande, Pour ajuster la capacité de production, l'entreprise a la possibilité de recourir à l'achat de produits finis.

8.2 Le but du travail

Le but principal de notre travail est de développer une architecture distribuée à base de système multi-agent (SMA) pour l'aide à la prise de décision dans le contexte de la chaîne logistique. Plus précisément nous présentons dans ce mémoire la problématique engendrée par la nécessité de satisfaire des commandes fermes.

L'architecture ainsi développée doit permettre, entre autres, de Synchroniser les décisions prises par les différents acteurs d'une chaîne logistique afin de pallier aux problèmes liés au flux d'information et flux de produits ; en particulier, lors de la présence des commandes fermes, des commandes imprévues ou des exceptions (problème de production, erreur sur prévisions, retard de livraison, etc.). Notre idée principale est de développer des agents cognitifs qui soient capables d'interagir et négocier afin de proposer des solutions aux managers pour leur aider à prendre une décision et cela dans un cadre méthodologique.

9 Conclusion

La gestion des chaînes logistique est un domaine de recherche très vaste qui traite un nombre important de problématiques décisionnelles touchant les trois niveaux de décisions : stratégique, tactique et opérationnel. Au cours de ce chapitre, nous avons donné une idée générale sur ces différentes problématiques.

Notre travail de mémoire souhaite être une contribution à la prise de décision dans une entreprise Pour qu'elle puisse satisfaire une demande client en cas de présence des événements imprévus,

Dans ce premier chapitre, nous avons présenté le thème général de notre travail, la conduite des relations interentreprises et plus particulièrement celle des relations entre Client et fournisseurs se constituant en architecture industrielle. Nous avons aussi posé le problème de la décision et notamment celui de sa distribution.

Chapitre 2

*Modèles D'aide A La Prise De Décision Pour Une
Demande Imprévue Dans Une Chaîne Logistique*

1 Introduction	45
2 L'entreprise	45
2.1 Niveaux décisionnels.....	45
2.1.1 Niveau stratégique.....	46
2.1.2 Niveau tactique.....	46
2.1.3 Niveau Opérationnel.....	46
3 Problématique de prise d'une commande	46
3.1 Mécanismes de prise de décisions.....	47
3.1.1 Acteurs de la prise de décisions.....	47
3.1.2 Processus de la prise de décision.....	47
3.2 Impacts des décisions opérationnelles dans le fonctionnement interne de l'entreprise.....	49
3.3 Impacts des décisions opérationnelles sur la chaîne logistique.....	49
4 Problèmes liés à la prise de commandes	50
5 Processus de prise de commande et présentation globale de notre approche	51
5.1 Mécanisme de traitement d'une nouvelle demande.....	52
5.2 Arrivée d'une nouvelle demande.....	53
5.3 Cas n°1 – Satisfaction de la demande en utilisant les produits disponibles dans les stocks	53
5.4 Cas n°2 et 3 – Satisfaction de la demande par production.....	54
5.5 Cas n°4 – Disponibilité de la matière première.....	56
5.6 Cas n°5 – Condition insuffisantes pour la réalisation de la demande.....	57
6 Problèmes de manque de composants	59
6.1 Positionnement du problème.....	59
6.2 Démarche de résolution.....	60
6.2.1 Création des scénarios.....	60
6.2.2 Recherche de fournisseurs.....	60
6.2.3 Envoi d'une demande aux fournisseurs.....	61
6.2.4 Réception des réponses.....	61
6.2.5 Sélection des propositions.....	61

<i>6.4 Modèles d'aide à la décision</i>	<i>62</i>
<i>6.3.1 Faisabilité d'un scénario d'approvisionnement</i>	<i>63</i>
<i>6.3.2 Modèle d'optimisation de la sélection.....</i>	<i>63</i>
<i>7 Problème d'insuffisance des conditions de production.....</i>	<i>66</i>
<i>7.2 Démarche de résolution.....</i>	<i>66</i>
<i>8 Synthèse de l'approche proposée.....</i>	<i>67</i>
<i>9 Conclusion.....</i>	<i>71</i>

1 Introduction

Nous détaillons dans le chapitre 2 la complexité ainsi que la problématique de prise de décision pour satisfaire une commande imprévue dans le contexte d'une chaîne logistique distribuée. Nous illustrons aussi l'organisation des activités nécessaires à cette prise de décision qui doit être réalisée dans un contexte où les délais de réponses sont réduits, c'est-à-dire au niveau de décision opérationnel.

Nous présentons aussi globalement l'approche de coopération que nous mettons en œuvre. A partir de l'analyse des critères et des conditions d'accomplissement d'une commande, nous nous concentrons essentiellement dans notre travail à la mise en œuvre de méthodes d'aide à la décision. La complexité de la prise de décision nous conduit au traitement de problèmes étroitement liés à la prise de commandes fermes qui divergent de la prévision et auxquelles l'entreprise ne peut répondre directement.

Dans la première partie de ce chapitre, nous présentons les niveaux décisionnels des entreprises actuelles cela permet de cadrer notre champ de travail et de formuler certaines hypothèses.

Une fois ce contexte de travail établi, dans la deuxième partie de ce chapitre, nous détaillons la problématique de prise de décision pour une commande ainsi que les enjeux et les impacts de telles décisions sur l'environnement dans lequel évolue une entreprise.

La dernière partie de ce chapitre présente globalement l'approche que nous proposons pour le traitement d'une commande ferme dans le cadre du contexte particulier de la chaîne logistique distribuée.

2 L'entreprise

Les entreprises constituent les maillons de la chaîne logistique et les centres de décision. Nous avons besoin de les caractériser à travers leurs interactions avec l'extérieur : les relations avec les clients et les fournisseurs et aussi à travers leur processus interne de production et de décision sur l'horizon opérationnel.

2.1 Niveaux décisionnels

Pour n'importe quel type de chaîne logistique, la prise de décision est divisée en trois niveaux: stratégique, tactique et opérationnel, correspondant respectivement à des horizons à long, moyen et court terme. Quelques problématiques et travaux rattachés à chacun des niveaux décisionnels sont présentés ci-dessous [22], [23], [24].

2.1.1 Niveau stratégique

Ce niveau, aussi appelé Strategic Management par [25] ou encore Strategic Planning par [45] regroupe toutes les décisions stratégiques. Ces décisions sont des directives et des lignes d'actions sur le long terme (de 6 mois à plusieurs années), comme, par exemple, la recherche de nouveaux partenaires industriels, la sélection des fournisseurs et sous-traitants, mais aussi les décisions d'implantation ou de délocalisation de zones d'intervention dans le cas de la logistique militaire, l'affectation d'une nouvelle zone d'approvisionnement à un centre de distribution (entrepôt), le développement d'un nouveau produit, la configuration de la chaîne logistique, son mode de fonctionnement, ainsi que les objectifs financiers à atteindre.

2.1.2 Niveau tactique

Le niveau décisionnel tactique s'intéresse aux décisions à moyen terme (de quelques semaines à quelques mois) qui devront être exécutées pour déployer la stratégie décidée par l'entreprise. Ces décisions portent sur les problèmes liés à la gestion des ressources de l'entreprise, en particulier la planification des activités en tenant compte des ressources disponibles sur un horizon fixé.

2.1.3 Niveau opérationnel

En ce qui concerne le niveau opérationnel, ou Operational Planning selon [45], les décisions ont une portée plus limitée dans l'espace et dans le temps (décisions sur la journée ou sur la semaine). A ce niveau, les décisions tactiques génèrent un plan détaillé de production ou d'ordonnancement, applicable au niveau d'un atelier ou d'une zone logistique.

Dans notre travail, nous nous plaçons à cheval entre le niveau tactique et le niveau opérationnel. Nous sommes au niveau opérationnel aux sens où nous réservons des ressources pour assurer la prise des commandes que nous acceptons. Mais nous sommes encore à un niveau agrégé : nous vérifions la faisabilité de la prise de commandes au niveau des plannings (de production, de transport...).

3 Problématique de prise d'une commande

Dans ce qui suit, nous mettons en lumière notre problématique de recherche qui est relative au pilotage du processus de la prise de commande imprévue qui se situe entre le niveau tactique et le niveau opérationnel.

Dans un premier temps, nous exposons un processus de prise de commandes fermes. Cela permet de spécifier l'ensemble des paramètres et des intervenants à prendre en compte dans ce genre de processus décisionnel. Dans le second volet, nous présentons quelques problèmes liés à ces processus décisionnels et comme dernière étape nous expliquons notre

démarche de résolution.

3.1 Mécanismes de prise de décisions

3.1.1 Acteurs de la prise de décisions

Les acteurs qui entrent en jeu dans la prise de décision dans le cadre d'une commande ferme sont multiples. Nous pouvons les classer en deux familles, la famille des acteurs internes et la famille des acteurs externes à une entreprise.

La famille des acteurs internes est composée des différentes structures (services) qui composent l'entreprise :

Classiquement, trois entités types sont présentées.

- L'acteur **distribution** gère aussi bien les demandes clients que la gestion des appels d'offres de l'entreprise (lors de la mise sur le marché d'un nouveau produit par exemple).
- L'acteur **fabrication** assure la mise à disposition des produits fabriqués par l'entreprise.
- L'acteur **approvisionnement** gère les besoins externes de l'entreprise qu'ils soient en matières physiques (composants ou matières premières) ou en services.

La seconde famille d'acteurs de la prise de décision se résume dans l'environnement extérieur de l'entreprise. Il est composé par l'ensemble des clients et des fournisseurs connectés à cette entreprise.

3.1.2 Processus de la prise de décision

La connaissance du processus de propagation du flux d'information dans le cadre d'une prise de décision est nécessaire. Cela nous permet d'avoir une vision précise sur la définition et le rôle de chacun des acteurs internes et externes qui participent à la prise de décision. Nous illustrons ainsi la complexité de ce processus décisionnel.

Dans le périmètre interne de l'entreprise, Lorsqu'une demande est émise par un service vers un autre, le mécanisme de prise de décision entre en jeu et son issue peut être conditionnée par d'autres décisions locales ou bien externes.

Dans la (figure 2.1)ci-dessous, nous donnons l'exemple d'un service d'approvisionnement qui est sollicité par un service de production afin de lui mettre à disposition les composants nécessaires au lancement de la production. La décision du

lancement de la production est donc conditionnée par la disponibilité des composants et donc la décision de confirmation du service des approvisionnements

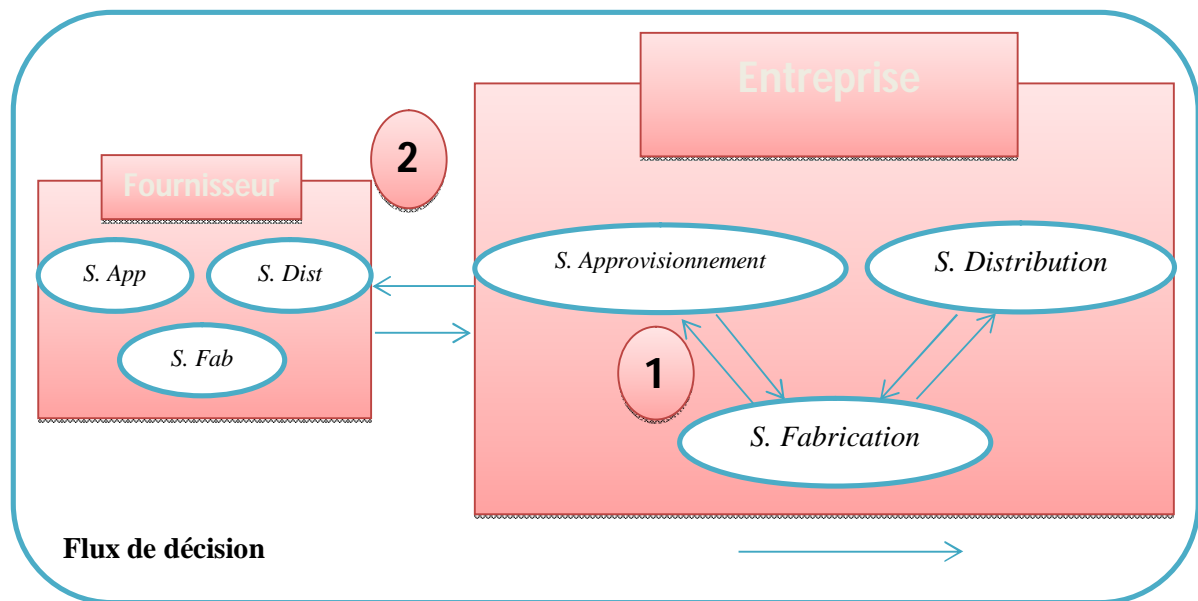


Figure 2.1 : Exemple de cheminement interne des décisions

Dans une vision externe à l'entreprise, nous pouvons exposer le cheminement suivant dans les prises de décision voire la (figure 2.2) :

Une prise de décision dans la vision externe d'une entreprise se déclenche généralement lorsqu'un client sollicite (1) celle-ci par le biais d'une demande de fourniture de produits. Cette demande peut elle-même générer des demandes secondaires (2).

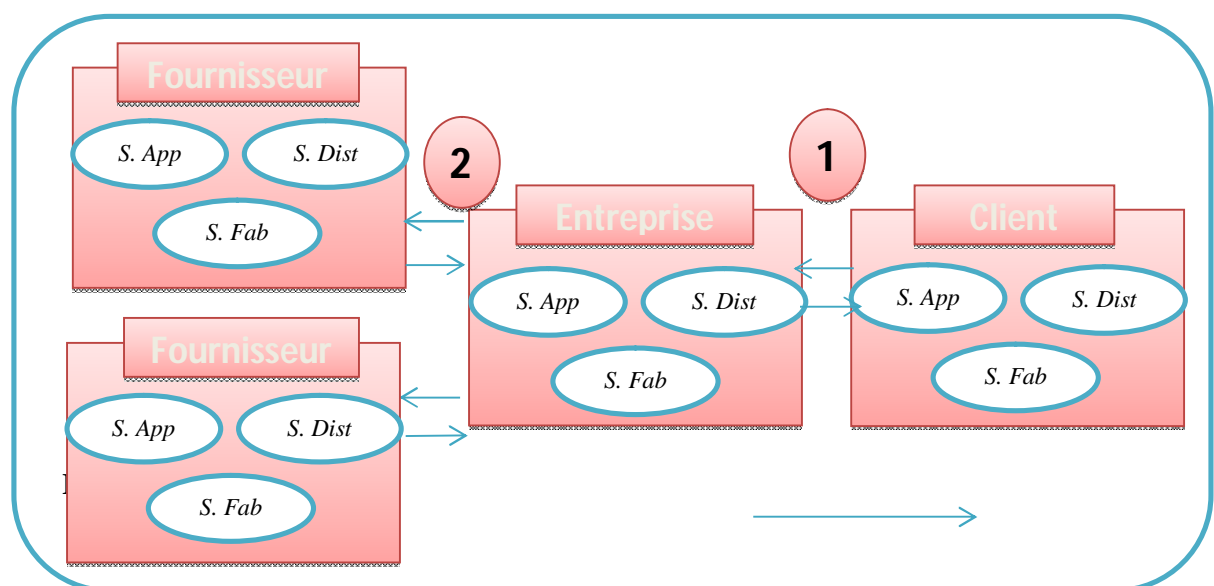


Figure 2.2 : propagation externe de la prise de décision

3.2 Impacts des décisions opérationnelles dans le fonctionnement interne de l'entreprise

Comme nous venons de le voir, la prise de décisions dans une entreprise est un phénomène complexe qui peut nécessiter l'intervention de plusieurs acteurs. L'impact d'une décision au sein de l'entreprise peut prendre des formes variées.

Nous pouvons les classer en deux grandes familles. Il s'agit de l'impact à court terme qui influe sur les ressources physiques dont dispose l'entreprise et de l'impact à long terme, lorsqu'une décision peut engendrer un changement dans la stratégie de fonctionnement de l'entreprise. Par exemple lors de la perte définitive d'un client important.

L'impact à court terme peut se produire par exemple lorsqu'une ressource est complètement affectée à la réalisation d'une décision dans une plage temporelle précise. La ressource ne peut pas donc être sollicitée jusqu'à la fin de la période dédiée à son exécution, car cela affectera la réalisation de la décision.

Pour l'impact à long terme, nous pouvons citer l'exemple d'une demande d'un client important qui ne peut pas être satisfaite. L'entreprise peut alors perdre ce client et donc avoir des conséquences déplorables à long terme sur son chiffre d'affaires qui peut engendrer par exemple des pertes de compétitivité et d'image de marque de l'entreprise.

3.3 Impacts des décisions opérationnelles sur la chaîne logistique (vision externe de l'entreprise)

L'impact des décisions prises en interne conduit généralement à l'occurrence de modifications dans l'environnement immédiat de l'entreprise (ses clients et ses fournisseurs directs). Mais, par le biais de la propagation des décisions, cet impact peut s'étendre à l'ensemble des partenaires de la chaîne logistique.

Par exemple, lorsque l'impact interne est sur les ressources libres les demandes des clients, à leur tour, peuvent ne pas être satisfaites si celles-ci ont besoin de la ressource pour les réaliser. Dans cette situation, ces clients n'ont d'autre choix que de trouver un autre fournisseur pour satisfaire leurs besoins initiaux ou d'y renoncer.

Nous voyons ainsi que le résultat d'une décision interne peut avoir un impact sur l'entourage de l'entreprise, en l'occurrence ses clients et ses fournisseurs.

De plus, lorsque l'entreprise se trouve dans une situation de surcharge de commande, c'est-à-dire, lorsqu'elle doit en traiter un qui dépasse ses capacités, la prise de décision relative au choix des commandes à satisfaire devient problématique et complexe.

4 Problèmes liés à la prise de commandes

Pour assurer une prise de commandes, une entreprise doit satisfaire plusieurs conditions qui entrent en jeu pour la prise de décisions. Ces conditions sont aussi bien liées à la nécessité du respect des différents paramètres de satisfaction de la demande qu'à leur impact sur l'environnement interne ou externe à l'entreprise.

Dans le cadre de ce mémoire, nous avons retenu les situations suivantes comme étant à prendre en compte dans la prise de commande imprévue :

➤ **Produit fini dans les stocks** : lorsqu'une entreprise reçoit une demande de fourniture d'un ou de plusieurs produits, la première action qu'entreprend l'entreprise est de vérifier si le ou les produits demandés sont en quantités suffisantes dans les stocks et non réservés pour un autre client. Dans le cas où cela est vérifié, l'entreprise peut alors satisfaire la demande sans autre impact dans son fonctionnement interne. Ce cas se produit généralement lorsque la demande est déjà planifiée et se réalise selon les prévisions. Cette situation est la plus courante. Elle n'engendre pas de problèmes de décisions particuliers et elle n'a aucun impact négatif sur la chaîne logistique. Si une partie des produits nécessaires ne sont pas disponibles, l'entreprise doit produire ces quantités manquantes. Il existe des méthodes de prévision ou de planification du stock (exemple constitution d'un stock de sécurité) qui permet à l'entreprise de palier à ce genre de problème, dans notre démarche nous supposons que ces techniques peuvent être insuffisantes pour satisfaire la demande initiale.

➤ **Capacité de production** : lorsque l'entreprise ne dispose pas de la quantité suffisante de produits pour répondre à une demande, elle doit produire dans les délais impartis une quantité de produits au moins égale à la quantité manquante pour satisfaire la demande. Il est donc nécessaire de s'assurer de la disponibilité des capacités de production aussi bien matérielles qu'humaines. Les capacités de production peuvent être internes à l'entreprise cible ou bien externes par l'utilisation de la sous-traitance.

➤ **Composantes ou matières premières dans les stocks** : lorsque l'entreprise veut s'engager dans un processus de production, elle doit s'assurer de la présence de composants ou des matières premières nécessaires à la fabrication du produit demandé. Dans le cas où ceci n'est pas vérifié, l'entreprise ne peut pas donc produire, elle peut s'approvisionner en conséquence auprès de ses fournisseurs.

Dans le contexte qui nous concerne, c'est-à-dire, la prise de commandes imprévues ou divergentes, les délais d'approvisionnement sont réduits et il est possible que l'entreprise ne trouve pas un fournisseur apte à lui fournir les quantités nécessaires pour la réalisation de sa production. Ce phénomène, bien que marginale, doit être bien maîtrisé, car il est source d'une dégradation des performances d'une chaîne logistique. Il est donc crucial d'y apporter des solutions innovantes et performantes.

- **Capacité de stockage** : c'est un paramètre majeur pour toute entreprise. Il est nécessaire de respecter toutes capacités de stockage dont dispose l'entreprise aussi bien pour les matières premières que pour les produits finis. Nous prenons l'hypothèse, dans ce mémoire, que pour chaque produit l'entreprise réserve une capacité de stockage.

Ainsi, lorsque la planification de ces activités est bien réalisée, la prise de commande ne nécessite pas de grande modification dans l'entreprise cible et dans son entourage immédiat. Par contre, lorsqu'un de ces facteurs de prise de décision est défaillant ou insuffisant en raison d'une mauvaise planification ou en raison de l'occurrence d'un aléa, l'entreprise peut perdre la demande de son client et peut même voir sa performance réduite. Nous considérons comme aléa tout évènement imprévu dont l'occurrence modifie l'état du système.

5 Processus de prise de commande et présentation globale de notre approche

L'objectif du travail que nous présentons dans ce mémoire est l'étude de la faisabilité de la prise d'une commande ferme qui diverge de la planification dans un contexte de chaîne logistique distribuée. Nous parlons alors de prise de commande imprévue.

Lorsqu'une entreprise reçoit d'un client une demande de produits, elle peut y répondre favorablement ou non.

Elle peut y répondre favorablement si :

- la demande ne modifie pas ou modifie de manière acceptable le planning de production de l'entreprise.
- l'entreprise est capable d'absorber la charge induite par cette demande. Pour cela, la modification des plannings de productions en découlant doit être sans impact dégradant les performances du système ni sur le plan local, ni sur le plan global.

L'entreprise ne peut que répondre défavorablement à une requête si elle ne peut la satisfaire sans la modifier. Dans ce cas, l'entreprise peut adopter deux comportements bien distincts :

- Refuser définitivement la demande.
- Entrer en négociations avec son client et lui proposer une alternative à sa demande initiale pour converger vers un consensus commun.

C'est dans la recherche de ce consensus que se situe le contexte de nos travaux. Ainsi, lorsque la commande ferme diverge par rapport aux volumes prévus, il est nécessaire de proposer des outils d'aide à la décision apte à coordonner les activités des partenaires. Pour cela, si la demande ne correspond pas à ce qui a été prévu entre l'entreprise et ses clients, l'entreprise doit intégrer de nouveaux paramètres dans sa prise de décision pour tenter de la satisfaire au mieux.

5.1 Mécanisme de traitement d'une nouvelle demande

Dans cette partie, nous nous focalisons sur le mécanisme de traitement d'une demande d'approvisionnement d'un produit qui arrive d'un client vers une entreprise. Le but de ce mécanisme est de permettre la prise de décision sur cette demande. La décision consiste soit à :

1. accepter la demande : répondre favorablement à la sollicitation du client en termes de quantité du produit, du respect des délais de livraison, mais aussi du prix proposé par l'entreprise.
2. refuser la demande : quand l'entreprise ne peut pas satisfaire la demande (exemple : le délai de réalisation ne permet pas de produire les quantités demandées) ou ne veut pas accepter la demande (exemple : le client est un mauvais payeur).
3. entrer en négociation : lorsqu'au moins une des composantes de la demande ne peut être satisfaite par l'entreprise, elle peut répondre par une contre-proposition à la demande. On entre alors dans une phase de négociation entre l'entreprise et son client. Cette phase peut s'achever soit par un compromis entre les deux protagonistes (acceptation d'une demande modifiée) ou par une divergence (la demande est définitivement refusée).

Afin d'avoir une vision claire des paramètres de prise de décision dans le cadre de notre étude, il nous semble nécessaire de faire une classification des contextes de prise de commandes ferme, cette classification est résumée dans le *tableau 2.1* suivant :

Dans le cas 1 : la décision de prise de commande ferme peut se faire sans l'intervention d'acteur extérieur à l'entreprise. Cela est dû à la satisfaction de toutes les flexibilités nécessaires à la prise de la commande.

Dans le cas 5 par exemple, la décision ne peut pas être locale, elle est conditionnée par la réussite des opérations d'acquisitions des flexibilités manquantes. Donc de la sollicitation de fournisseurs (de biens ou de services).

<i>Ressources</i> <i>Cas</i>	<i>Quantités produits finis</i>	<i>Capacité de production</i>	<i>Quantité composants</i>	<i>Capacité transport</i>	<i>Décision locale possible sans approvisionnement</i>
<i>Cas 1</i>	OUI	–	–	OUI	Possible
<i>Cas 2</i>	NON	OUI	OUI	OUI	Possible
<i>Cas 3</i>	NON	NON	OUI	OUI	Possible
<i>Cas 4</i>	NON	OK	NON	OUI	Impossible
<i>Cas 5</i>	NON	NON	NON	OUI	Impossible

OUI : Disponible pour la réalisation de la demande.

NON : capacités insuffisantes pour la réalisation de la demande.

Tableau 2.1 : Contexte de prise de commandes

5.2 Arrivée d'une nouvelle demande

Lorsqu'une demande est émise par un client vers une entreprise. Celle-ci est traitée par l'entreprise en vue de prendre une décision. Dans notre démarche, le traitement de cette demande consiste à transformer la demande en une courbe cumulée [152] qui intègre les données de la demande.

A partir d'un vecteur de flux de données, qui contient les informations de la demande, l'entreprise constitue une courbe cumulée. Le flux de données reçu se compose des informations suivantes :

1. La nature du produit commandé.
2. Les quantités souhaitées du produit.
3. Les délais de mise à disposition.
4. Le coût unitaire demandé.

Une fois l'ensemble des courbes cumulées constitué, l'analyse de la demande peut débuter et plusieurs processus de prise de décision sont à modéliser. Les différentes situations sont détaillées dans les parties suivantes.

5.3 Cas n°1 – Satisfaction de la demande en utilisant les produits disponibles dans les stocks

Une fois la demande formatée à l'aide de courbes cumulées, l'entreprise regarde dans son

stock de produits finis si les quantités demandées y sont disponibles ou sur le point d'arriver. Ces quantités sont nécessairement de types libres.

Si la disponibilité du produit est vérifiée dans les délais demandés, l'entreprise peut donc fournir celui-ci à son client. Les capacités de stocks correspondants à la demande passent de l'état libre vers l'état définitivement réservé.

Que se passe-t-il si cette condition n'est pas satisfaite ?

➤ Le niveau de stock de produits finis ne suffit pas pour satisfaire la demande : Dans ce cas, pour satisfaire la demande, l'entreprise doit se procurer la quantité manquante nécessaire du produit demandé par production. La capacité disponible dans le stock de produits finis devient de type réservé flexible et on examine la possibilité de compléter ce stock par production de la partie manquante.

5.4 Cas n°2 et 3– Satisfaction de la demande par production

Dans le cas où les quantités demandées par le client ne sont pas disponibles en totalité dans le stock de produits finis, l'entreprise doit produire les quantités manquantes pour satisfaire la demande.

Cette opération consiste en l'introduction de la charge induite par la fabrication des quantités manquantes dans le planning de production en respectant les quantités souhaitées ainsi que les délais de mise à disposition du produit fini demandé.

L'entreprise peut introduire la charge induite par la production des quantités manquantes si les capacités de production ainsi que les capacités de matières premières sont disponibles. Dans ce cas, l'entreprise réserve les capacités correspondantes et cela dans le respect des quantités à produire ainsi que les délais de mises à disposition.

Les capacités réservées sont définitivement introduites dans le planning de production et la commande devient donc ferme.

Si l'entreprise ne peut introduire en partie ou en totalité la charge de production induite par la demande du client. L'entreprise peut adopter deux stratégies afin de lui permettre d'obtenir une capacité de production nécessaire à l'achèvement de ce processus.

➤ ***stratégie de re-planification d'une ancienne demande***

Cette stratégie consiste à libérer des capacités de production afin de permettre le placement de la charge induite. Ceci est réalisé en déplaçant dans le planning des capacités préalablement alloué pour une autre demande afin de libérer les capacités nécessaires.

Cette opération peut s'effectuer uniquement dans le cas où ce déplacement ne modifie pas les conditions de la demande préalable. Cette opération peut consister à déplacer vers la

gauche, c'est-à-dire, à avancer les dates de mise en production de la demande préalable sous réserve de disponibilité des capacités autres que celles nécessaires pour la nouvelle demande (figure 2.3). Cette stratégie ne peut être utilisable que de manière locale dans la mesure où l'entreprise ne devrait pas modifier la nature des commandes préalablement acceptées.

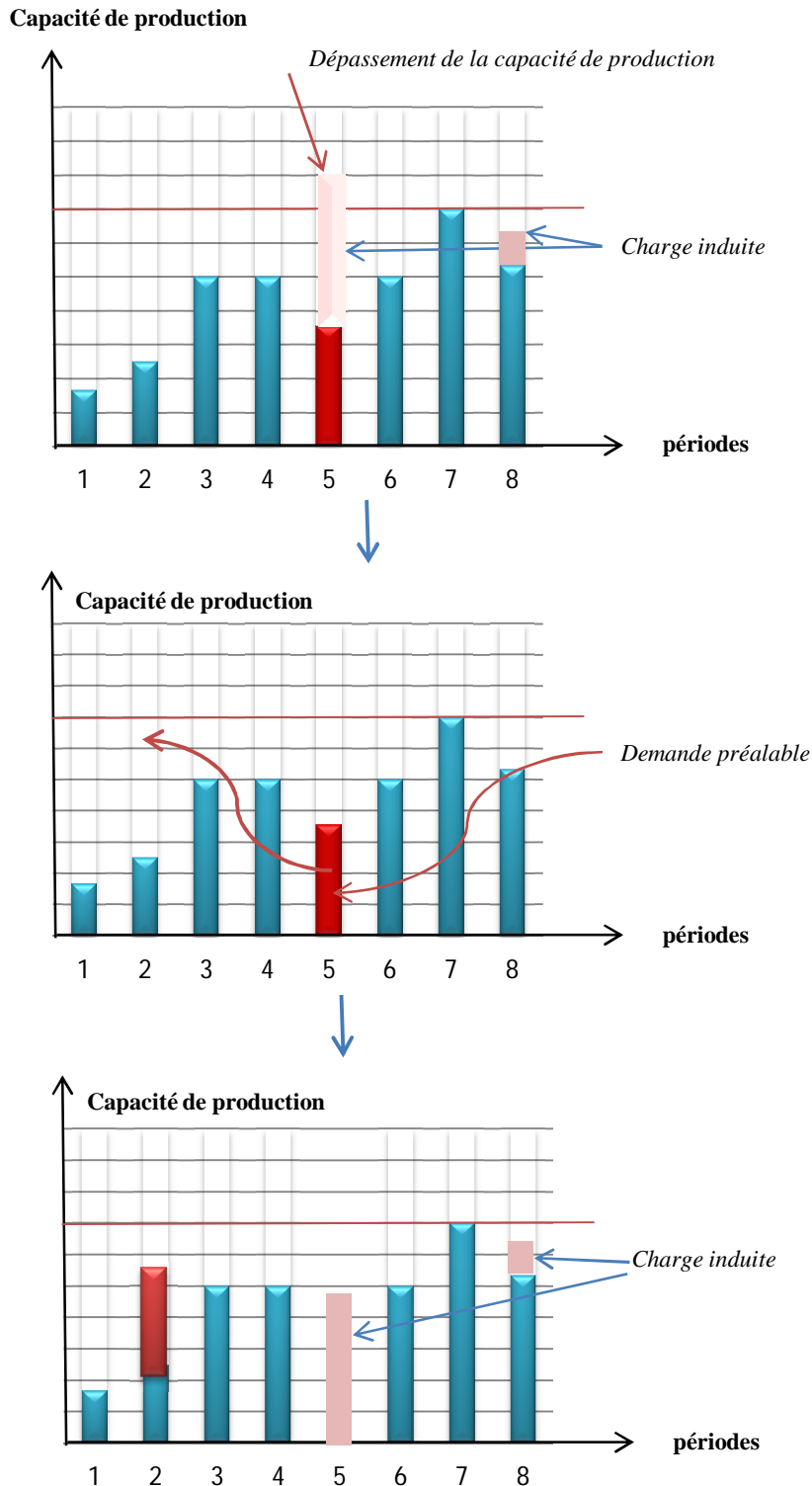


Figure 2.3 : Exemple de re-planification

➤ *stratégie de sous-traitance*

Dans ce cas, l'entreprise recherche des capacités de production ou de sous-traitants. Les capacités internes de production ne sont pas suffisantes afin d'assurer le respect de la demande et la procédure de re-planification ne donne pas de solution satisfaisante.

Une fois les sous-traitants potentiels trouvés, l'entreprise envoie des demandes (scénarios) de mises à disposition des capacités de production. Lorsque les sous-traitants répondent à la demande de l'entreprise. Les besoins en sous-traitance sont transformés en courbes cumulées de proposition de mise à disponibilité de moyens de production. Alors, une phase de sélection peut débuter qui peut aboutir à une solution qui combine de la production interne et de la sous-traitance

Cette solution consiste d'une part à réserver une quantité de produits finis dans le stock, mais aussi à produire le complément réalisable en interne et enfin à faire produire par sous-traitance le restant de la demande. Bien sûr, tout cela est conditionné par la disponibilité de capacités de transport et le respect des délais et des coûts.

5.5 Cas n°4 – Disponibilité de la matière première

C'est un point essentiel pour le bon déroulement de la phase de production. Dans le meilleur des cas, l'entreprise dispose d'une quantité suffisante de stock de matières premières pour répondre à la production. Dans le cas contraire, un problème de disponibilité des matières premières survient. Une formalisation de ce problème est proposée dans les sections suivantes de ce chapitre.

Pour pouvoir produire complètement les quantités souhaitées, l'entreprise doit rechercher des fournisseurs de matières premières pour compléter son stock de composants. Dans la mesure où les délais de production sont très réduits du fait du contexte de la demande, nous proposons une méthode de recherche et de sélection de fournisseurs afin de satisfaire au mieux le manque de matières premières. Les grandes étapes de cette procédure sont :

1. Création de la demande : à partir des quantités manquantes de composants, l'entreprise construit des scénarios d'approvisionnement.
2. Recherche de fournisseurs : l'entreprise dispose dans son carnet d'adresse d'un ensemble de fournisseurs, qu'ils soient habituels ou non.
3. L'envoi de la demande : après une phase de sélection de ces fournisseurs, elle transmet des demandes d'approvisionnements sous forme de scénarios (courbe cumulée de demande d'approvisionnement) aux fournisseurs sélectionnés. Ces demandes décrivent le manque en matières premières.

4. Réception des propositions d'approvisionnements : les propositions d'approvisionnements sont eux aussi sous forme de courbes cumulées. Nous supposons qu'un fournisseur peut envoyer une ou plusieurs propositions (scénarios). Toutefois si une de ces propositions est choisie, elle est prise en compte en totalité, et toutes les autres propositions de ce même fournisseur sont écartées.
5. Sélection des propositions : pour réaliser la sélection des propositions des fournisseurs, nous introduisons un modèle de sélection (*voir chapitre 2.4*). Cette opération a pour but de choisir la ou les meilleures propositions d'approvisionnement afin de satisfaire le manque de matières premières.

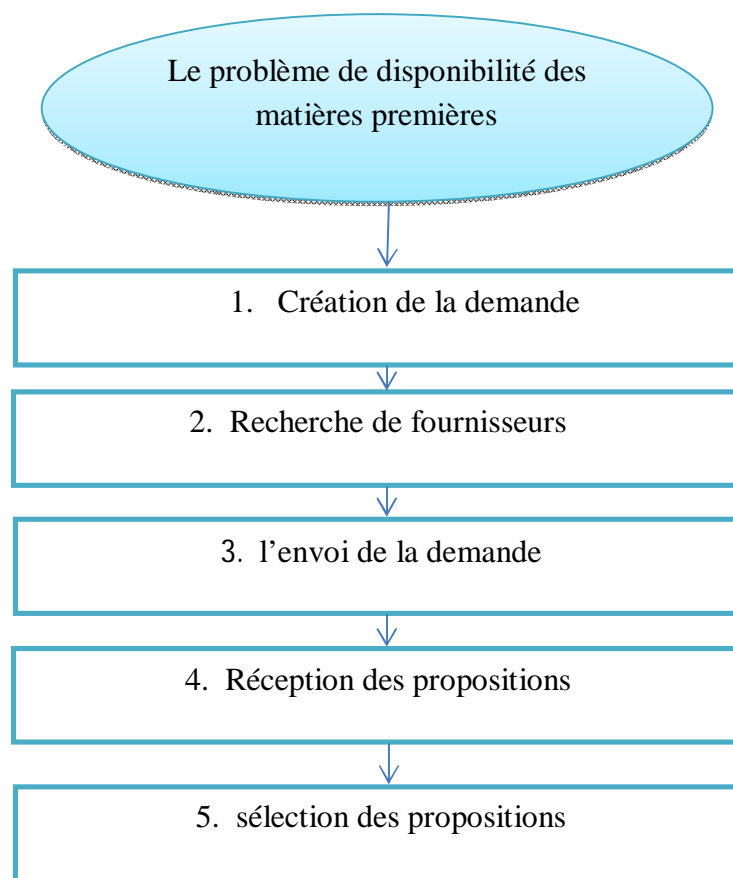


Figure 2.4 : Diagramme de l'approche pour la résolution du problème de disponibilité des matières premières

5.6 Cas n°5 – Condition insuffisantes pour la réalisation de la demande

La politique de l'entreprise vise en plus de gagner à aussi fidéliser le client car le fait de livrer des commandes imprévues et urgentes permet aux clients d'apprécier le service offert.

Le problème se résume alors à comment trouver la totalité ou une quantité de produit fini chez d'autre entreprise et livrer la commande imprévue tout en respectant la date de livraison et en minimisant le maximum possible les coûts de transport.

Dans le cadre du SC, notre architecture étend et automatise cette pratique en impliquant plusieurs entreprise de mêmes produits ou de produits équivalents dans une coopération interentreprises. Quand une commande urgente imprévue et ferme caractérisée par une quantité commandée et une date de livraison est passée par un client à une entreprise. A ce moment, nous proposons une méthode de recherche et de sélection de l'entreprise de produit fini afin de satisfaire au mieux le manque. Les grandes étapes de cette procédure sont :

1. Chercher la quantité de produit : elle peut être livrée totalement par une seule entreprise ou rassemblée par partie chez plusieurs entreprises.
2. Classer une série de parcours possibles par ordre croissant du coût de transport (qui dépend étroitement de la distance) pour transporter la quantité de produits commandée tout en respectant la date de livraison.
3. Proposer la quantité de produit fini à prendre chez chaque participant faisant partie d'un parcours.

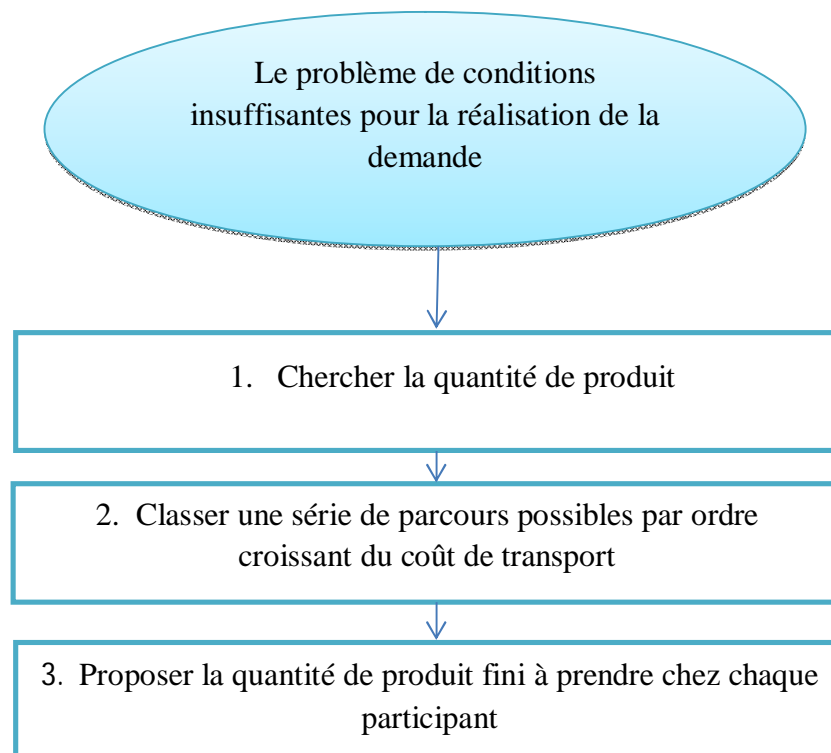


Figure 2.5 : Diagramme de l'approche pour la résolution du problème de recherche d'entreprise de produit fini.

6 problème de manque de composants

Le problème de manque de composants se produit lorsqu'une entreprise est confrontée à un manque de composants ou de matières premières nécessaires pour réaliser une quantité de produits finis relatifs à une demande imprévue de son client. Nous nous plaçons dans le contexte de *cas 4* présenté dans le *tableau 2.1* de la section précédente. L'entreprise ne dispose pas d'une quantité suffisante de produits finis et doit donc produire une partie de la demande du client. Pour cela, l'entreprise doit puiser dans son stock de matières premières. Le cas 4 exprime le fait que l'entreprise ne dispose pas des quantités de matières premières nécessaires à l'achèvement de cette production.

Dans cette première partie, nous nous intéressons donc à la recherche de fournisseurs de matières premières dans le but de répondre à la question suivante :

Comment faire pour trouver et sélectionner les fournisseurs pouvant résoudre le problème de déficience en matière première ?

Pour y répondre, nous développons dans ce qui suit une méthode de sélection de fournisseurs.

6.1 Positionnement du problème

Dans la littérature, durant les années 90, plusieurs entreprises cherchaient à collaborer avec leurs fournisseurs dans le but d'améliorer leurs performances et leur compétitivité [65], [66]. Le problème de sélection de fournisseurs devient un facteur très important dans l'établissement d'un système de réseau d'entreprises. Les principaux objectifs de cette démarche de sélection de fournisseurs sont de réduire les risques dus aux achats et de construire une relation durable et fiable entre l'acheteur et le fournisseur [68].

De nombreux modèles ont été développés pour la sélection de fournisseurs qui sont basés sur des perceptions plutôt simplistes des processus de prise de décision [67]. La plupart de ces méthodes ne semblent pas aborder les questions complexes et non structurées, contexte de la prise de décisions d'achat actuelle [67]. En fait, dans de nombreux modèles de décision, seul le critère de quantité est pris en considération pour le choix du fournisseur. Toutefois, plusieurs facteurs d'influence ne sont pas souvent pris en compte dans le processus de prise de décision, tels que des informations incomplètes, des critères qualitatifs et des critères de préférences...

À ce niveau de la thèse, nous nous appuyons sur les travaux de thèse de HanenBouchriha [62] pour classifier les différentes méthodes de sélection de fournisseurs. Il existe trois types de méthodes [61] ; les méthodes de pondération, les méthodes de programmation mathématique [64] (programmation linéaire, programmation mixte en nombre entier et la programmation mathématique multi-objectifs [156]) ainsi que des approches statistiques / probabilistes [63], à ces trois types s'ajoutent les méthodes de classement multi-

critères telle que la méthode AHP (Analytic Hierarchy process) [157].

Dans notre approche de sélection de fournisseurs, nous supposons que les relations sont déjà établies entre les partenaires de la chaîne logistique, nous sommes au niveau opérationnel de décision. Nous utilisons la programmation mixte en nombres entiers pour résoudre le problème de sélection de fournisseurs. Notre démarche s'inscrit dans une optique de satisfaction d'une demande imprévue.

6.2 Démarche de résolution

La démarche de résolution du problème de déficience en matière première repose sur un processus qui combine traitement de l'information et optimisation de la sélection des fournisseurs.

Ce processus s'appuie sur les activités suivantes :

1. création de scénarios d'approvisionnement.
2. recherche des fournisseurs potentiels.
3. envoi d'une demande aux fournisseurs potentiels.
4. réception de propositions d'approvisionnement.
5. sélection des propositions des fournisseurs.

Dans ce qui suit, nous détaillons les différentes étapes de ce processus.

6.2.1 Création des scénarios

Tout en respectant le planning de production préalablement établi, l'entreprise souhaite insérer une nouvelle charge qui correspond à une demande imprévue. Nous supposons qu'il existe différentes façons de réaliser cette insertion. Ce qui signifie qu'il existe plusieurs propositions de planning. Pour chaque planning de production, l'entreprise doit quantifier le profil des consommations en matières premières correspondant. Ceci permet de définir, pour chaque planning de production, un scénario d'approvisionnement différent.

Un scénario représente une demande d'approvisionnement relative à un planning de production qui permet de satisfaire la demande initiale du client. Il se présente sous forme d'une courbe cumulée de la quantité de la matière première demandée aux fournisseurs en fonction de délais de mise à disposition de celle-ci.

6.2.2 Recherche de fournisseurs

L'entreprise se lance dans la recherche de fournisseurs potentiels. Cette recherche consiste à trouver des fournisseurs qui peuvent approvisionner en composants ou en matières premières l'entreprise dans des délais courts. Ces fournisseurs figurent dans le carnet

d'adresses de l'entreprise, ils peuvent être habituels ou non. L'entreprise constitue donc une liste de fournisseurs à contacter.

6.2.3 Envoi d'une demande aux fournisseurs

L'entreprise contacte tous ses fournisseurs potentiels en leur envoyant une demande préalablement élaborée. Cette demande contient l'ensemble des scénarios qui intéressent l'entreprise.

6.2.4 Réception des réponses

Nous supposons dans cette étape que les fournisseurs font au plus une proposition pour chaque scénario. Cette hypothèse permet la réduction des flux d'information qui circule dans le réseau. Une fois que toutes les réponses sont collectées par l'entreprise ou après un certain temps (temps limite de réception des propositions) l'entreprise passe à l'étape de traitement des réponses.

6.2.5 Sélection des propositions

Une fois la réception des propositions terminée, la procédure de traitement peut commencer. Elle consiste à construire la meilleure solution réalisable pour l'approvisionnement des matières premières à partir d'un ensemble de propositions issues des fournisseurs

Pour cela, les réponses des fournisseurs sont transformées en courbes cumulées des propositions. Afin de vérifier si l'ensemble des propositions couvre un scénario d'approvisionnement, nous cumulons à leur tour les courbes cumulées des propositions qui correspondent à un scénario. En faisant la comparaison entre la courbe cumulée résultante et la courbe cumulée du scénario d'approvisionnement, nous pouvons savoir si ce scénario peut être satisfait ou non.

Si le scénario ne peut être satisfait, il est éliminé. Graphiquement, la courbe cumulée du scénario dépasse en au moins un point la courbe cumulée des propositions.

Dans le cas contraire, nous pouvons continuer par l'étape d'optimisation, c'est-à-dire, la sélection des meilleures propositions pour un scénario donné .

Dans le cas où il existe un scénario faisable, nous pouvons appliquer le modèle d'aide à la décision.

S'il existe au moins un scénario réalisable, la phase de sélection peut débuter. En s'appuyant sur les modèles d'aide à la décision que nous présentons dans la suite de ce chapitre.

Les résultats de l'analyse des sélections peuvent être les suivants :

Pas de solution réalisable : lorsque les propositions des fournisseurs ne couvrent aucun des scénarios d'approvisionnement, l'entreprise n'a pas d'ensemble de fournisseurs qui soient aptes à résoudre son problème de déficience en matière première. Cette situation induit que l'entreprise ne peut produire la quantité nécessaire à la satisfaction de la demande de son client.

1. Elle ne pourra accepter au mieux que partiellement une commande. Pour cela, elle peut entrer en phase de négociation avec son partenaire pour trouver une solution consensuelle. Cette recherche est assurée par le biais de contre-propositions.
2. solution unique : il y a possibilité d'avoir une solution unique lorsqu'une seule proposition d'un fournisseur couvre de manière optimale, au sens des coûts d'acquisition, la demande d'approvisionnement en matière première. Dans ce cas, l'entreprise peut entreprendre la mise en action de la production.

En effet, par cette démarche, l'entreprise s'assure seulement de la possibilité ou non de production des quantités nécessaires pour satisfaire la demande de son client.

3. solution composée : il est nécessaire d'élaborer une solution composée lorsque le résultat de la sélection nous donne un ensemble de propositions de fournisseurs où l'addition de celles-ci permet le recouvrement optimal de la demande d'approvisionnement. Alors, il convient de sélectionner un sous-ensemble de propositions couvrant le besoin de manière optimale au sens des coûts. C'est l'objectif de la section suivante.

6.3 Modèles d'aide à la décision

Les modèles que nous développons dans cette section s'appuient sur les hypothèses de notre étude, c'est-à-dire que l'ensemble des partenaires de la chaîne logistique joue le jeu gagnant-gagnant. Nous considérons aussi que, lorsqu'une proposition est sélectionnée, elle est entièrement prise en compte. Il n'y a donc pas de segmentation d'une proposition. Les paramètres pris en compte dans ces modèles sont relatifs aux capacités de stockage dont dispose l'entreprise, aux capacités de réception des composants ainsi qu'aux paramètres liés à la demande d'approvisionnement en composants.

Nous entamons cette section par l'analyse de faisabilité d'un scénario, celle-ci nous permet d'éliminer les scénarios non faisables. Ensuite, nous présentons le modèle de sélection des propositions (modèle π) pour les scénarios retenus dans l'étape précédente. Nous poursuivons par la présentation du modèle relaxé qui permet d'obtenir une solution au problème relaxé de sélection. Nous terminons cette section par une synthèse des modèles proposés.

6.3.1 Faisabilité d'un scénario d'approvisionnement

➤ Notations

- i : indice du fournisseur
- j : indice du scénario
- $Cumq_{i,j,t}$: quantité cumulée proposée par le fournisseur i pour le scénario j à la période t
- $CumQ_{j,t}$: quantité cumulée demandée pour le scénario j à la période t

➤ Equation de faisabilité d'un scénario

La faisabilité du scénario j est vérifiée par l'équation (E.1). On compare la courbe cumulée de l'addition de toutes les propositions relatives à un scénario d'approvisionnement avec le scénario correspondant.

$$\forall t: \sum_i Cumq_{i,j,t} \geq CumQ_{j,t} \quad (E1)$$

Si l'équation E.1 n'est pas satisfaite, le scénario j est écarté.

Dans le cas contraire, nous pouvons conclure que le scénario j est réalisable. On peut dans ce cas passer à l'étape d'optimisation.

6.3.2 Modèle d'optimisation de la sélection

L'optimisation consiste à la sélection des propositions des fournisseurs qui satisfont le problème de déficience en matière première. Cette opération s'appuie sur un modèle de programmation linéaire en variables mixtes (π). Ce modèle permet de minimiser le coût total d'acquisition de la matière première demandée (EQ.1) dans le respect des contraintes liées aux capacités internes de l'entreprise ainsi que les spécifications de la production (le respect des quantités et de délais imposés).

a) Notations complémentaires

- x_{ij} : $\left\{ \begin{array}{l} = 1 \text{ si la proposition relative au scénario } j \text{ est sélectionné} \\ = 0 \text{ sinon} \end{array} \right.$

- $Q_{t,j}$: quantité de matière première demandée à la période t pour le scénario j

- $q_{i,j,t}$: quantité de matière première proposée à la période t par le scénario j
- $c_{i,j}$: coût unitaire de la matière première proposé par le fournisseur i pour le scénario j
- $CF_{i,j}$: somme sur toutes les périodes t des quantités de matières premières proposées $q_{i,j,t}$ multipliées par les coûts $c_{i,j}$
- CD_j : coût unitaire d'approvisionnement demandé pour le scénario j
- QD_j : la quantité totale demandée pour le scénario j
- QR_t : capacité de réception potentielle de l'entreprise à la période t
- QS_t : capacité de stockage de la matière première potentielle à la période t
- ΔQR_t : capacité fixe additionnelle de réception à la période t
- ΔQS_t : capacité fixe additionnelle de stockage à la période t
- y_t : nombre des ΔQR_t
- z_t : nombre des ΔQS_t
- CR_t : coût d'acquisition de ΔQR_t
- CS_t : coût d'acquisition de ΔQS_t

b) Le modèle π

La fonction objective

$$Z_j = \text{Min} \left(\sum_i CF_{i,j} x_{i,j} \right) \quad (\text{EQ.1})$$

Sous les contraintes

$$\forall t : \sum_i \text{Cum} q_{i,j,t} x_{i,j} \geq \text{Cum} Q_{j,t} \quad (\text{E.2})$$

$$\forall t : \sum_{i,j,t} q_{i,j,t} x_{i,j} \leq QR_t \quad (\text{E.3})$$

$$\forall t : \sum_{i,j,t} q_{i,j,t} x_{i,j} \leq \text{Cum} Q_{j,t} + QS_t \quad (\text{E.4})$$

$$x_{i,j} \in \{0,1\}$$

(E2) représente la famille de contraintes de satisfaction des quantités demandées aux périodes souhaitées. Elle permet d'identifier l'ensemble des fournisseurs aptes à réaliser l'approvisionnement.

(E3) représente la famille des contraintes de capacité de réception, si l'entreprise ne dispose pas des capacités suffisantes cela induit au mieux un retard dans l'exécution de l'opération de livraison qui peut engendrer la non-validité du scénario correspondant.

(E4) est relative à la famille des contraintes de la capacité de stockage de l'entreprise. De manière analogue à (E3), le manque de capacité de stockage peut pénaliser la réalisation du scénario correspondant.

Dans notre travail, nous ne nous intéressons pas aux méthodes de résolution de tel programme de modélisation. Pour la résolution de ce programme π nous pouvons utiliser un outil de type solveur tel que [154] ou [155].

Si une solution existe, alors le solveur fournit une solution optimale pour le scénario correspondant. Dans ce cas l'entreprise dispose d'un ensemble de fournisseurs aptes à l'approvisionner. Le problème de déficience de matière première est donc résolu ce qui signifie que l'entreprise peut éventuellement lancer la production en vue de la réalisation de la demande de son client.

7 Problème d'insuffisance des conditions de production

Nous décrivons quelques processus et algorithmes permettant de gérer les commandes imprévues au sein d'une chaîne logistique de distribution de produits finis. L'arrivée d'une commande imprévue qui représente l'écart entre la demande prévisionnelle et la demande initiale du client, cause un problème à l'entreprise au sens de fidélisation de client car les conditions de réalisation de la production (capacité de production, la matière première, la capacité de transport...) peut être insuffisante, la démarche suivante représente une proposition pour satisfaire la commande client.

7.1 Démarche de résolution

Supposons qu'un client ait passé une commande urgente non prévue et ferme caractérisée par (Voir le Tableau 2.1 le cas 5) :

- une quantité commandée CQ (Ordered Quantity) ne pouvant pas être livrée totalement ou partiellement à partir du stock disponible,
- une date de livraison DL (Delivery Time) qui fixe directement la durée dont dispose l'entreprise pour livrer la commande.

Deux cas sont alors possibles :

1. L'entreprise ne dispose d'aucune partie de la quantité commandée CQ,
2. Elle dispose d'une partie et doit compléter le reste.

Dans les deux cas, on note : $CQ = DisQ + QR$ Avec :

- DisQ : quantité dont dispose l'entreprise, qui peut être nulle (cas 1) ou inférieur à OQ (cas 2).
- QR (required quantity): quantité recherchée.

Dans ce cas, nous nous intéressons donc à la recherche des fournisseurs des produits finis dans le but de répondre à la question suivante :

Trouver la quantité QR pour livrer la commande imprévue tout en respectant la durée TL et en minimisant le maximum possible les coûts de transport

L'une des pratiques utilisées par les entreprises et en particulier celle faisant objet de notre cas d'étude consiste à chercher la quantité RQ chez une autre entreprise qui fabrique le même produit fini ou un produit fini équivalent dans un cadre de coopération interentreprises.

Pour mettre en pratique ce processus de coopération entre les acteurs, satisfaire leurs exigences et répondre à leurs attentes, deux conditions doivent être vérifiées :

- La recherche de la solution doit être automatisée. De ce fait, le manager contacte les autres, juste pour confirmer la solution.
- Minimiser le maximum possible les coûts de transport qui seront ajoutés au coût d'achat déjà élevé puisque la commande ne sera pas livrée directement par l'entreprise.

Dans le cadre de notre approche les étapes de l'algorithme (*Figure 2.6*) pouvant être résumé comme suit :

- Chercher la quantité de produit RQ.
- Classer une série de parcours possibles par ordre croissant du coût de transport (qui dépend étroitement de la distance) pour transporter la quantité de produits CQ tout en respectant la durée TL.
- Proposer la quantité Q_j à prendre chez chaque participant faisant partie d'un parcours.

8 Synthèse de l'approche proposée

La démarche que nous entreprenons dans ce travail s'articule non seulement autour d'un processus de prise de décision, mais aussi sur la proposition d'outils d'aide à la décision. Comme montré dans la (*figure 2.7*), L'originalité de notre approche réside dans le fait de prendre en compte tout le processus de prise de décision dans le contexte distribué de la chaîne logistique.

La partie A de cette (*figure 2.7*) concerne la planification des activités de production, la partie B décrit la procédure de disponibilité des composants nécessaires pour satisfaire une demande. Enfin, la partie C décrit la procédure de disponibilité des produits finis cette dernière est considérée comme le dernier refuge ou solution pour satisfaire une demande imprévue.

Algorithme de recherche de la quantité manquante dans l'entreprise :**Constants**

- (dd_j) distances, (td_j) temps entre l'entreprise et les fournisseurs.

$dd_j = (dd_1, dd_2, \dots, dd_n)$ / n le nombre des entreprises des produits finit

$td_j = (td_1, td_2, \dots, td_n)$

Begin

// Déterminer la place des autres entreprises j fournisseurs candidate

If (fournisseurs appartient à la chaîne logistique) **Then**

Leurs distances et temps serrant définis (td, dd)

End If

Read TL // la date de livraison.

Read CQ // la quantité commandée.

Read DisQ // La quantité de produit fini disponible dans l'entreprise.

QR = CQ - DisQ // déduire la quantité manquante.

Envoyer un message à tous les entreprises ou clients qui appartiennent à la même chaîne logistique et leur demander la quantité maximum de produit fini qu'ils peuvent délivrer, ces derniers peuvent répondre favorablement en envoyant Q_{max} ou refuser.

Cree une liste vide **LC** // Construire une liste vide qui va contenir les

//entreprises ou clients qui disposent de même

//produit fini.

Read N // le nombre des clients ou entreprise appartenant à la liste LC

```
Do //  
  
  Cree une liste vide LCP // La liste des chemins possible  
  
  • A cette étape, en va réaliser la liste LCP des chemins possibles puis les classé selon le prix de transport, on doit aussi proposer la quantité  $Q_j$  à prendre depuis chaque client ou entreprise appartenant au chemin.  
  
  Chaque chemin possible est caractériser par :  
  
  • Chaque client ou entreprise vent répondre par l'envoi de la quantité  $Q_j$  qu'ils peuvent envoyer ou répondre par le refus de la proposition.  
  • Un point de départ.  
  • Une destination(Client).  
  • Elle contient N client ou entreprise, chacun appartient à la liste LCP et peut délivrer  $Q_j$ .  
  • Il doit satisfaire les conditions suivantes :  
  
  If (DisQ > 0) Then  
  
    • La somme de DisQ et des quantités  $Q_j$  doit être égale CQ.  
    • Le temps de parcours de chemin td doit être inférieur ou égale à la date de livraison TL.  
  
  Else  
  
    • La somme des  $Q_j$  doit être égale QR.  
    • Le temps de parcours de chemin TP doit être inférieur ou égale à la date de livraison TL.  
  
  End If  
  
  While(la liste LCP est vide)  
  
    // Fin de la boucle  
  
  End // Fin de l'algorithme de recherche
```

Figure 2.6 : Algorithme de recherche de la quantité.

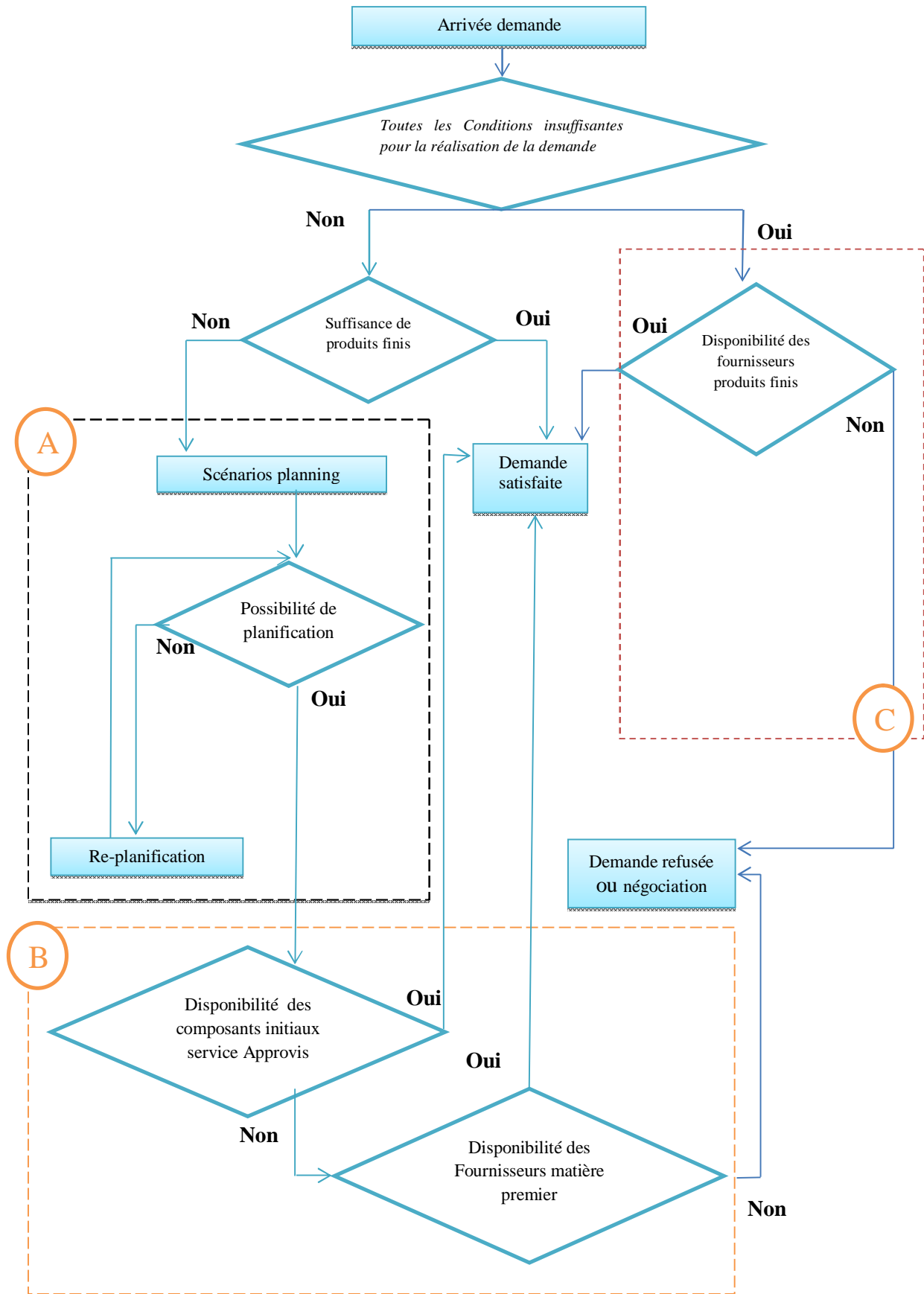


Figure 2.7 : Synthèse de l'approche globale

9 Conclusion

Lorsqu'une demande arrive dans un court délai et qu'elle n'est pas conforme à un planning préalablement établi, l'entreprise se trouve dans une situation délicate. Elle dispose de peu de temps pour s'organiser afin de répondre favorablement à la demande. Nous avons au cours de ce chapitre mis en lumière la complexité de cette prise de décision dans ce contexte où le temps alloué à une décision est réduit et où son impact tant local que global peut être majeur.

La démarche de résolution de notre problème s'appuie sur deux modèles de choix des fournisseurs. Le premier est formalisé par un programme linéaire en variable mixte. L'objectif est d'optimiser, en termes de coûts d'acquisition, la sélection de propositions de fournisseurs afin de permettre à une entreprise de produire dans le but de satisfaire une demande et ainsi de prendre en charge une commande non planifiée. Les critères de réalisation du modèle se basent sur la nécessité du recouvrement de la demande d'approvisionnement de l'entreprise. L'algorithme de sélection que nous proposons est l'élément central de notre démarche. Il s'agit d'un outil d'aide à la décision pour le choix des propositions des fournisseurs qui assure le bon déroulement de la production, le deuxième algorithme utilisé pour le choix des fournisseurs de produit finis pour satisfaire la demande client.

Dans le chapitre suivant, nous présentons notre approche de modélisation d'une entreprise partenaire d'une chaîne logistique distribuée. A cette fin, nous utilisons les techniques issues des systèmes multi agents (SMA).

Chapitre 3

*Systemes Multi-Agent : Une Approche Pour Les
Architectures Distribuées D'aide A La Décision*

2 Introduction.....	74
2 La notion d'agent.....	74
3 Les Systèmes Multi-Agents (SMA).....	75
3.3 Définition.....	75
3.4 Interactions.....	76
3.3 Langages de communication entre agents.....	77
3.4 Typologie des agents.....	78
4.4 Architectures d'intégration des agents.....	81
5 Applications des systèmes multi agents.....	82
5 Méthodologies de développement de SMA.....	86
6 Quelques Plates-formes de développement des SMA	90
7 Conclusion	93

1 Introduction

La conception, l'implémentation et la validation des systèmes multi-agents soulèvent de nombreuses questions de recherche qui leur sont spécifiques. Elles incluent la mise en œuvre de stratégies de coordination qui permettent effectivement à des groupes d'agents de résoudre des problèmes; de mécanismes de négociation qui permettent d'amener un ensemble d'agents dans un état global acceptable ; de techniques pour détecter les conflits et les résoudre; de protocoles par l'intermédiaire desquels les agents peuvent communiquer et raisonner sur leurs communications ; et enfin de mécanismes par lesquels les agents peuvent maintenir leur autonomie tout en contribuant au fonctionnement général du système.

Ce chapitre présente les systèmes multi-agents, les différentes sections proposées permettent d'introduire les notions d'agents, d'interactions entre les agents et des protocoles de communication entre les agents.

2 La notion d'agent

Durant ces dernières années nous avons assisté à un fort et rapide développement des recherches sur les agents et les systèmes multi-agents, vu qu'il n'existe pas de définition d'agent qui fasse l'unanimité dans la littérature. Il est donc intéressant de confronter différentes définitions afin d'avoir une bonne représentation du concept. La définition proposée par J. Ferber [69] met en évidence plusieurs caractéristiques.

On appelle agent une entité physique ou virtuelle

- qui est capable d'agir dans un environnement.
- qui peut communiquer directement avec d'autres agents.
- qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser).
- qui possède des ressources propres (on entend par ressource la somme des connaissances et des moyens qui sont mis à la disposition des agents afin qu'ils puissent agir dans son environnement).
- qui est capable de percevoir (mais de manière limitée) son environnement.
- qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune).
- qui possède des compétences et offre des services.
- qui peut éventuellement se reproduire.
- dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa

perception, de ses représentations et des communications qu'elle perçoit.

Pour Mickael Wooldridge[70], un agent est un système informatique capable d'agir de manière autonome et flexible dans un environnement.

Par flexibilité on entend :

- **Réactivité** : un système réactif maintient un lien constant avec son environnement et répond aux changements qui y surviennent.
- **Pro-activité** : un système pro-actif (aussi appelé téléonomique) génère et satisfait des buts. Son comportement n'est donc pas uniquement dirigé par des événements.
- **Capacités sociales** : un système social est capable d'interagir ou coopérer avec d'autres systèmes.

Nous retrouvons dans les deux définitions, les notions de réactivité, de pro-activité et une dimension sociale même si Ferber n'utilise pas ces termes-là.

En plus des propriétés citées ci-dessus, nous pouvons en identifier d'autres issues des définitions qui apparaissent dans la littérature, comme par exemple [71], [72], [73] [74] :

- **Raisonnement** : Un agent peut décider quel but poursuivre ou à quel événement réagir, comment agir pour accomplir un but, ou suspendre ou abandonner un but pour se dédier à un autre.
- **Apprentissage** : L'agent peut s'adapter progressivement à des changements dans des environnements dynamiques grâce à des techniques d'apprentissage.
- **Mobilité**: Dans des applications déterminées il peut être intéressant de permettre aux agents de migrer d'un nœud à un autre dans un réseau tout en préservant leur état lors de sauts entre nœuds.

3 Les Systèmes Multi-Agents (SMA)

3.1 Définition

Parmi les différentes définitions des systèmes multi-agents nous retiendrons celle de Ferber[69] qui le définit comme un système composé des éléments suivants (*Figure 3.1*):

Un environnement **E**, c'est à dire un espace disposant généralement d'une métrique.

- a) 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.
- b) Un ensemble **A** d'agents, qui sont des objets particuliers représentant les entités actives du système.
- c) Un ensemble de relations **R** qui unissent des objets (et donc des agents) entre eux.

- d) Un ensemble d'opérations **Op** permettant aux agents de **A** de percevoir, produire, consommer, transformer et manipuler des objets de **O**.
- e) 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."

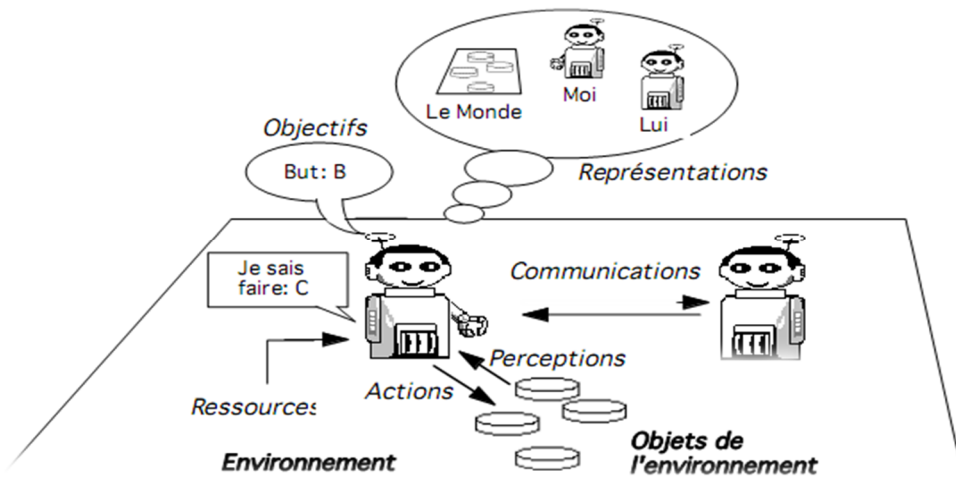


Figure 3.1: Représentation imagée d'un agent en interaction avec son environnement et les autres agents

3.2 Interactions

L'interaction est un support d'actions collectives permettant d'assurer la communication entre des agents. L'interaction entre agents permet d'influencer leurs prises de décisions par des états de coopération ou de négociation [75].

J. Ferber [69] propose la définition suivante de l'interaction : « Une interaction est une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques ».

La communication est une forme particulière d'interaction, qui tend à modifier l'état mental de l'agent destinataire et éventuellement de l'agent émetteur. Les situations d'interactions entre agents sont nombreuses et diverses : coopération, coordination, échange d'informations, résolution de conflits, etc.

La coopération consiste à répartir le travail entre plusieurs agents pour la mise en œuvre commune. Alors que la coordination s'intéresse à la manière dont les actions des agents sont organisées dans le temps et dans l'espace pour accomplir les buts. On distingue quatre formes de coordination d'actions [69] :

- *La coordination par synchronisation* : il s'agit de la forme la plus basique pour laquelle les actions sont décrites précisément au niveau de leur enchaînement.

- *La coordination par planification* : cette technique repose sur un découpage de l'action en deux phases. La première consiste à créer un ensemble de plans d'actions qui décrivent précisément les actions à effectuer pour atteindre un but. La seconde est l'exécution de l'un de ces plans.
- *La coordination réactive* : cette technique considère qu'il est plus facile de mettre en œuvre des mécanismes de coordination fondés sur des agents réactifs que de planifier l'ensemble des actions.
- *La coordination par réglementation* : consiste à ordonner les règles de comportement qui visent à éliminer les conflits potentiels entre agents.

3.3 Langages de communication entre agents

Grâce à la coordination, un système multi-agent peut réaliser ses tâches avec plus d'efficacité qu'un seul agent. Mais pour coordonner l'activité d'un ensemble hétérogène d'agents autonomes, il faut que les agents communiquent dans un langage compréhensible par tous les autres.

L'utilisation d'un langage de communication commun implique que tous les agents comprennent son vocabulaire sous tous ses aspects concernant:

- la syntaxe, qui précise le mode de structuration des symboles.
- la pragmatique pour pouvoir interpréter les symboles.
- l'ontologie pour pouvoir utiliser les mêmes mots d'un vocabulaire commun.

KQML (*Knowledge Query and Manipulation Language*) et FIPA-ACL (*Foundation for Intelligent Physical Agents – Agent Communication Language*) [76] sont deux principaux langages de communication entre agents. Ils sont presque identiques en ce qui concerne leurs concepts de base et les principes qu'ils observent. Ils diffèrent principalement dans les détails de leurs cadres sémantiques. Dans un sens, cette différence est substantielle :

- il n'est pas possible de proposer une traduction systématique entre les performatives de KQML et celles complètement équivalentes de FIPA, ou vice-versa.
- les différences inéluctables pourraient avoir peu d'importance pour les programmeurs d'agents intelligents, si leurs agents ne sont pas de véritables agents BDI (Belief – Desire – Intention), en Français "Croyance – Désir – Intention".

Chaque message KQML ou FIPA-ACL comprend plusieurs éléments. Voici quelques-uns :

- Performative : Type de l'acte communicatif (passage d'information, réquisition d'information...).
- Sender : L'émetteur du message.
- Receiver : Le destinataire du message.
- Content : Le contenu du message (l'information transportée par la performative).

Les deux langages n'impliquent aucun engagement de base vis-à-vis d'un langage pour le contenu. Plusieurs langages peuvent être utilisés pour coder le contenu, comme :

- Le langage KIF (Knowledge Interchange Format).
- SQL, PROLOG, STEP.

3.4 Typologie des agents

Il existe plusieurs types d'agents qui caractérisent la complexité de l'infrastructure à modéliser. Une classification détaillée des types d'agents est proposée par [139] :

- Agent à simple réflexe : ce type d'agent se comporte de manière simple. Il réagit de façon prédéterminée à une situation donnée de son environnement sans avoir une vision des actions précédentes. Cela est dû à l'absence de mémoire ou d'historique. Ce type d'agent est considéré comme simple.

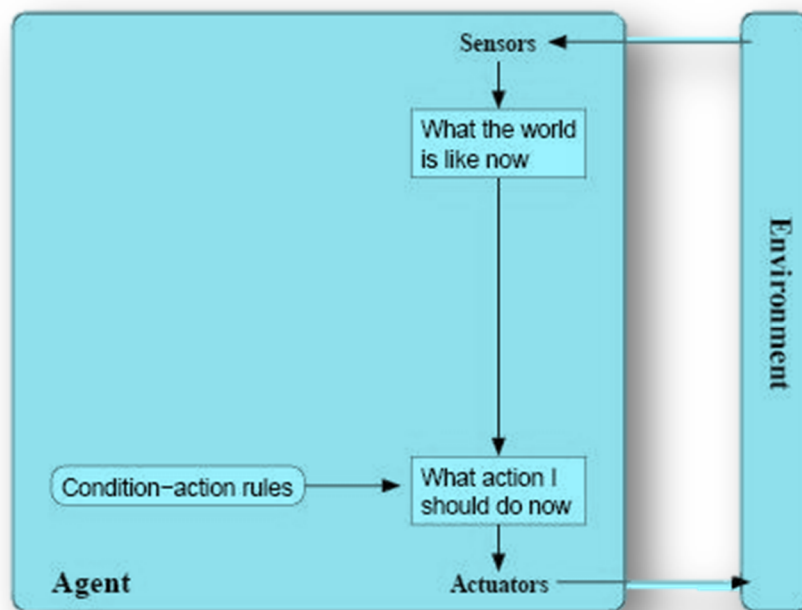


Figure 3.2 : Composition d'un agent simple réflexe

- Agent à modèle basé sur le réflexe : il ne perçoit pas l'ensemble de son environnement. Pour pallier cela, l'agent est complété d'un modèle général qui représente l'évolution théorique de son environnement. Il réagit de la même façon que l'agent à simple réflexe. Cependant, pour mettre en œuvre ses actions, il se base sur la vision que lui offre son modèle.

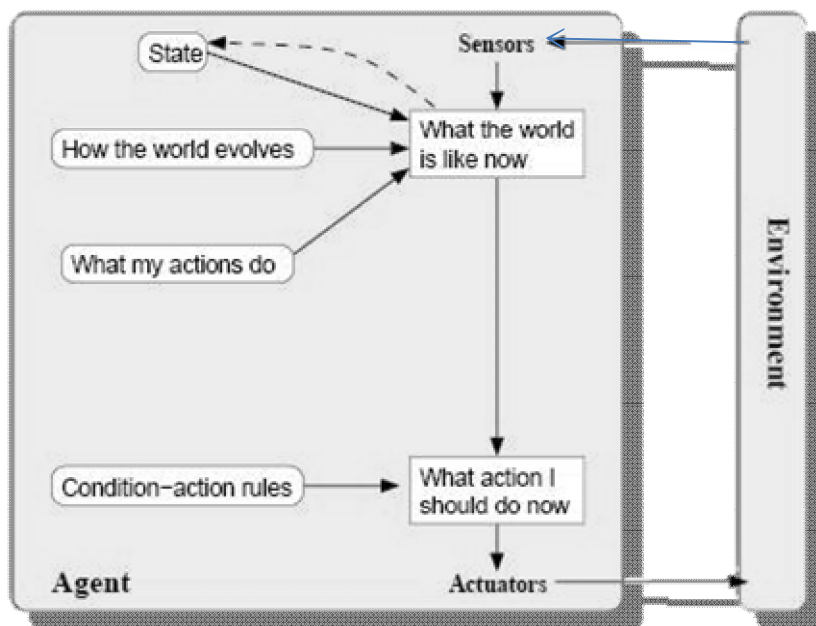


Figure 3.3 : composition d'un Agent à modèle basé sur le réflexe

- Agent basé sur le but : ce type d'agent a pour but la description d'informations décrivant les situations souhaitées, parce que l'état actuel du modèle de l'environnement ne suffit pas toujours à sélectionner une action efficace. Les décisions prises par ce type d'agent sont plus ou moins à longues périodes d'utilisation.

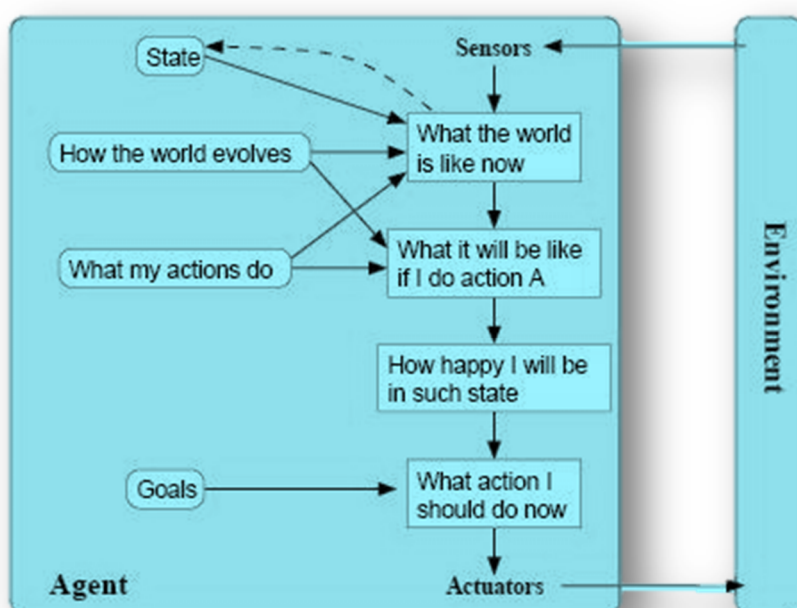


Figure 3.4 : Composition d'un agent basé sur le but

- Agent basé sur le service : ce type d'agent est basé sur une fonction de service qui s'inscrit dans une partie du modèle de l'environnement. Ceci afin d'améliorer le comportement de cet agent.

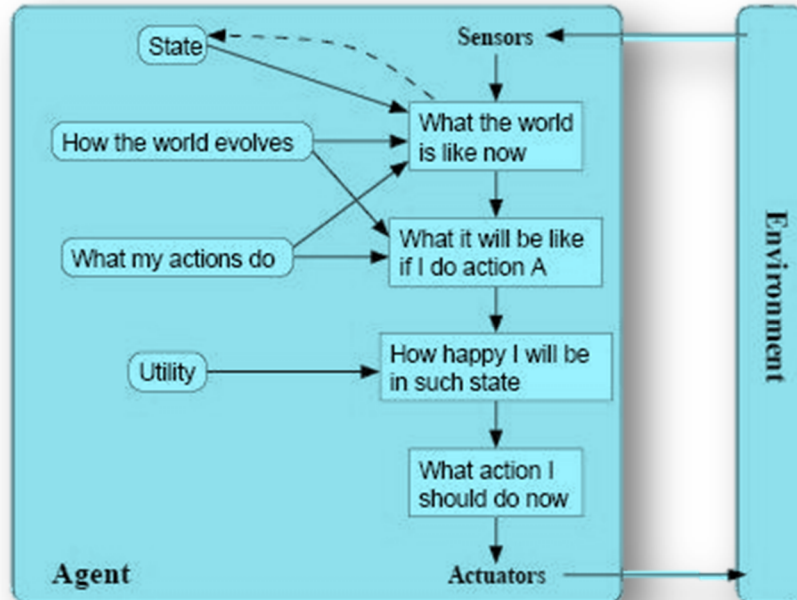


Figure 3.5 : Composition d'un agent basé sur le service.

- Agent basé sur la connaissance (agent cognitif) : ce type d'agent est généralement utilisé pour la collecte des informations liées à l'environnement. C'est un agent de préservation de la connaissance qui sert à son apprentissage, il se base sur cela pour réaliser ses actions. L'avantage de ce type d'agent est l'adaptabilité à n'importe quelle structure.

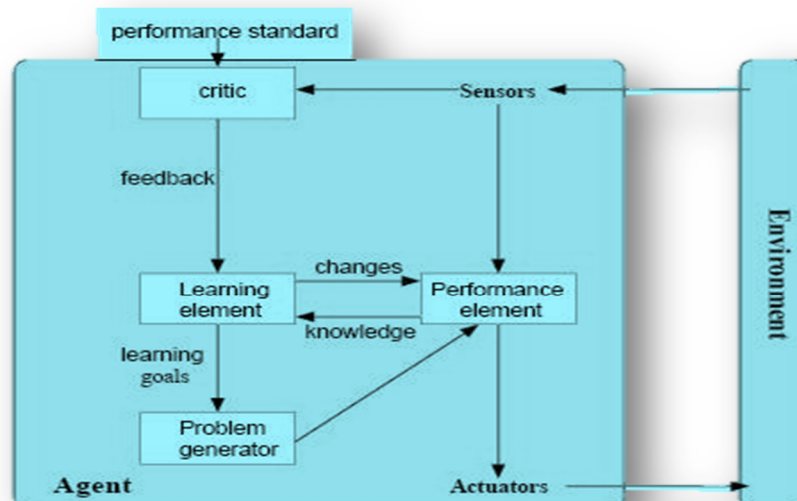


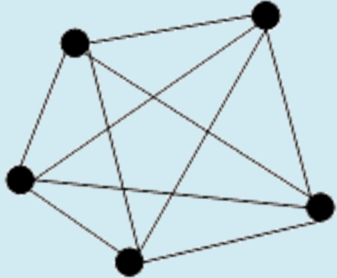
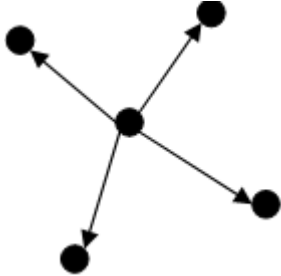
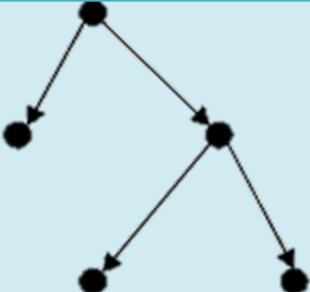
Figure 3.6 : Composition d'un agent basé sur la connaissance.

Notre choix de typologie d'agent pour la modélisation de l'architecture se porte sur les agents cognitif, car comme nous allons le voir tout au long de ce chapitre nous disposons d'une base de connaissance et d'un moteur de raisonnement. Les agents sont utilisés pour la prise de décision dans une situation imprévue.

3.5 Architectures d'intégration des agents

Nous avons présenté dans le chapitre 1, les différentes architectures des entreprises en réseaux (voir le tableau 3.1). L'intégration des systèmes multi agents utilise aussi ces types d'architectures afin de calquer le comportement de ces entreprises en réseaux.

Nous présentons ici six types d'architectures d'intégration dans le tableau 3.1.

Architecture	Type	Figure
Complète	Tous les agents qui la composent sont connectés par des flux d'informations. Une représentation simple de ce type est un graphe complet. Dans ce type d'architecture, tous les agents travaillent en commun et ont une vision totale de leur environnement.	
architectures centralisées	La structure est gérée par un agent Unique de coordination. Les agents sont connectés dans ce cas par des flux de coordination émanant de l'agent de coordination. Nous pouvons l'illustrer par un arbre orienté qui dispose d'un sommet source.	
architectures hiérarchiques	Chaque agent ou une partie d'entre eux est contrôlé par un coordinateur (un agent de coordination) qui supervise la bonne marche du système. On se rapproche d'une structure centralisée. Pour représenter ce genre d'intégration, nous pouvons utiliser un arbre orienté où chaque arc représente un flux de coordination	

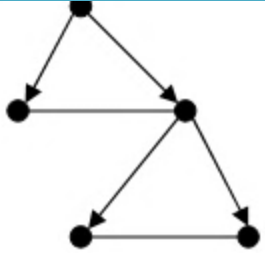
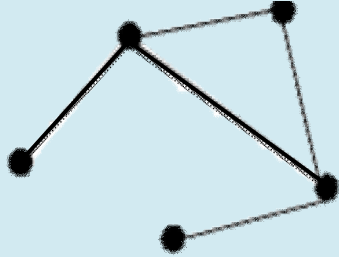
<p>l'architecture intégrée</p>	<p>Elle se démarque de l'architecture hiérarchique par l'existence de coordination entre les agents d'un même niveau hiérarchique</p>	
<p>architectures distribuées</p>	<p>Chaque agent est autonome dans ses Actions ou dans ses prises de décisions sans qu'il existe un agent de coordination de la structure globale [77]. Nous pouvons alors Représenter cette structure par un graphe Non orienté</p>	

Tableau 3.1 : Architectures d'intégrations des agents

Nous pouvons aussi trouver des architectures hétérarchiques qui est une architecture de pilotage fondée sur une prise de décision isoarchique, c'est-à-dire totalement auto organisée et décentralisée (s'effectuant sans aucune considération d'ordre hiérarchique). Dans ce contexte, il est considéré qu'un système de production est décomposable en entités à piloter. L'absence de centre de décision hiérarchique amène à doter chacune de ces entités d'une capacité propre de prise de décision qui leur permet, au travers d'un système de communication, d'élaborer, par négociation et en temps réel, une solution de pilotage qui soit la plus efficace et la plus efficiente possible [78].

4 Applications des systèmes multi agents

Les systèmes multi-agent sont devenus populaires pour la modélisation des systèmes complexes comme la chaîne logistique. En effet, plusieurs projets concernant l'utilisation des systèmes multi-agent dans la chaîne logistique ont été réalisés. Ces projets se confrontent à des problèmes différents de la chaîne logistique dont la conception et la gestion sont deux grandes catégories de problématiques. Clautier [79], Maturana [82] et Parunak [81] ont montré l'intérêt d'utiliser cette approche dans le domaine des chaînes logistiques. En particulier, Cloutier précise que le paradigme des agents est une métaphore naturelle avec les organisations en réseau depuis que les unités de production distribuées possèdent les mêmes caractéristiques que les agents de [80]:

- autonomie : une entreprise exécute des tâches d'elle-même sans intervention extérieure ;

- capacités sociales : un acteur de la chaîne logistique peut communiquer avec un autre acteur pour, par exemple, lui passer une commande de produits ou de services ;
- réactivité : une firme modifie son comportement si le marché ou la concurrence évolue ;
- pro-activité : une entreprise peut initier d'elle-même de nouvelles activités, comme par exemple décider de lancer un nouveau produit sur le marché ;

La littérature montre que les projets utilisant les SMA dans la CL se confrontent essentiellement à trois problèmes principaux : La modélisation, la conception et la gestion (piloteage). De plus, la manière de résoudre ces problèmes diffèrent aussi en fonction des projets : par exemple, le nombre et le rôle des agents varient considérablement. Nous présentons ci-dessous une liste non exhaustive de projets utilisant les SMA dans la CL :

1. OCEAN (Organisation and Control Emergence with an Agent Network) est un système de pilotage à architecture ouverte, décentralisée et basée sur des contraintes (distribution des ressources de production, distribution des données techniques, adaptabilité pour répondre à la dynamique de l'environnement). Le but de ce système est de montrer que des coopérations au niveau global peuvent apparaître à partir des compétitions au niveau individuel [87]. Ce travail a été réalisé à l'INSA de Lyon (France) et à l'université Montpellier 2 (France).
2. MASCF (*Multi-Agent Supply Chain Framework*) [83]: MASCF est un outil qui offre une méthodologie générique conçue pour faciliter et simplifier les phases d'analyse et de conception des applications basées sur les SMA dans les CL. MASCF adopte le modèle standard 'SCOR' pour la description de la CL et la méthodologie 'Gaia' pour le développement des SMA. Puisque le modèle SCOR et la méthodologie Gaia sont les éléments principaux de MASCF, ce dernier est alors considéré comme un outil générique largement applicable pour modéliser des CL par des systèmes multi-agent. MASCF ne spécifie aucune plateforme de développement des SMA, par conséquent, le développeur choisit la plate-forme appropriée.
3. MASC (*Multi Agent Supply Chain*) [84] [85]: MASC est une architecture à base d'agents dédiée à la modélisation et la simulation des processus de collaboration dans les chaînes logistiques. Les auteurs ont défini un modèle basé sur des agents distribués de type BDI (Belief, Desire and Intention) qui décrivent les objectifs, les intentions et les croyances des acteurs dans une organisation de type AGR (Agent-Groupe-Rôle) et réifiant quatre principaux rôles: Client, Fournisseur, Négociateur et Producteur. Ces agents appelés 'agents-acteurs' représentent les centres de décision de la chaîne. Trois types de décisions: stratégiques, tactiques et opérationnelles peuvent être abordées selon deux visions différentes. Une vision 'court et moyen termes' qui ne considère que les facteurs et les critères opérationnels tel que le coût, la qualité, le prix et le

niveau de stock. Une vision 'long terme' qui s'intéresse quant à elle aux facteurs et critères qui déterminent la viabilité à long terme de l'entreprise sur un marché concurrentiel. Des propositions ont été faites afin de rapprocher ces deux visions dans une démarche globale d'aide à la décision.

4. Dans [88], les auteurs proposent un système multi-agent utilisant une négociation multi contractuelle pour coordonner la fabrication dans une chaîne logistique. Le système est développé à l'aide de la plate-forme JADE et est utilisé dans une compagnie de fabrication de téléphones mobiles.
5. Dans [100], les auteurs ont utilisé la technologie des agents pour simuler une CL et les méthodes de gestion du stock de chaque entité qui y appartient. Les agents emploient des algorithmes génétiques pour calculer la quantité optimale à commander pour chaque échelon. Chaque agent communique et partage les informations de la demande avec les autres afin de minimiser le coût global. Les supply chain managers des différentes entités partagent les informations avec confiance. En outre, les agents explorent et stockent les connaissances des supply chain managers sous forme de règles et les réutilisent plus tard en cas de problèmes semblables. Avec cette approche, les auteurs ont montré que le coût total est réduit au minimum.
6. Le simulateur du jeu du Bois Québécois [101] (jeu dérivé du jeu de la bière (en anglais, Beer Game) : c'est un jeu qui a été proposé par Sterman en 1989 pour faire prendre conscience aux étudiants et aux industriels de la dynamique d'une chaîne logistique.) est un simulateur à base de système multi-agent servant à modéliser et simuler la chaîne logistique de l'industrie forestière du Québec. L'auteur a implémenté ce simulateur afin de valider deux principes qu'il a proposés pour diminuer l'effet coup de fouet. Ce jeu permet d'enseigner ce qu'est l'effet coup de fouet ainsi que la complexité de la dynamique d'une chaîne logistique.
7. Dragon Chain [102] a été implémenté pour automatiser la gestion de la chaîne logistique, et plus particulièrement pour réduire l'effet coup de fouet. Pour cela, ils se sont basés sur le modèle du jeu de la bière (à savoir les variantes Beer Game et Columbia Beer Game) et ont utilisé des agents utilisant des algorithmes génétiques pour déterminer la meilleure politique de réapprovisionnement. Dans ce projet, chaque entreprise a été modélisée par un agent.
8. L'Agent Building Shell de l'université de Toronto (Ontario, Canada) est une bibliothèque logicielle de classes supportant des outils d'implémentation d'agents. L'architecture de ces agents est réalisée selon quatre couches : (i) la couche de gestion de la connaissance, (ii) la couche d'ontologie, (iii) la couche de la coopération et des conflits et (iv) la couche de communication et de coordination (cette quatrième couche

est supportée par le langage COOL (pour COOrdination language). Plusieurs chercheurs ont travaillé sur ce projet, en particulier Fox et Barbuceanu [97], [89], [99], 1996, [98]. Dans ce projet, chaque entreprise a été modélisée par un agent.

9. Le projet développé par Telle [86] est dédié aux relations entre Airbus et ses fournisseurs. La méthodologie et l'outil de simulation associés qui sont proposés dans ce projet sont destinés au diagnostic puis à l'aide à l'amélioration des performances des relations Donneurs d'ordres / Fournisseur (DO/F) au sein d'une chaîne logistique. L'outil proposé utilise des techniques relevant de l'évaluation des performances et de l'aide à la décision dans le domaine de la coopération industrielle. La démarche d'aide à la coopération proposée s'appuie sur la chronologie type des étapes utilisant des approches évaluatives. Sa spécificité tient au fait qu'elle est conçue pour être utilisée par un couple d'utilisateurs, acteurs d'une réelle relation DO/F. Dans ce projet, le fournisseur est modélisé par 4 sous-agents. Un sous-agent pour la fonction approvisionnement, un sous-agent pour la fonction production, un sous-agent pour la fonction distribution et un sous-agent système de conduite.
10. MASCOT (*Multi-Agent Supply Chain cOordination Tool*) est une architecture reconfigurable, à niveau et basée sur les agents pour la planification et la coordination qui vise à améliorer l'agilité d'un réseau logistique. Précisément, ce système coordonne la production à travers les multiples installations (internes et externes) de l'entreprise, et évalue les décisions de conception de nouveaux produits/sous-composants et les décisions d'affaires stratégiques [90]. Ce travail a été réalisé à l'université Carnegie Mellon (Pittsburgh, PA, USA). Dans ce projet, chaque entreprise a été modélisée par un agent.
11. DASCh (*Dynamic Analysis of Supply Chain*) a été développé dans ERIM (Ann Arbor, MI, USA) par l'équipe de Parunak [92], [91] pour explorer les techniques de modélisation des réseaux de fournisseurs et des fournisseurs de leurs fournisseurs. En particulier, les flux de produits et d'informations sont agentifiés afin de modéliser le fait qu'ils ne soient pas parfaits (Prise en compte des délais et incertitudes sur les flux par des agents). Chaque entreprise a été modélisée par 2 agents et un agent pour les flux.

Ce bref état de l'art montre bien que les SMA sont devenus populaires pour la modélisation et le pilotage des chaînes logistiques. Néanmoins, nous relevons quelques lacunes:

- Le développement des applications pour la chaîne logistique à base de SMA reste assez compliqué et prend beaucoup de temps. Actuellement, le processus de modélisation prend beaucoup de temps puisqu'il n'existe pas de méthodes génériques pour la modélisation des CL en utilisant les systèmes multi-agent à l'exception de méthodologies telle que celle proposée dans [95] qui sont encore en cours de développement. Un nombre limité d'auteurs (comme [96], [94]) ont discuté

l'importance de développer des composants génériques et réutilisables, la plupart des applications utilisent des modèles particuliers liés à une problématique industrielle ou logistique spécifique.

- La littérature ne considère pas explicitement les phases d'analyse et de conception de l'application à base de SMA (qui sont des phases importantes dans le cycle de développement des applications industrielles) mais s'attaque directement à la phase d'implémentation à partir des besoins.
- Dans la littérature, il apparaît que la plupart des applications sont orientées recherche. En effet, un nombre important de chercheurs ont utilisé les SMA pour modéliser et simuler la chaîne logistique en vue de tester et d'évaluer des pratiques et des techniques de collaboration basées en général sur le partage d'informations

Au contraire il existe un nombre réduit de prototypes qui ont utilisé des langages de programmation (comme Java), logiciels commerciaux (comme ADE, G2), ou logiciels gratuits (freeware)/outils libres (open-source toolkits) (comme Swarm, JATLite, JADE) pour développer des applications réelles à base de SMA.

5 Méthodologies de développement de SMA

Actuellement, il existe un nombre important de méthodologies et d'outils de conception de SMA [158]. Chacune d'entre elles se fonde sur une base conceptuelle propre et recouvre un certain nombre d'aspects dont la prise en compte est considérée essentielle. La plupart de ces propositions méthodologiques sont inspirées des résultats du domaine du génie logiciel orienté objets. Dans cette section nous présentons un certain nombre d'exemples de ces méthodologies et nous terminons par quelques arguments qui renforcent notre choix de la méthodologie à utiliser pour notre application SMA dans le domaine de la chaîne logistique.

AUML [159] a pour but d'étendre UML avec des facilités permettant de décrire des agents. Ce formalisme a comme but de recommander une technologie pour l'adoption d'une sémantique, d'un méta-modèle et d'une syntaxe abstraite communes pour l'analyse et la conception des méthodologies basées sur les agents. Cette technologie devra permettre de couvrir le cycle de vie des produits et des outils de travail AUML et être en accord avec les spécifications existantes de FIPA et l'OMG (Object Management Group). Actuellement, les versions 2.0 et 2.1 d'UML ont intégré plusieurs concepts d'Agent UML, comme ceci est montré sur le site officiel d'AUML. Ainsi avec UML 2.0 et versions ultérieures, on pourra modéliser une application orientée agent. Pour plus de détails sur ce point, le lecteur pourra se référer à l'article de B. Bauer et J. Odell [160].

M-UML a été proposée par K. Saleh et C. El-Morr [161] pour modéliser les agents mobiles.

GAIA [162] est une méthodologie où le système multi-agent est vu comme une organisation composée de rôles interagissant entre eux. La méthodologie prend comme point de départ une description de la structure organisationnelle du système, composée de rôles et d'interactions entre eux. Deux phases sont distinguées (figure 3.7) : la phase d'analyse qui permet de constituer le modèle des rôles et le modèle des interactions, et la phase de conception durant laquelle sont créés les modèles d'agents, de services et d'accointances.

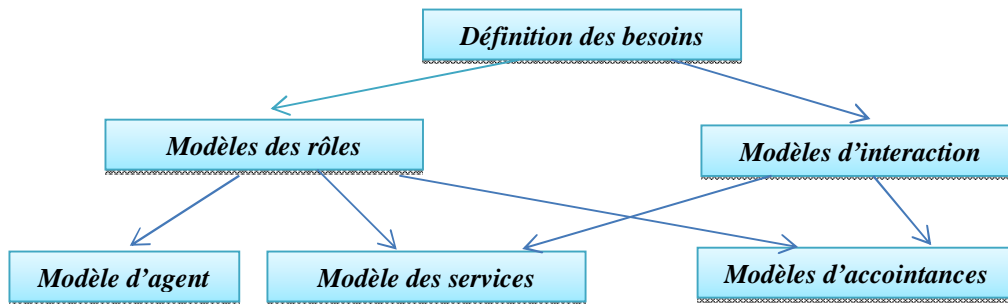


Figure 3.7 : Les relations entre les modèles de la méthodologie GAIA

Cette méthodologie aborde la conception d'un SMA sans un modèle d'architecture d'agent préétabli. Son utilisation conduit à un ensemble de spécifications qui peuvent se considérer au niveau d'analyse. Mais la conception est très faible, puisque aucun modèle computationnel n'est envisagé. Il n'y a pas non plus de guides méthodologiques, par exemple, pour assigner les rôles aux agents [105]. Et bien que dans sa version la plus récente sont envisagés des aspects organisationnels [106], aucun patron d'organisation n'est proposé, ni comment déterminer le type de rôles qui devrait exister dans une organisation. Par conséquent l'utilité de Gaia pour la réalisation de SMA est assez limitée, mis à part les apports qui peuvent être obtenus quant à l'étude qu'il fait sur la définition de rôles et d'interactions, bien que ces dernières se spécifient de manière assez simple et sans considérer les aspects de concurrence [105].

MaSE [103] (*Multi-agent System Engineering*) est une méthodologie générique qui considère comme point de départ le contexte initial du système, supposé contenir un ensemble bien défini d'exigences fonctionnelles, et permettre d'en extraire un ensemble de buts hiérarchisés. Elle est centrée sur les aspects pratiques de construction de SMA. Cette méthodologie est composée de deux phases. La phase d'analyse se déroule en trois étapes : capture des buts, application des cas d'utilisation et raffinement des rôles. La phase de conception comporte quatre étapes: création des classes d'agents, construction des conversations, assemblage des classes d'agents et conception du système. La méthodologie MaSE utilise des notations similaires à UML et propose un cycle de développement interactif et incrémental avec des activités d'analyse et de conception, qui sont assistées par l'outil Agent Tool [104].

MaSE propose de commencer avec une analyse des buts car elle considère que les buts sont plus stables dans l'analyse et la conception que les fonctions, les processus et les structures d'information. Cette analyse des buts et leur distinction claire par rapport à une analyse fonctionnelle classique sont les apports les plus intéressants de MaSE. Elle propose aussi un mécanisme pour l'identification des rôles et leur assignation aux agents.

Pour arriver finalement à un modèle computationnel des agents, il réalise, cependant, une certaine simplification quand on considère que chaque type d'agent correspond à une classe d'objets. D'autre part, certaines considérations, comme celle d'assumer que l'on dispose d'un ensemble de besoins fonctionnels initial ou la détermination d'un ensemble établi de conversations, font que leur applicabilité se limite à résoudre des problèmes similaires à ceux traités avec une méthodologie orientée objets classique [105]. C'est ainsi que le résultat est un ensemble de classes dont le comportement est défini par des automates mais il n'est pas clair comment on pourrait aborder la construction des agents délibératifs, comme c'est le cas des agents BDI, puisque on ne définit plus, par exemple, comment gérer et contrôler la satisfaction ou l'échec des buts.

AALAADIN [107] est un méta-modèle reposant sur les concepts organisationnels que sont les agents, groupes et rôles (AGR), pris ici comme concept primitifs, et issus d'une analyse préalable du système envisagé. L'organisation est définie comme un cadre d'activité et d'interaction, mis en place par la définition de groupes, rôles et de leurs relations, et considérée comme l'ensemble des relations structurelles dans un ensemble d'agents. Un SMA est vu par l'approche AALAADIN, comme un ensemble de groupes d'agents interconnectés par l'intermédiaire d'agents appartenant à plusieurs groupes (*figure 3.8*). Aucune contrainte ou pré-requis n'est imposée sur l'architecture interne des agents et aucun formalisme ou modèle particulier n'est supposé pour en décrire leur comportement. Chaque agent est simplement décrit comme une entité autonome communicante qui joue des rôles au sein des différents groupes. La méthode AALAADIN a donné lieu au développement de la plate-forme MadKit, développée par O. Gutknecht et J.Ferber [108], sur laquelle nous reviendrons.

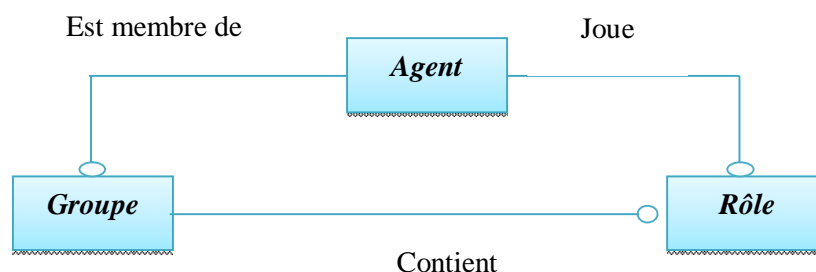


Figure 3.8 : Les trois concepts centraux d'AALAADIN

La liste présentée ci-dessus n'est pas limitative. Dans la littérature on trouve d'autres

méthodologies partant des résultats du domaine du génie logiciel orienté objets (ADELFE [109], AAIL/BDI [110], Kendall [111], MASB [118], ODAC [117], MESSAGE [119]) ou appliquent des techniques de construction de systèmes experts (CoMoMAS [113], MAS-Common KADS [114]). D'autres donnent plus d'importance aux relations entre les divers aspects du SMA, comme dans VOYELLES [116], INGENIAS [115] ou MASSIVE [112].

Ci-dessous, nous montrons quelques arguments guidant et renforçant notre choix de la méthodologie à utiliser pour la phase de modélisation de notre architecture :

- Il existe un nombre important de méthodologies et d'outils de conception de SMA et la communauté travaillant dans le domaine des agents se trouve en face du problème d'identification d'un vocabulaire commun pour les supporter [120]. Cependant, il n'existe pas encore une méthodologie standard suffisamment mature qui aborde tous les aspects nécessaires pour définir un SMA, ni tout le cycle de développement de ce type de logiciel.
- J. Pavón, l'un des auteurs de la méthodologie INGENIAS [115] affirme dans [105] que malgré l'existence de beaucoup d'applications basées sur les agents, il manque encore l'expérience pour la conception et la construction de SMA à un niveau industriel (une révision plus récente de l'état de l'art et des applications de la technologie des agents se trouve dans [121]).
La grande majorité des applications basées sur les agents est construite sans utiliser des composants agents réutilisables et ne sont pas généralisables. C'est pour cela que la recherche sur les méthodes, les outils et les plates-formes d'agents a beaucoup d'importance pour l'implantation de la technologie d'agents en dehors du domaine purement académique. Dans la situation actuelle, peu de méthodologies sont amenées des cas réels significatifs à la pratique et sont assistées par des outils. En plus, les méthodologies existantes ne considèrent que certains aspects du cycle de vie, généralement l'analyse et la conception. Pour l'implémentation, la plupart des méthodologies sont conditionnées par l'utilisation d'une architecture d'agents déterminée.
- Le paradigme d'agent peut s'étendre à celui d'objet. En conséquence, une méthodologie de développement de SMA devrait profiter de l'expérience de l'approche objet. Les méthodologies orientées objets ont acquis un haut niveau de maturité et leurs avantages sont largement reconnus. Une grande partie des développeurs de logiciel sont familiarisés avec elles et utilisent la large gamme d'outils disponibles [105].

Afin de donner un cadre méthodologique au processus de modélisation, nous allons par la suite appliquer le formalisme AUML et par conséquent UML 2.1 qui contient un grand nombre de concepts liés aux agents. En fait, UML est un standard largement connu, accepté et a fait ses preuves pour la modélisation des applications orientées objet. Le fait que

le paradigme des agents est considéré comme une extension du paradigme objet a conduit plusieurs auteurs pour étendre UML afin de tenir compte des concepts agents. Une discussion sur ce sujet a été présentée dans [122]. En choisissant AUML, nos modèles proposés dans ce manuscrit seront largement diffusés et compris. D'autres arguments plus techniques liés à la problématique que nous traitons et qui renforcent notre choix seront présentés dans les prochains chapitres.

6 Quelques Plates-formes de développement des SMA

Des plateformes génériques sont apparues pour le développement de systèmes reposant sur l'utilisation du paradigme agent. Après avoir examiné plusieurs de ces plateformes, très peu apparaissent comme pouvant servir réellement à la conception de tout type d'application. En effet, la plupart s'adapte souvent à la conception d'un type d'applications ou d'une classe d'applications mais rarement plus. Il apparaît, de plus, que la définition d'un agent est beaucoup trop vague pour qu'un modèle générique et universel puisse exister. Parmi les plateformes les plus connues on peut citer : SWARM [123] [131] [132], Voyager [130], DIMA [128], MADKIT [129], JADE [124], [125], [126], [127].

SWARM est une plate-forme de développement qui possède une forte orientation vers le développement d'applications de simulation à événements discrets de systèmes complexes. Elle a été développée en Objective C et un portage vers Java est en cours. De nombreuses applications de simulations reposent sur l'utilisation de la plate-forme SWARM, en voici quelques exemples :

- Heatbugs: un modèle temporel d'agents locaux interagissant au travers d'un environnement pour créer un comportement global complexe,
- Mousetrap fission : une simulation à événements discrets de fission nucléaire,
- Market : un petit modèle de marché pour une matière première.

De nombreux autres exemples d'applications basées sur SWARM peuvent être trouvés à l'adresse [133].

Voyager est une plate-forme de développement commerciale. Elle fonctionne en environnement distribué et repose sur le standard Corba. Son domaine d'application est essentiellement orienté vers le commerce électronique et les serveurs d'applications WEB [130].

DIMA (*Développement et Implémentation de systèmes Multi-Agents*) a été développée par Zahia Guessoum dans le cadre de ses travaux de thèse [128].

L'architecture des agents qui a été choisie pour le développement de cette plate-forme est une architecture modulaire (module de supervision, module de communication, module de

perception, module de raisonnement) .

MadKit (*MultiAgent Development Kit*) est une plate-forme générique de conception et d'exécution de systèmes multi-agents [107]. Elle a été développée par Olivier Gutknecht au sein du Laboratoire LIRMN. L'ensemble du développement a été fait en java. Cette plate-forme s'appuie pour la réalisation de SMA sur le modèle conceptuel Aalaadin [134]. L'originalité de cette plate-forme est qu'elle est basée sur un modèle organisationnel plutôt que sur une architecture d'agents ou un modèle d'interaction spécifique.

Cette plate-forme fournit un ensemble de classes Java permettant la réalisation d'agents par héritage de classes. Il est aussi possible d'utiliser une interface graphique, nommée G-Box, qui permet de visualiser les agents sous forme de composants graphiques sur lesquels il est possible d'agir.

Zeus est une plate-forme de développement de SMA générique, personnalisable et pourrait être augmentée par l'adjonction de nouveaux composants. Elle est présentée sous forme d'un ensemble de classes implémentées dans le langage Java. En cela, elle est complètement portable et peut être utilisée sur tout système disposant d'une machine virtuelle Java. Un agent est composé de plusieurs objets et threads. Ces classes sont catégorisées dans trois groupes fonctionnels : une librairie de composants, un outil de développement et un outil de visualisation. De par sa généralité, elle peut être utilisée dans la conception d'un grand nombre d'applications mais en retour nécessitera des augmentations assez nombreuses et spécifiques à chaque type d'application[135].

JADE (*Java Agent DEvelopment framework*) est une plate-forme multi-agent créée par le laboratoire TILAB décrite par Bellifemine et al. dans [126] [127]. JADE permet le développement de systèmes multi-agent et d'applications conformes aux normes FIPA [76]. Elle est 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 « Director Facilitator » 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.

Ces trois modules sont activés à chaque démarrage de la plate-forme. Il existe des outils qui ont été développés dans la plate-forme JADE afin de supporter la difficulté de la tâche du débogage des applications multi-agents. Chaque outil est empaqueté comme un agent, obéissant aux mêmes règles, aux mêmes possibilités de communication et aux mêmes cycles de vie d'un agent générique (agentification de service).

- Agent RMA Remote Management Agent

Le RMA permet de contrôler le cycle de vie de la plate-forme et tous les agents la composant. L'architecture répartie de JADE permet le contrôle à distance d'une autre plate-forme. Plusieurs RMA peuvent être lancés sur la même plate-forme du moment qu'ils ont des noms distincts.

- Agent Dammy

L'outil DummyAgent permet aux utilisateurs d'interagir avec les agents JADE d'une façon particulière. L'interface permet la composition et l'envoi de messages ACL et maintient une liste de messages ACL envoyés et reçus. Cette liste peut être examinée par l'utilisateur et chaque message peut être vu en détail ou même édité. Plus encore, le message peut être sauvegardé sur le disque et renvoyé plus tard.

- Agent Directory Facilitator

L'interface du DF peut être lancée à partir du menu du RMA. Cette action est en fait implantée par l'envoi d'un message ACL au DF lui demandant de charger son interface graphique. L'interface peut être juste vue sur l'hôte où la plate-forme est exécutée. En utilisant cette interface, l'utilisateur peut interagir avec le DF.

- Agent Sniffer

Quand un utilisateur décide d'épier un agent ou un groupe d'agents, il utilise un agent sniffer. Chaque message partant ou allant vers ce groupe est capté et affiché sur l'interface du sniffer. L'utilisateur peut voir et enregistrer tous les messages, pour éventuellement les analyser plus tard. L'agent peut être lancé du menu du RMA ou de la ligne de commande suivante : `Java jade.Bootsniffer;jade.tools.sniffer.sniffer`

- Agent Inspector

Cet agent permet de gérer et de contrôler le cycle de vie d'un agent s'exécutant et la file de ses messages envoyés et reçus.

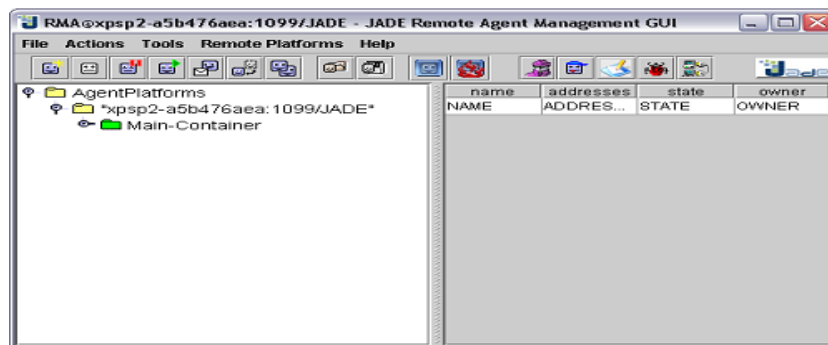


Figure 3.9 : L'interface de la plate forme JADE.

7 Conclusion

Ce chapitre nous a permis de mettre le point sur les systèmes multi-agent. Nous avons présenté les grands concepts qui définissent le domaine, les méthodologies et les plates-formes de développement de système multi-agent. Deux choix importants ont été défendus dans ce chapitre et constituent les briques de base des chapitres suivants. Le premier choix concerne le langage AUML qui sera utilisé pour la phase de modélisation (chapitre 4). Le deuxième choix concerne la plate-forme JADE qui sera utilisée dans la phase d'implémentation (chapitre 5).

Chapitre 4

*Architecture Distribuée A Base D'agent Pour L'aide
A La Décision Dans La Chaine Logistique*

2 Introduction.....	96
2 Justification du choix.....	96
3 Notre approche multi agents.....	97
3.1 Identification des agents et leurs rôles.....	97
3.2 Structure statique du système.....	99
3.2.1 Structure de l'agent « App ».....	99
3.2.2 Agent « Dist »	102
3.2.3 Agent « Fab »	105
4 Protocoles de négociation.....	106
4.1 Diagramme de séquence.....	107
4.2 Négociation heuristique.....	108
5 Modélisation des comportements des agents.....	109
5.1 Comportement des agents.....	109
5.1.1 Comportement de l'agent « Dist ».....	110
5.1.2 Comportement de l'agent « App »	112
5.1.3 Comportement de l'agent « Fab ».....	114
6 Exemple d'interaction entre les agents pour la prise de commande ferme....	115
7 Conclusion.....	117

1 Introduction

Dans le chapitre précédent, nous avons présentés plusieurs travaux liés à la modélisation des chaînes logistiques par des SMA. On a montré aussi que chaque modèle traite un problème précis. De notre part, dans ce chapitre nous allons proposer notre modèle qui convient à un grand nombre de chaînes logistiques ou groupements d'entreprises.

La spécificité de notre modèle est qu'il traite les situations d'urgences. Par exemple, une situation de planification opérationnelle (court terme: commande urgente [136] [137], commande non prévue, etc.) conduisant à une négociation pour avoir la quantité commandée dans les dates de livraison exigés, etc.

Tout d'abord, on présente notre approche méthodologique à suivre afin de délimiter la structure et le périmètre de la CL à modéliser, ceci nous mène à identifier les agents nécessaires et représenter une vue statique de notre système. Ensuite nous proposons et nous modélisons avec AUML quelques protocoles de négociation et d'échanges utilisés par les agents de notre architecture. Puis nous modélisons le comportement de chaque agent par un diagramme d'états-transitions AUML. Enfin, nous concluons en présentant notre contribution à la modélisation d'une CL et aux protocoles de négociation entre agents.

2 Justification du choix

Malgré nous avoir justifié dans les chapitres précédents le choix de notre approche agent pour la modélisation de notre problématique, nous allons citer encore qu'elle que avantage de l'utilisation SMA dont les principales sont les suivantes :

L'approche multi-agents est justifiée par : l'adaptation à la réalité ; la coopération; la résolution de problèmes complexes; l'intégration d'expertise incomplète ; la modularité; l'efficacité; la fiabilité; et la réutilisation. Elle repose sur un ensemble de concepts tels que la communication, l'organisation, la coopération, la coordination entre des entités plus ou moins intelligentes et autonomes. L'ensemble des concepts de l'approche multi-agents font de celle-ci une approche très riche et très puissante en termes de modélisation.

3 Notre approche multi agents

Notre approche est basée sur le modèle du Système Multi-Agents. Nous décrivons plus précisément les différents composants (agents) de notre système et leurs interactions en montrant les relations établies entre les différents agents.

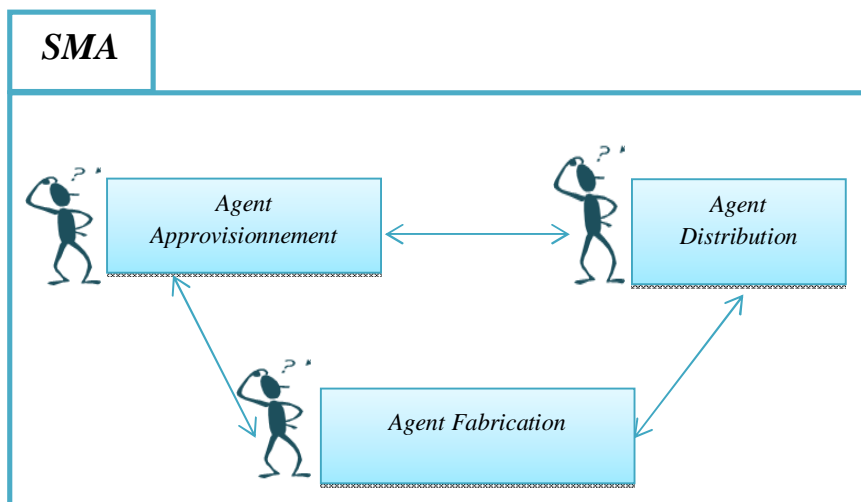
3.1 Identification des agents et leurs rôles

Nous présentons, dans cette partie, l'approche multi agents développée pour la conduite d'une chaîne logistique distribuée. Chaque entreprise du réseau est représentée par 3 agents. Le tableau 4.1 présente les noms et les rôles de ces agents.

<i>Nom de l'agent</i>	<i>Rôles</i>
<i>App</i>	Agent lié au processus des achats et d'approvisionnements de l'entreprise aussi bien en composants qu'en matière de prestation de services telle que la sous-traitance ou le transport.
<i>Fab</i>	Agent lié au processus de fabrication(production).
<i>Dist</i>	Le rôle principal de ce type d'agent est la prise en charge des procédures de ventes. Lorsqu'une demande d'un client arrive, cet agent est activé. Il peut alors soit répondre par lui-même à cette demande soit s'appuyer sur les autres agents de la structure pour lui apporter les éléments nécessaires à la prise de décision.

Tableau 4.1 : Agents du système

Chacun de ces agents(*Figure 4.1*), cherche à coopérer avec les autres pour augmenter le nombre de scénarios possibles devant une commande client imprévue, En ce qui concerne la réactivité et la cognition, nos agents auront des stimuli réactifs(transfert de messages, calcul de coût suite à la réception d'un message) ainsi que des stratégies de décisions.



SMA : Système Multi Agent

Interaction



Figure 4.1 : Agents cognitifs de l'entreprise.

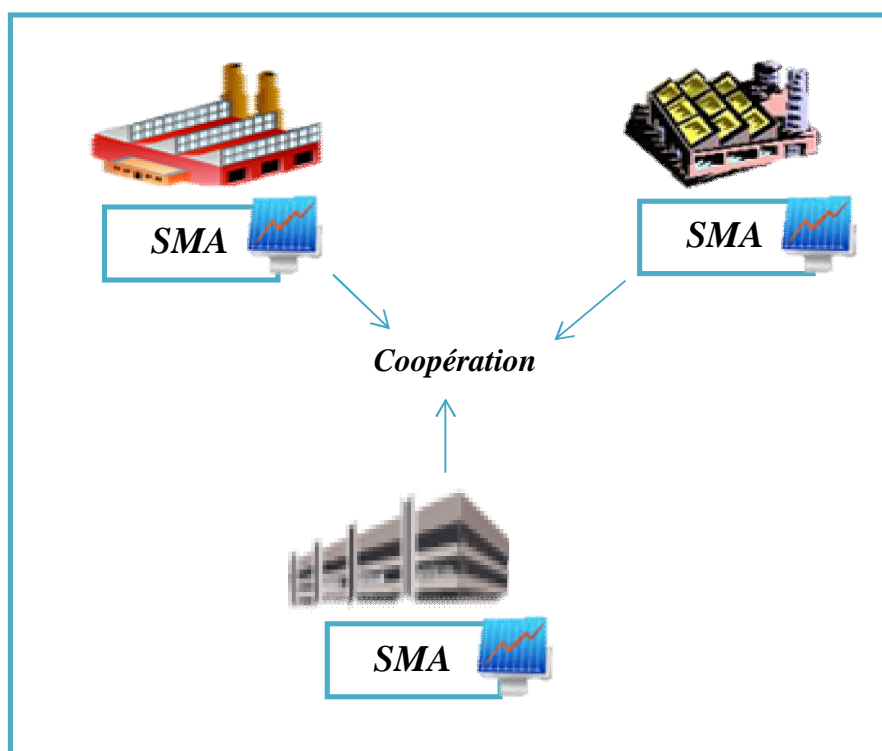


Figure 4.2 : Architecture distribuée d'aide à la décision dans une chaîne logistique.

3.2 Structure statique du système

Huget [138] a proposé une extension des diagrammes de classes UML afin de prendre en compte les caractéristiques des agents. AUML permet de représenter plusieurs niveaux d'abstraction lors de la conception des diagrammes de classes. Deux niveaux d'abstraction sont couramment utilisés : Le niveau conceptuel et le niveau implémentation.

Le niveau conceptuel est une vue assez haute du système multi-agent éliminant toutes les informations qui sont superficielles pour comprendre la structure du système : les attributs, les opérations, les rôles, etc. Le niveau implémentation donne en détail le contenu des agents. Nous présentons ci-dessous le diagramme de classe niveau implémentation de chaque agent.

Dans notre architecture, chacun des agents présentés dans le *tableau 4.1* possède les attributs suivants:

- Id : une identification
- Nom
- Loc : localisation

Ces attributs sont appelés 'des contraintes systèmes' comme spécifié dans la modélisation AUML.

3.2.1 Structure de l'agent « App »

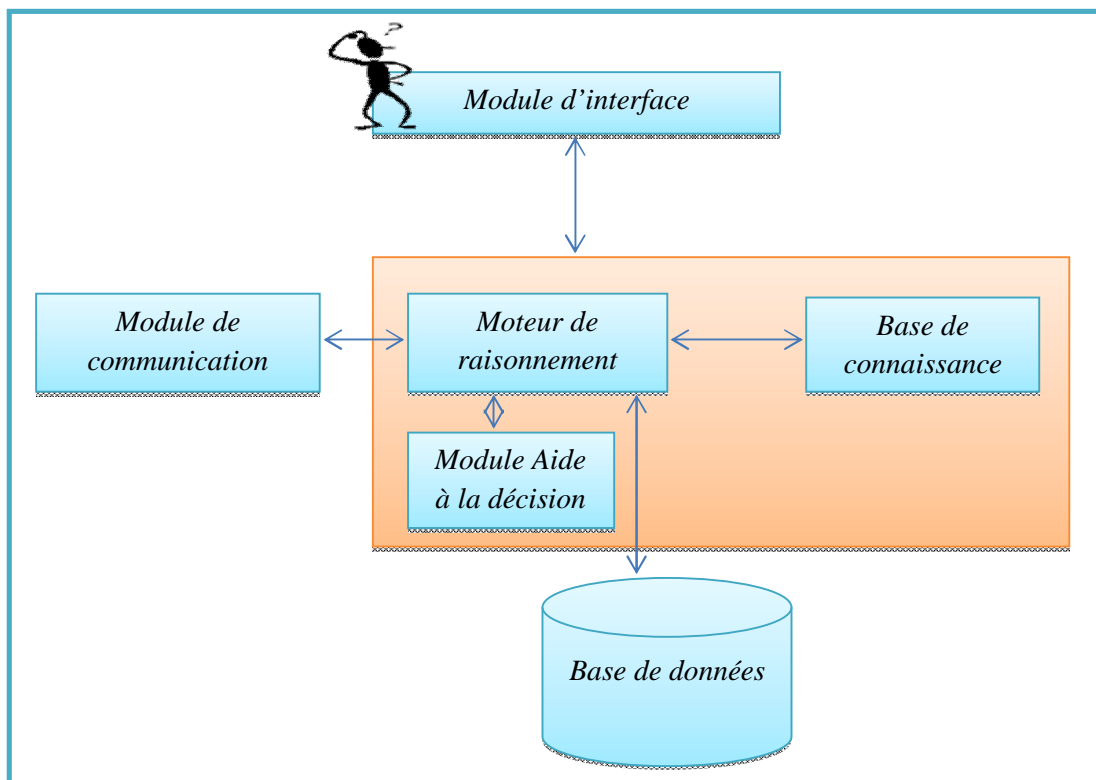


Figure 4.3 : Structure de l'agent Approvisionnement

Comme c'est montré sur le tableau 4.1, Le rôle de cet agent est de gérer les achats de l'entreprise, de demande d'approvisionnement jusqu'à la réception des matières premières, produits semi finis ou produits finis. Il comporte les modules suivants : (1) Un module de communication, (2) Un moteur de raisonnement, (3) Une Base de Connaissance approvisionnement, (4) Une base de données, (5) module d'aide à la décision.

On donne la définition de chacun comme suit :

- **Module de communication** : Le module de communication est responsable de l'interaction de l'agent avec les autres agents du système pour obtenir, envoyer, recevoir des informations. Par exemple envoyer à l'agent distributeur « Dist_i » d'une autre entreprise une demande pour recevoir ses propositions à propos d'un scénario d'approvisionnement. Comme il peut par exemple envoyer un message à l'agent fabrication « Fab » pour lui informer si les quantités demandées des matières premières sont disponibles ou pourrait l'être.
- **Base de Connaissance App** : Contient un ensemble de critères définissant les conditions d'approvisionnement, soit les critères liés aux choix des produits (prix, les marques et leurs caractéristiques), ou les critères liés aux choix des offres de fournisseurs (prix du produit, la garantie, la qualité, la gamme de produits, délai de livraison, la disponibilité).
- **BD approvisionnement** : englobe les données concernant les achats. Ainsi nous citons:
 - Toutes les données concernant les fournisseurs (nom, localisation, tel, fonction, code comptable, etc.) .
 - Toutes les informations de la demande d'achat (N°, Demandeur, Article, Quantité, Prix unitaire, Date, Prix total, etc.) .
 - Toutes les informations concernant les contrats établis (Article, Prix, délai de livraison, Durée du contrat, etc.) .
 - Toutes les données concernant les articles à acheter (Nom, famille, sous famille, tarif de vente, tarif d'achat, etc.) .
 - Toutes les informations concernant le bon de commande (N°, date, fournisseur, Article, Quantité, marque, Prix, etc.).
 - Les données concernant la méthode utilisée pour gérer le stock d'une matière première. Par exemple : système à point de commande, système à re-complètement périodique (Quantité fixe à commander, Point de commande, Niveau de re-complètement, etc.).

- Toutes les données concernant les articles à stocker (désignation, Marque, Fournisseur, Prix, Formule de calcul, etc.).
- **Moteur de raisonnement** : contrôle l'exécution de ses actions.
- **Interface utilisateur** : Ce module permet l'interaction avec l'utilisateur, elle fournit les outils nécessaires pour que l'utilisateur puisse faire une re-planification par exemple. Les résultats de l'application de l'aide à la décision s'affichent sur cette interface.
- **Module aide à la décision** : Ce module comporte les méthodes proposé d'aide à la décision, Nous distinguons deux contextes d'aide à la décision: (i) Aide à la décision pour la sélection des fournisseurs des matières premières en réalisant un gain de temps et d'argent (ii) Aide à la décision pour la sélection des fournisseurs de produits finis. Les propositions de ce module sont guidées par la négociation et la communication avec les autres agents de l'entreprise « Fab » et « Dist ».

L'agent « App » gère entre autres, plusieurs données relatives aux matières premières constituant le produit fini fabriqué par l'entreprise. En effet, la fabrication du produit fininécessite plusieurs matières premières ou produits semi fini. Le stock de chaque matière première (appelé aussi stock de fabrication) peut être géré par une méthode de gestion de stock. Le management industriel et logistique connaît plusieurs systèmes et politiques de gestion de stock. Elles dépendent de la durée de vie des articles (pouvant être longue ou courte) et du type du processus de la demande. Cette dernière peut être globalement constante (on parle dans ce cas de demande stationnaire) ou au contraire présenter des évolutions importantes au cours du temps, on parlera alors de demande non-stationnaire.

L'agent « App » utilise des protocoles de négociation afin de négocier les command espasées, les scénarios possibles en cas des demandes clients imprévues, etc. Dans ce cadre, on a proposé le protocole de Négociation_Heuristique.

La *figure 4.4* montre le diagramme de classe niveau implémentation de l'agent « App ».Ce diagramme sera complété par d'autres informations dans les sections suivantes.parmi ses opérations, on trouve :

- MAJ_Plan_App() : cette opération permet de mettre à jour le plan d'approvisionnement en cas de modification.
- MAJ_Stock_App() : cette opération permet de mettre à jour le stock d'approvisionnement lors de la réception des matières premières.
- passer_CMD() : cette méthode permet de passer une commande caractérisée par une quantité et un délai de livraison à l'agent « Dist » d'un autre acteur.

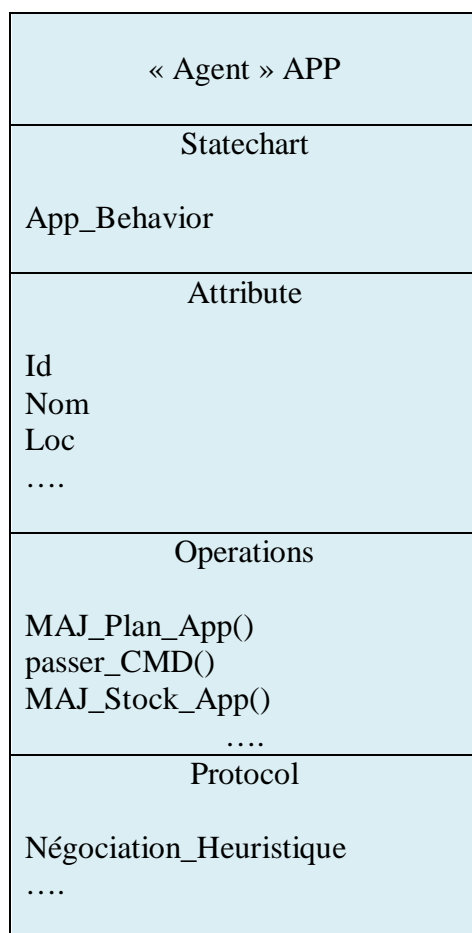


Figure 4.4 : Le niveau implémentation pour l'agent « App ».

L'agent « App » passe d'un état donné à un autre en fonction des actions se produisant dans l'environnement ou en fonction des messages reçus. Son comportement est modélisé par le diagramme d'états-transitions (App_Behavior) de la figure 4.11.

3.2.2 Agent « Dist »

Le rôle de cet agent est de gérer les ventes de l'entreprise, de la demande jusqu'à la délivrance des produits finis. Il comporte les modules suivants : (1) Un module de communication, (2) Un moteur de raisonnement, (3) Une Base de Connaissance distribution, (4) Une base de données.

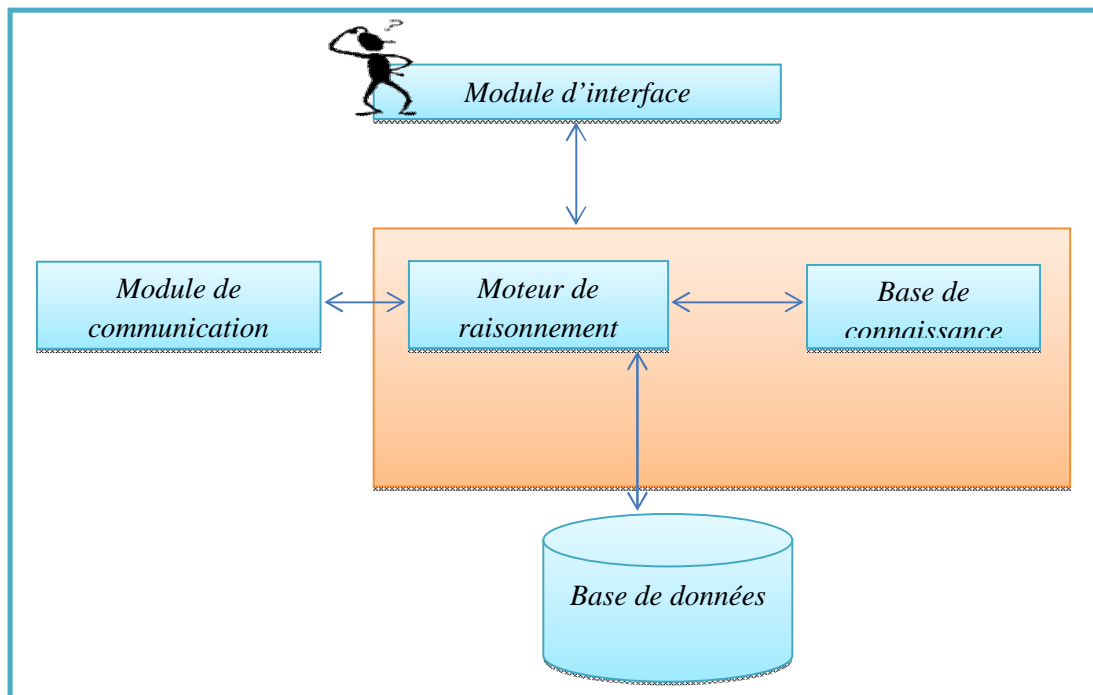


Figure 4.5 : Structure de l'agent Distribution « Dist »

On donne la définition de chacun comme suit :

- **Module de communication** : Le module de communication est responsable de l'interaction de l'agent avec les autres agents du système pour obtenir, envoyer, recevoir des informations. Par exemple recevoir une demande d'achat de l'agent « App_i » d'une autre entreprise, Comme il peut envoyer un ordre de production à l'agent fabrication « Fab ».
- **Base de Connaissance Dist**: Contient un ensemble de critère définissant les conditions de distribution, soit les critères liés aux choix des offres des clients .
- **BD distribution** : englobe les données concernant les produits finis et les clients de l'entreprise. Ainsi nous citons:
 - Toutes les données concernant les clients (nom, localisation, tel, fonction, code comptable, etc.) .
 - Toutes les informations concernant les contrats établis (Article, Prix, délai de livraison, Durée du contrat, etc.) .
 - Toutes les données concernant le produit a vendre (Nom, famille, sous famille, tarif de vente, tarif d'achat, etc.) .

- Les données concernant la méthode utilisée pour gérer le stock de produit fini. Par exemple : système à point de commande, système à re-complètement périodique (Quantité fixe à commander, Point de commande, Niveau de re-complètement, etc.).
 - Toutes les données concernant les articles à stocker (désignation, Marque, Fournisseur, Prix, Formule de calcul, etc.).
- **Moteur de raisonnement** : contrôle l'exécution de ses actions.
- **Interface utilisateur** : Ce module permet l'interaction avec l'utilisateur, elle fournit les outils nécessaires pour que l'utilisateur puisse changer les plan de distribution par exemple. Le résultat de la recherche et de la communication avec les autres agents « App », « Fab » s'affichent sur cette interface..

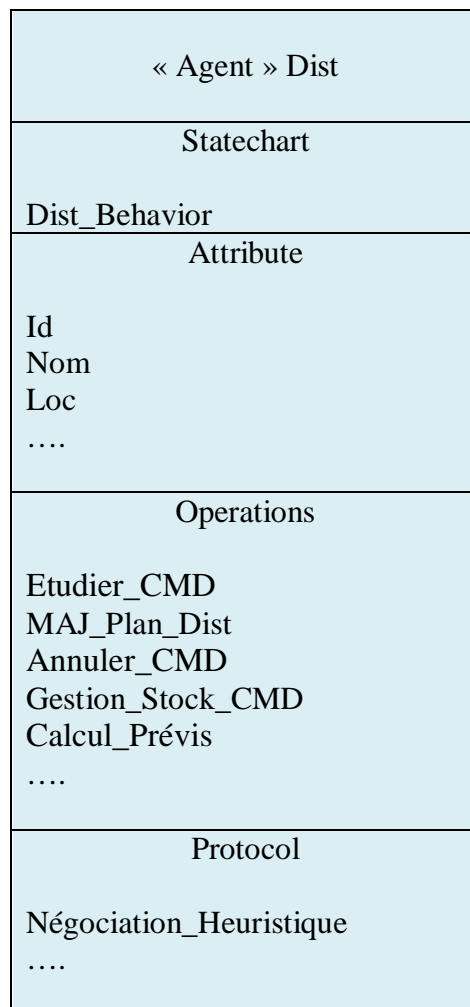


Figure 4.6 : Le niveau implémentation pour l'agent « Dist ».

En particulier, il est lié à la gestion des commandes des clients et le stock de distribution (stock de produits fins). En conséquence, parmi ses opérations, on trouve :

- **Etudier_CMD** : Dans cet état, l'agent « Dist » étudie la possibilité de livraison de la commande qui peut être nouvelle ou en cours d'étude.
- **MAJ_Plan_Dist** : Dans cet état, l'agent « Dist » met à jour le plan de livraison pour tenir compte de la commande confirmée par l'agent « App ».
- **Annuler_CMD** : Dans cet état, l'agent « Dist » doit annuler la commande confirmée ou se trouvant en cours d'étude.
- **Calcul_Prévis** : Dans cet état, l'agent calcule les prévisions et met à jour les plans nécessaires.

L'agent gère entre autres les informations suivantes pour chaque produit fini fabriqué par l'entreprise :

L'agent « Dist » reçoit des commandes envoyées par les agents « App » des autres acteurs. Chaque commande possède deux principaux paramètres :

- quantité commandée.
- délai de livraison.

Chacun de ces deux paramètres peut être modifiable ou non (possibilité d'avoir une négociation). Cela correspond à la possibilité de modifier ou non la valeur de l'attribut dans une contre-proposition. Par exemple, la quantité commandée peut être livrée intégralement ou par partie. Ceci influe sur la décision à prendre par l'agent. *La figure 4.6* montre le diagramme de classe de l'agent « Dist ». Il utilise des protocoles de négociation afin de négocier les commandes passées, les scénarios possibles en cas d'aléa, etc.

L'agent « Dist » passe d'un état donné à un autre en fonction des actions se produisant dans l'environnement ou en fonction des messages reçus. Son comportement est modélisé par le diagramme d'états-transitions (*Dist_Behavior*) de *la figure 4.10*.

3.2.3 Agent « Fab »

L'agent « Fab » est lié au macro-processus 'Fabrication' de l'entreprise. En plus des attributs (id, nom et loc), cet agent gère des données techniques relatives à la fabrication d'un produit fini comme par exemple le cycle de production d'une unité de produit fini et la capacité de production. Parmi ses opérations, on trouve notamment :

- **MAJ_Prog_Fab ()** : cette méthode permet la mise à jour du programme de production en cas de modification.

- annuler_DMD () : elle permet d'annuler une demande de production encours d'étude ou déjà commencée.

L'agent « Fab » passe d'un état donné à un autre en fonction des actions se produisant dans l'environnement (par exemple : une perturbation) ou en fonction des messages provenant des agents « App » et « Dist » du même acteur. Son comportement est modélisé par le diagramme d'états-transitions (Fab_Behavior) de la figure 4.12.

La figure 4.7 montre le diagramme de classe niveau implémentation de l'agent « Fab ».

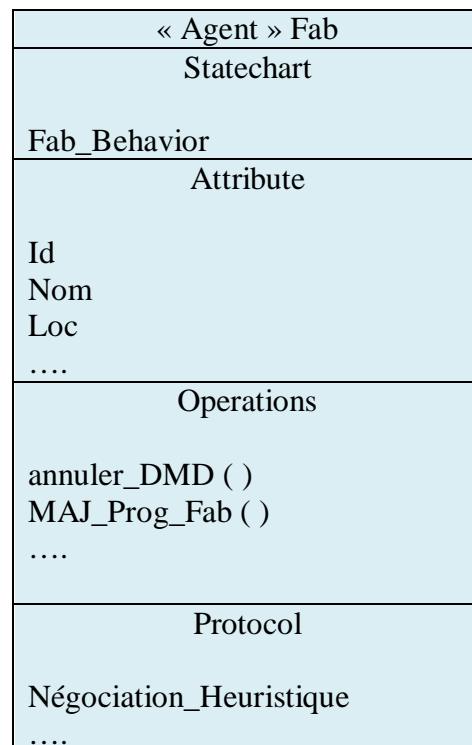


Figure 4.7 : Le niveau implémentation pour l'agent « Fab ».

4 Protocoles de négociation

La négociation est le mécanisme par lequel les agents peuvent arriver à un accord commun. Dans le cas des agents cognitifs et des SMA, La négociation est une composante de base de l'interaction et cela parce que les agents sont autonomes, il n'ya pas de solution imposée à l'avance et les agents doivent arriver à trouver des solutions dynamiquement, pendant qu'ils résolvent les problèmes. Pour modéliser la négociation entre les agents composant notre système, nous avons pris en compte les aspects suivants :

- L'objet de négociation : un objet abstrait qui comprend les attributs que les agents veulent négocier. Dans notre architecture, plusieurs objets sont sujets à négocier selon la situation. Nous trouvons entre autres, le scénario acceptable en cas d'aléas, la commande avec ses attributs quantité et date de livraison,
- Le processus de décision : le modèle que l'agent utilise pour prendre des décisions pendant la négociation. La partie la plus importante de la prise des décisions dans ce cas est la stratégie de négociation qui permet de déterminer quelle primitive de négociation l'agent doit choisir à un certain moment.
- Le langage de communication : le langage utilisé par les agents pour échanger des informations pendant la négociation. Dans notre architecture, les agents communiquent à l'aide du langage FIPA-ACL [76], il s'agit du langage standard utilisé par l'environnement JADE [124] que nous avons utilisé pour développer notre architecture (chapitre 5).
- Le protocole de négociation définissant l'ensemble des règles qui régit la négociation : Les participants possibles à la négociation, les propositions légales que les participants peuvent faire, les états de la négociation et enfin une règle pour déterminer quand on est arrivé à un accord ou quand il faut s'arrêter parce qu'aucun accord n'a pas pu être trouvé.

4.1 Diagramme de séquence

Un diagramme n'est pas un modèle mais une représentation graphique de quelques éléments du modèle. Les diagrammes constituent l'expression visuelle et graphique. Il existe plusieurs types de diagrammes dynamiques dans AUML : les diagrammes de déploiement, de collaboration, de séquences, de cas d'utilisation, d'états-transitions et d'activités. Un diagramme ne montre pas toutes les informations, plusieurs diagrammes sont nécessaires pour illustrer le modèle complet.

C'est pourquoi plusieurs diagrammes sont nécessaires pour donner une description dynamique complète de l'agent. Nous avons retenus le diagramme de séquence pour modéliser les interactions entre les différents agents, en particulier, les protocoles de négociation. Plusieurs protocoles standard de négociation ont été modélisés par le diagramme de séquence comme FIPA Request Protocol, FIPA Contract Net Protocol, FIPA English Auction Protocol [76].

Un diagramme de séquence est en deux dimensions : l'axe vertical correspond au temps tandis que l'axe horizontal recense les agents participants aux interactions. Pour représenter la création ou la destruction d'un agent sur un diagramme de séquence, il faut démarrer ou arrêter sa ligne de vie aux points appropriés, dans les autres cas de figure, la ligne de vie est

représentée de haut en bas du diagramme. Ils doivent être clairs pour faciliter la communication. Les détails non pertinents doivent être évités. Un diagramme de séquence montre les interactions entre agents (ou rôles) sous la forme de séquences dans le temps. En particulier, il montre les agents participants à l'interaction par leur 'ligne de vie' et les messages qu'ils échangent ordonnancés dans le temps.

4.2 Négociation heuristique

La négociation heuristique est montrée dans la *figure 4.8* [170]. Dans ce protocole, plusieurs propositions et contre-propositions peuvent être échangées dans les différentes étapes afin d'arriver à un accord. L'agent A, avec la proposition pA , est l'initiateur de la négociation, alors que l'agent B (participant) peut répondre avec les réponses $p1B$, $p2B$, $p3B$ (to ModifyRequest). Le nombre des contre-propositions est limité. Une fois que cette limite est atteinte, les agents arrivent à un rejet. Nous proposons de récapituler le fonctionnement du protocole de négociation heuristique à l'aide d'un digramme de séquence AUML (*figure 4.9*).

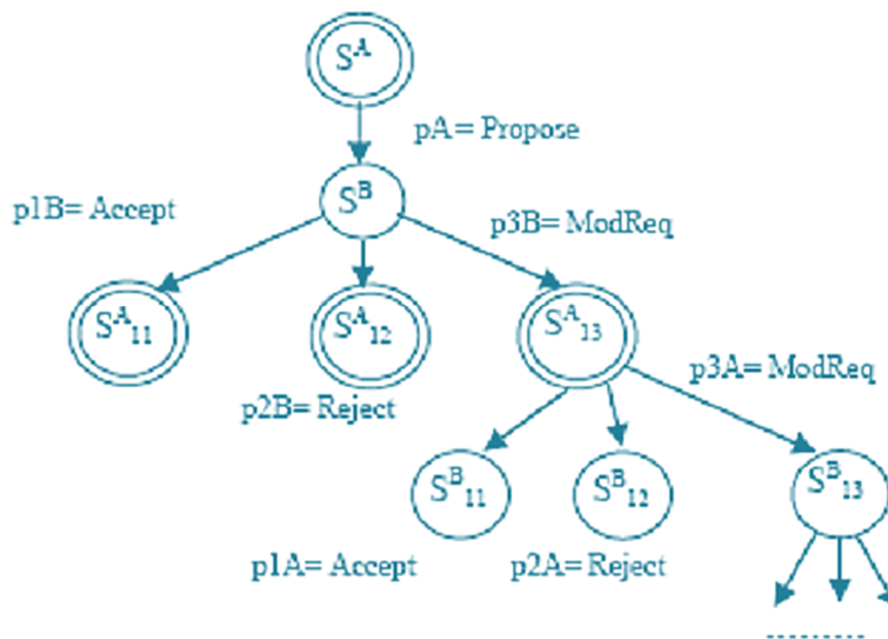


Figure 4.8 : Négociation heuristique

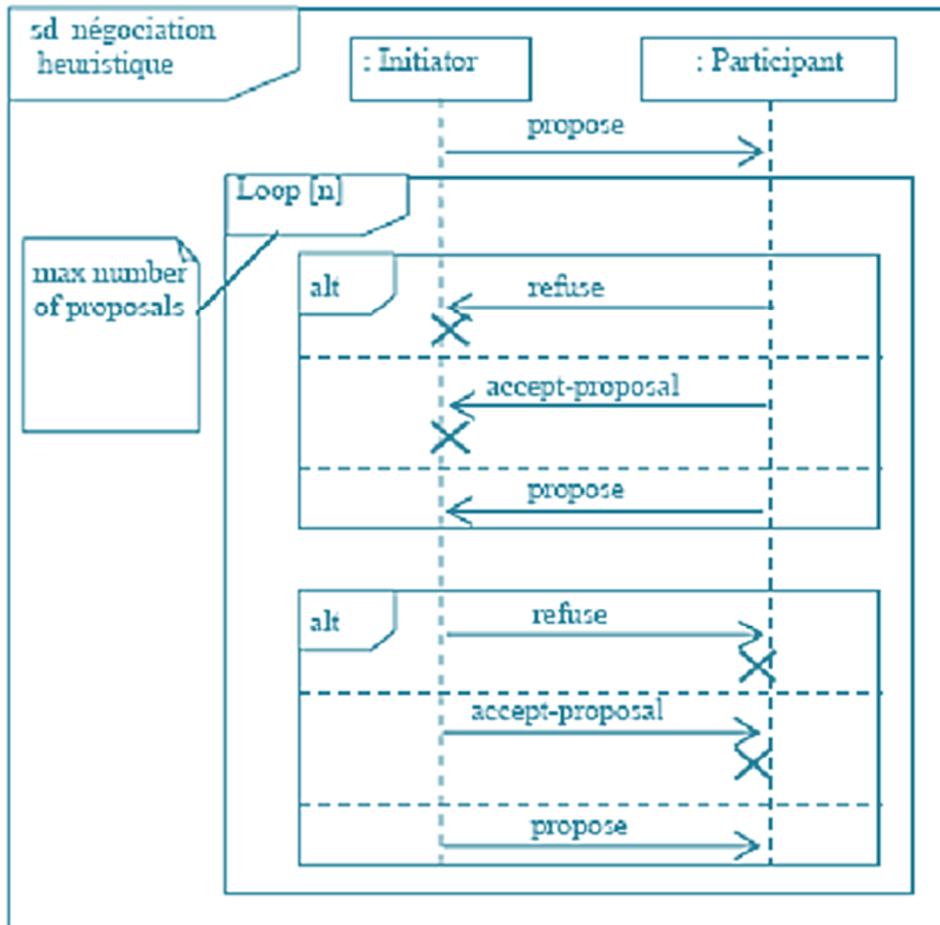


Figure 4.9 : Négociation heuristique

5 Modélisation des comportements des agents

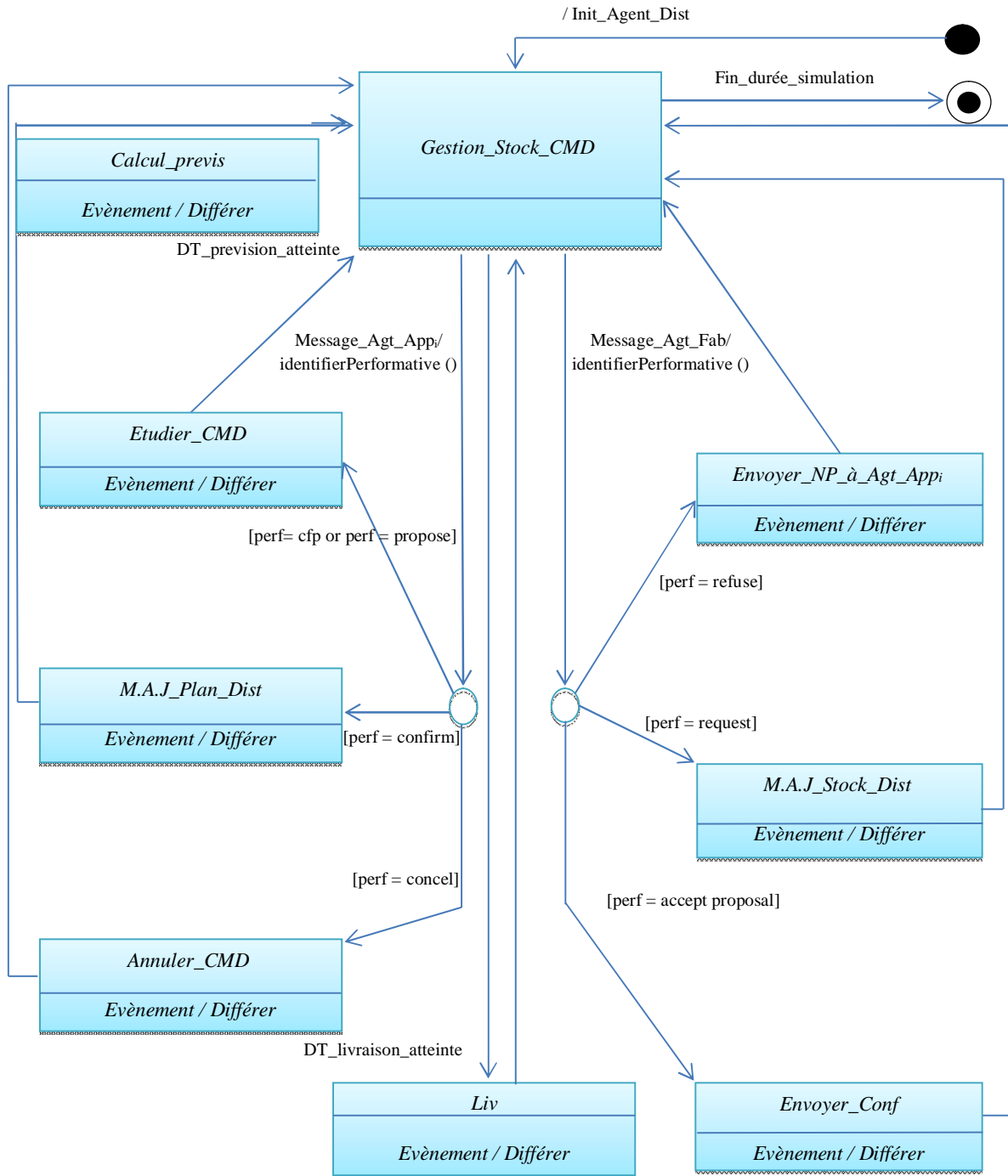
Les sections précédentes ont conduit à l'énumération des agents constituant le système et la modélisation de quelques protocoles de négociation. Dans cette section, nous modélisons le comportement de chaque agent par un diagramme d'états-transitions AUML. Les agents passent d'un état donné à un autre en fonction des actions se produisant dans l'environnement ou en fonction des messages reçus. Les diagrammes d'états (statechart diagram) AUML semblent être une solution efficace pour décrire leurs comportements[162].

5.1 Comportement des agents

Les négociations entre les agents se font à base de la communication par messages asynchrones. Un message asynchrone n'interrompt pas l'exécution de l'expéditeur. L'expéditeur envoie le message sans savoir quand, ni même si le message sera traité par les destinataires.

5.1.1 Comportement de l'agent « Dist »

Le comportement de l'agent « Dist » dépend du type de l'acteur, qui représente selon notre travail une entreprise possédant une activité de production (figure 4.10), dans ce qui suit une description des états de l'agent « Dist » est présentée :



Evènement : représente l'ensemble des évènements provoquant la transition de l'état « Gestion_stock_CMD » vers un autre état : Message_Agt_Fab, Message_Agt_Appi, Fin_durée_simulation, DT_livraison_atteinte, DT_livraison_atteinte

Figure 4.10 : Diagramme d'états-transitions de l'agent « Dist ».

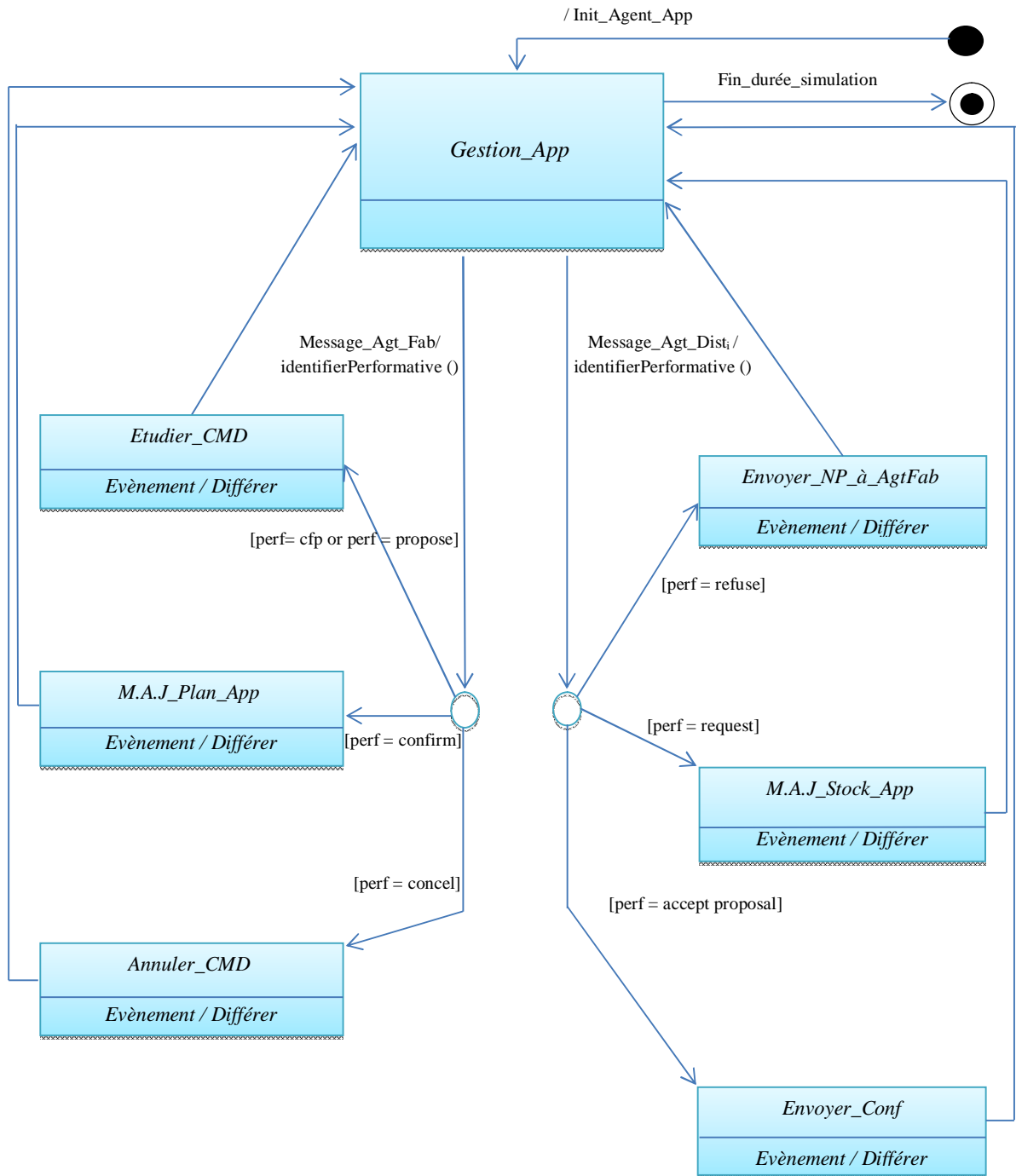
-
- **Gestion_Stock_CMD** : Lorsque l'agent est dans cet état, il gère le stock de produits finis (stock de distribution) en utilisant une méthode de gestion de stock. Par exemple, le système à point de commande, le Système à re-complètement périodique, etc. Il maintient aussi un plan de livraison des commandes confirmées. Selon la méthode utilisée pour la gestion du stock, des événements internes peuvent être déclenchés. Par exemple la période de révision est atteinte (dans le cas d'un système à re-complètement périodique). L'action devant être exécutée suite au déclenchement de ce type d'événement est l'envoi d'un message de performative égale à « REQUEST » à l'agent « Fab » dans le cas d'une usine. Au même temps, il attend l'arrivée d'une commande provenant de l'agent « App_i » d'un autre acteur. L'événement « Message_AgentApp_i » montre que l'agent peut traiter plusieurs commandes en même temps provenant d'acteurs différents. Il attend aussi des réponses provenant de l'agent « Fab ».
 - **Etudier_CMD** : Dans cet état, l'agent « Dist » étudie la possibilité de livraison de la commande qui peut être nouvelle ou en cours d'étude. Pour ce faire, il consulte le stock et le programme de livraison, puis communique avec l'agent « Fab » si nécessaire. Il peut répondre à ce moment à l'agent « App_i » ou ultérieurement quand il reçoit la réponse de l'agent « Fab ».
 - **M.A.J_Dist** : Dans cet état, l'agent « Dist » met à jour le plan de livraison pour tenir compte de la commande confirmée par l'agent « App_i ». Eventuellement, il peut envoyer un message de confirmation à l'agent « Fab ».
 - **Annuler_CMD** : Dans cet état, l'agent « Dist » doit annuler la commande confirmée ou se trouvant en cours d'étude.
 - **Envoyer_Conf** : Deux cas de figures sont possibles dans cet état : L'agent « Dist » envoie un message de réponse à l'agent « App_i » en lui indiquant que la livraison de la commande passée est possible et attendra sa confirmation. L'agent « Dist » envoie un message de confirmation à l'agent « Fab » pour commencer la production (par exemple, dans le cas de la mise à jour du stock dans un système à point de commande).
 - **Envoyer_NP_à_AgtApp_i** : Dans cet état, l'agent identifie les raisons pour lesquelles l'agent « Fab » n'a pas accepté la demande d'effectuer l'action de production, il cherche ensuite une nouvelle proposition et l'envoie à l'agent « App_i ».
 - **M.A.J_Stock** : Dans cet état, l'agent « Dist » met à jour le stock après la fin de la production.
 - **Liv** : Dans cet état, l'agent livre la commande programmée dans le plan de livraison.

- **Calcul_Prévis**: Dans cet état, l'agent calcule les prévisions et met à jour les plans nécessaires.

5.1.2 Comportement de l'agent « App »

La figure 4.1 illustre le comportement de l'agent « App ». Les différents états de l'agent sont illustrés dans ce qui suit :

- **Gestion_App** : L'agent gère le stock d'approvisionnement de chaque matière première.
- **Etudier_CMD** : Dans cet état, l'agent vérifie si les quantités demandées des matières premières sont disponibles ou pourrait l'être.
- **M.A.J_Plan_App** : L'agent met à jour le plan d'approvisionnement pour tenir compte de la commande confirmée.
- **Annuler_CMD** : Dans cet état, l'agent doit annuler la commande confirmée ou se trouvant en cours d'étude.
- **Envoyer_Conf** : Deux cas de figures sont possibles dans cet état : L'agent « App » envoie un message de réponse à l'agent « Fab » en lui indiquant que les quantités demandées des matières premières sont disponibles ou elles le seront. L'agent envoie un message de confirmation d'une commande d'approvisionnement à l'agent « Dist_i » d'un autre acteur (entreprise).
- **Envoyer_NP_à_AgtFab** : L'agent envoie une nouvelle proposition à l'agent « Fab ».
- **M.A.J_Stock_App** : L'agent reçoit les matières premières, il met à jour le stock d'approvisionnement. Ceci dépend bien sûr des méthodes utilisées pour la gestion du stock de chaque matière première.

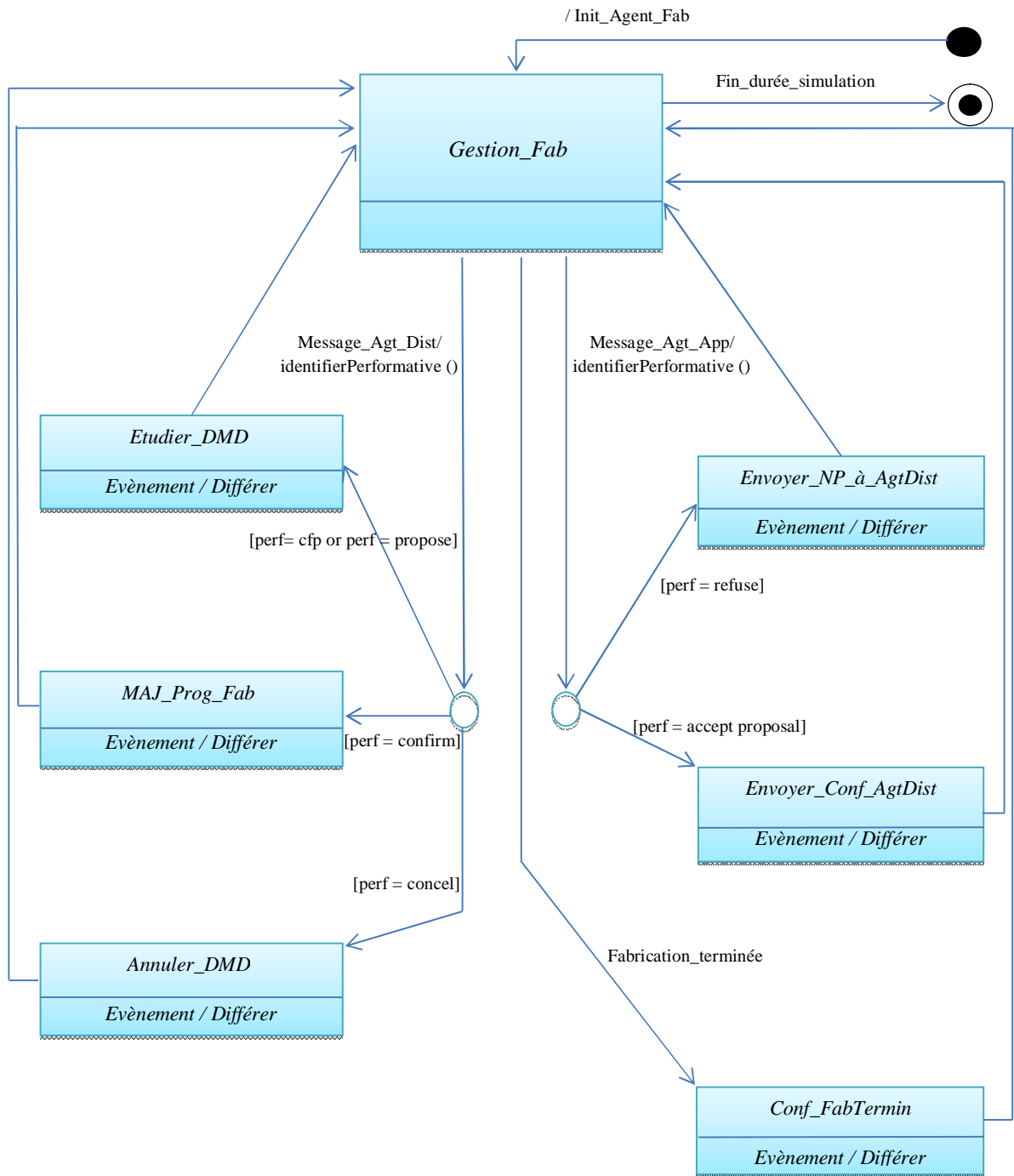


Evènement : représente l'ensemble des évènements provoquant la transition de l'état « Gestion_App » vers un autre état : Message_Agt_Fab, Message_Agt_Disti, Fin_durée_simulation.

Figure 4.11 : Diagramme d'états-transitions de l'agent « App »

5.1.3 Comportement de l'agent « Fab »

La figure 4.12 illustre le comportement de l'agent « Fab ».



Evènement : représente l'ensemble des évènements provoquant la transition de l'état « Gestion_App » vers un autre état : Message_Agt_App, Message_Agt_Dist ,Fin_durée_simulation, Fabrication_terminée.

Figure 4.12 : Diagramme d'états-transitions de l'agent « Fab »

Les différents états de l'agent sont illustrés au-dessous :

- **Gestion_Fab** : Lorsque l'agent est dans cet état, il gère un programme de production. En même temps, il attend l'arrivée d'un message provenant de l'agent « Dist » ou l'agent « App ». Ce message peut contenir une nouvelle demande de production ou bien une réponse concernant une commande en cours d'étude.
- **Etudier_DMD** : Dans cet état, l'agent analyse la demande de production puis communique avec l'agent « App » si nécessaire. Il peut répondre à ce moment à l'agent « Dist » ou ultérieurement quand il reçoit la réponse de l'agent « App ».
- **MAJ_Prog_Fab** : Dans cet état, l'agent met à jour le programme de production pour tenir compte de la commande confirmée. Eventuellement, il peut envoyer un message de confirmation à l'agent « App ».
- **Annuler_DMD** : Dans cet état, l'agent annule la production concernant une commande confirmée ou en cours d'étude.
- **Envoyer_conf_AgtDist** : Dans cet état, l'agent confirme à l'agent « Dist » que la production est possible et attend juste sa confirmation pour commencer.
- **Envoyer_NP_à_AgtDist** : Dans cet état, l'agent identifie les raisons pour lesquelles l'agent « App » a refusé, ensuite il envoie une nouvelle proposition à l'agent « Dist ».
- **Conf_FabTermin** : Dans cet état, l'agent envoie un message à l'agent « Dist » pour mettre à jour le stock, car la production de la quantité demandée de produits est terminée.

6 Exemple d'interaction entre les agents pour la prise de commande ferme

Dans le chapitre 2 de ce mémoire, nous présentons le mécanisme de prise de décision pour l'acceptation ou le refus d'une commande ferme, nous présentons aussi différents cas de figure d'occurrences de problèmes liés et qui sont résumées dans le tableau 2.1.

Nous développons dans ce qui suit les mécanismes de propagation des flux d'informations ainsi que les actions prédéterminées entre les agents pour la prise de décision dans le cadre d'une prise de commande ferme (selon le *tableau 2.1* de chapitre 2).

- **Dans le cas 1** : les quantités demandées par le client sont ou seront disponibles aux dates prévues. L'agent Approvisionnement « App » d'un client envoie une demande d'approvisionnement à l'agent distribution « Dist » de l'entreprise, cette demande a les caractéristiques décrites dans le chapitre 2. L'agent « Dist » de l'entreprise transforme alors cette demande sous forme d'une courbe cumulée et la demande est ensuite analysée par cet agent. L'analyse consiste à vérifier la disponibilité en quantité et en délai du produit à fournir.

Dans le cas qui nous concerne, les quantités souhaitées sont disponibles dans le stock de produits finis. Alors l'agent « Dist » peut répondre favorablement à cette demande.

- **Dans le cas 4** : l'entreprise a besoin de composants et de capacités de transport pour la prise d'une commande ferme. L'entreprise reçoit une demande d'approvisionnement de la part d'un client, cette demande est analysée par l'agent « Dist » de l'entreprise. L'analyse vérifie si les stocks de produits finis sont suffisants pour satisfaire pleinement la demande. L'agent « Dist » envoie alors une requête de mise à disposition d'un complément de produits nécessaires vers l'agent fabrication « Fab ».

L'agent « Fab » vérifie la possibilité de placement de la charge induite par cette demande de produits dans son planning de production. Cette étape peut se réaliser par le biais de la re-planification par exemple. Nous supposons dans ce cas que l'entreprise n'a pas de problèmes de placement, mais ne dispose pas des quantités suffisantes en composants pour la réalisation de la production.

L'agent « Fab » envoie alors une demande en composants à son agent « App » qui correspond au moins à la quantité nécessaire pour satisfaire la demande du client. L'agent « App » prend alors en charge la réalisation de scénarios d'approvisionnement ainsi que le contact et la négociation avec les partenaires de l'entreprise pour satisfaire la demande en composants.

Une fois cette demande satisfaite l'agent « App » propage le résultat de ses recherches vers l'agent « Fab ». Celui-ci, à son tour, envoie une réponse à l'agent « Dist ». Dans ce cas, la réponse est favorable, c'est-à-dire que les quantités demandées par le client deviennent disponibles.

7 Conclusion

Ce chapitre est le cœur de notre contribution. Dans un premier temps, nous avons présenté un cadre méthodologique pour délimiter les frontières de la modélisation de la chaîne logistique distribuées, Notre approche consiste en la décomposition d'une entreprise en trois types d'agents cognitifs « App », « Dist », « Fab ». Qui reprend les rôles et les spécificités de chaque structure de l'entreprise. Cette démarche est motivée par la nature même de la chaîne logistique étudiée qui est distribuée. L'ensemble connecté des agents forme l'architecture de l'entreprise modélisée. L'ensemble des entreprises modélisées et connectées forme l'architecture globale de la chaîne logistique.

Chapitre 5

Implémentation Et Validation

1 Introduction.....	120
2 Implémentation.....	120
2.1 La norme FIPA pour les systèmes multi-agents.....	120
2.2 Modèle d'agents JADE et notion de comportement	122
2.2.1 Modèle d'agents.....	122
2.2.2 Critères de choix.....	125
2.3 Des diagrammes d'états-transitions vers l'implémentation des agents.....	127
2.4 Architecture fonctionnelle de JADE et cycle de vie d'un agent.....	130
2.4.1 Architecture de la plate-forme JADE.....	130
2.4.2 Cycle de vie d'un agent.....	131
3 Validation de notre approche.....	133
3.1 Évaluation des performances dans les chaînes logistiques par simulation d'un système multi agent.....	133
3.2 Données prises en compte dans le modèle	134
3.3 Illustration de notre approche de prise d'une demande imprévue	135
3.4 Simulation.....	141
3.5 Résultats des simulations.....	142
4 Conclusion.....	143

1 Introduction

Dans les chapitres précédents, nous avons exposé une démarche d'aide à la décision pour la prise de commande imprévue dans un contexte de chaîne logistique distribuée. A partir de problèmes liés à cette prise de décision, nous développons des modèles capables de traiter des problèmes de disponibilité des matières premières et des produits finis. L'architecture multi agents est généralement la plus fiable à mettre en œuvre pour les infrastructures complexes, comme les architectures distribuées auxquelles nous nous intéressons. En effet, elle garantit une flexibilité du modèle qui prend en compte la modularité du système réel. En outre, ce type d'architecture est efficace lorsqu'une architecture centralisée ne peut être mise en place du fait de l'indépendance des acteurs qui la compose.

Le meilleur moyen pour construire un système multi-agent (SMA) est d'utiliser une plate-forme multi-agent. Une plate-forme multi-agent est un ensemble d'outils nécessaire à la construction et à la mise en service d'agents au sein d'un environnement spécifique. Comme première partie dans ce chapitre nous allons étudier la plate-forme multi-agents JADE (Java Agent DEvelopment framework), Dans la deuxième partie de ce travail, nous proposons un fonctionnement pour ce système ainsi que son utilisation dans le but de l'évaluation des performances de la chaîne logistique par simulation.

2 Implémentation

2.1 La norme FIPA pour les systèmes multi-agents

FIPA (Fondation for Intelligent Physical Agents) est une organisation dont l'objectif est de produire des standards pour l'interopération d'agents logiciels hétérogènes [163] Le premier document de la FIPA, et le FIPA97, établissant les règles normatives qui permettent à une société d'agents d'inter opérer. Les documents FIPA décrivent le modèle de référence d'une plate-forme multi-agents (*Figure 5.1*) où ils identifient les rôles de quelques agents clés nécessaires pour la gestion de la plateforme, et spécifient le contenu du langage de gestion des agent et l'ontologie du langage. Dans la *figure 5.1* on voit qu'il y a trois rôles (agent) principaux identifiés dans une plate-forme d'agent (en plus des rôles optionnels) [164] [165] :

- **Le Système de gestion d'Agent** (AMS- Agent Management System) Agent qui exerce le contrôle de supervision sur l'accès et l'usage de la plateforme ; il est responsable d'authentifier les agents résidents et de contrôler les enregistrements.
- **Le Canal de Communication** (ACC- Agent Communication Canal) Agent qui fournit le chemin pour les interactions de base entre les agents dans et en d'hors de la plateforme ; c'est la méthode de communication implicite qui offre un service fiable et précis pour le routage des messages ; il (l'agent) doit aussi être compatible avec le protocole IIOP pour assurer l'interopérabilité entre les différentes

plateformes.

- **Le Facilitateur d'Annuaire** (Directory Facilitator 'DF') est l'agent qui fournit un service de pages jaunes à la plate-forme multi-agent.

Le standard spécifie aussi le Langage de Communication d'Agents (Agent Communication Language 'ACL'). La communication des agents est basée sur l'envoi de messages. Le langage FIPA-ACL est le langage standard des messages et impose le codage, la sémantique et la pragmatique des messages. La norme n'impose pas de mécanisme spécifique pour le transport interne des messages. Plutôt, puisque les agents différents pourraient s'exécuter sur des plates-formes différentes et utiliser des technologies différentes d'interconnexion, FIPA spécifie que les messages transportés entre les plates-formes devraient être codés sous forme textuelle. On suppose que l'agent est en mesure de transmettre cette forme textuelle. La norme FIPA préconise des formes communes pour les conversations entre agents par la spécification de protocoles d'interaction, qui incluent des protocoles simples de type requête-réponse, mais aussi des protocoles spécifiques aux agents.

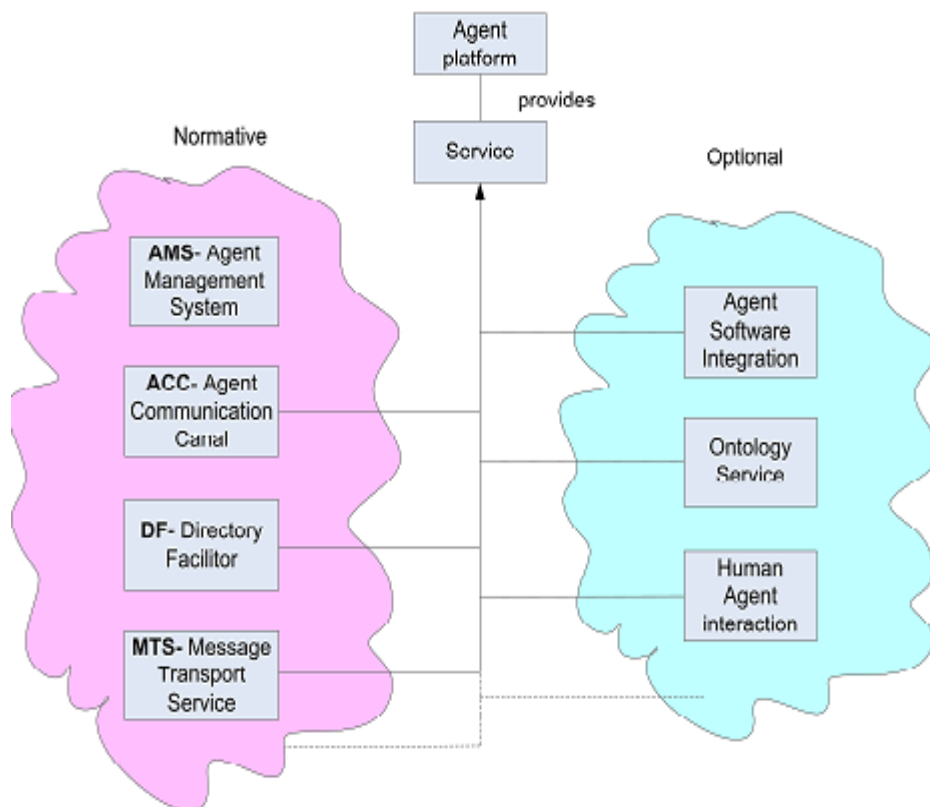


Figure 5.1: Services fournis par la Plate-forme FIPA.

2.2 Modèle d'agents JADE et notion de comportement

JADE est une plateforme multi-agent développée en Java par CSELT (Groupe de recherche de Gruppo Telecom, Italie) qui a comme but la construction des systèmes multi agent et la réalisation d'applications conformes à la norme FIPA. JADE comprend deux composantes de base : une plateforme agents compatible FIPA et un paquet logiciel pour le développement des agents Java.

Dans cette section, nous présentons en détail le modèle d'agents dans la plateforme JADE afin de montrer quelques-unes de ses performances et ses caractéristiques. Ses dernières sont qualifiées comme importantes dans le choix de cette plateforme pour la phase d'implémentation de notre architecture distribuée.

2.2.1 Modèle d'agents

On a vu qu'une propriété importante d'un agent est son autonomie : un agent ne doit pas se limiter à réagir aux événements externes, mais il doit être aussi capable de prendre l'initiative de nouveaux actes communicatifs d'une façon autonome. Ceci exige que chaque agent ait un thread interne de contrôle ; cependant, un agent peut engager des conversations simultanées multiples, tout en poursuivant d'autres activités qui n'impliquent pas d'échanges de messages.

JADE utilise l'abstraction Comportement (Behaviour) pour modéliser les tâches qu'un agent peut exécuter. Les agents instancient leurs comportements selon leurs besoins et leurs capacités. Du point de vue de la programmation concurrente, un agent est un objet actif, ayant un thread de contrôle. JADE utilise un modèle de programmation concurrente "un thread-par-agent" au lieu d'un modèle "un thread-par-comportement" pour éviter une augmentation du nombre de threads d'exécution exigés sur la plateforme d'agents. Ceci signifie que, pendant que les agents différents s'exécutent dans un environnement multi-thread de préemption, deux comportements d'un même agent sont planifiés coopérativement.

En dehors de la préemption, les comportements travaillent tous comme des threads d'exécution coopératifs, mais il n'y a pas de pile qui ait besoin d'être sauvée. Un planificateur (scheduler), exécuté par la classe de base Agent et caché au programmeur, exécute une politique de « round-robin » de non-préemption entre tous les comportements disponibles dans la file des processus prêts. Ainsi, il permet l'exécution d'une classe dérivée de la classe Comportement jusqu'à ce qu'elle abandonne le contrôle d'exécution par elle-même. Si la tâche qui a le contrôle n'est pas encore finie, elle sera re-planifiée pendant le prochain tour du « round-robin » à moins qu'elle ne soit pas bloquée ; en fait, un comportement peut se bloquer lui-même, par exemple pendant qu'il attend des messages, pour éviter le gaspillage de temps de CPU, réalisant ainsi un comportement d'attente occupée. La *figure 5.2* résume les étapes d'exécution d'un thread d'agent. Donc, le développeur d'agents doit étendre la classe Agent et implémenter les tâches spécifiques de l'agent par une ou plusieurs classes Comportement, les instancier et les ajouter à l'agent en utilisant la méthode *addBehaviour()*.

JADE inclut quelques comportements prêts à être utilisés pour les tâches les plus communes dans la programmation des agents, tels que l'envoi et la réception des messages et la décomposition des tâches complexes en des agrégations de tâches plus simples. La *figure 5.3* montre la hiérarchie de quelques classes de comportement JADE. Par exemple, la classe *OneShotBehaviour* permet d'instancier un comportement qui se termine juste après sa première exécution. Ainsi, sa méthode *action()* qui représente la tâche à effectuer est exécutée une seule fois. La méthode *done()* de cette classe est déclarée finale (i.e. ne pouvant pas être redéfinie) et retourne toujours vrai (*true*). La classe *CyclicBehaviour* permet d'instancier un comportement qui ne se termine pas. Par conséquent, sa méthode *action()* exécute les mêmes opérations chaque fois où est appelée jusqu'à ce que le thread interne de l'agent exécutant ledit comportement se termine. La méthode *done()* de cette classe est déclarée finale et retourne toujours *false*.

La classe *Agent* représente une super-classe commune pour tous les agents définis par l'utilisateur. Du point de vue du programmeur, la conséquence est qu'un agent JADE est une classe Java qui étend la classe de base *Agent*. Cela permet à l'agent d'hériter un comportement fondamental caché (qui traite toutes les tâches liées à la plate-forme, telles que l'enregistrement, la configuration, la gestion à distance, etc.), et un ensemble de méthodes qui peuvent être appelées pour implémenter les tâches spécifiques à l'agent, par exemple envoi des messages, utilisation des protocoles d'interaction standard, enregistrement sur plusieurs domaines, etc. Entre autres, JADE offre aussi une classe *JessBehaviour* qui permet l'intégration avec le système expert JESS (Java Expert SystemShell), où JADE fournit le noyau de l'agent et garantit (autant que possible) la conformité avec les normes FIPA, alors que JESS est le moteur d'inférence de l'agent qui exécute le raisonnement nécessaire pour la résolution du problème.

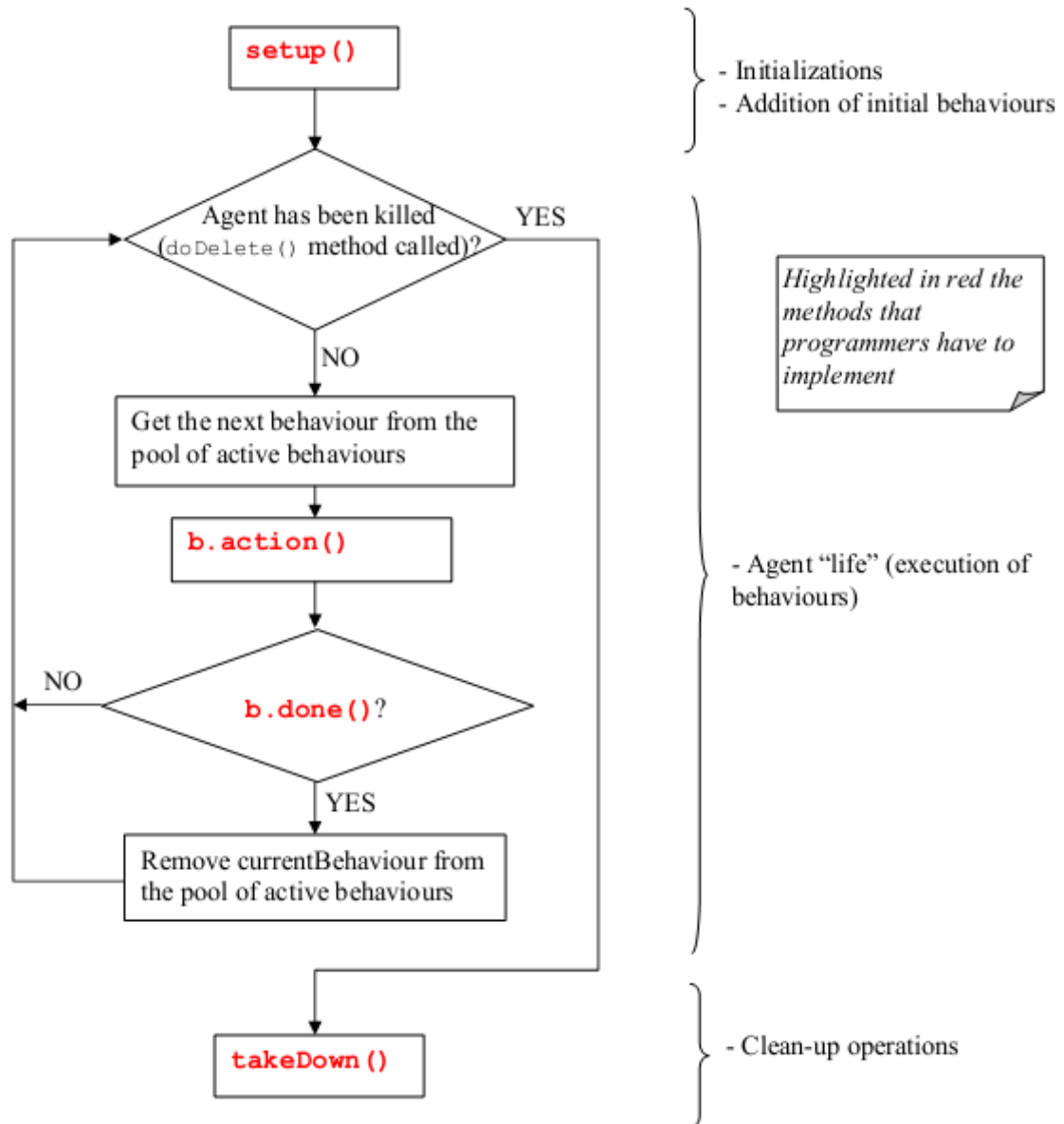


Figure 5.2 : Etapes d'exécution d'un thread d'agent.

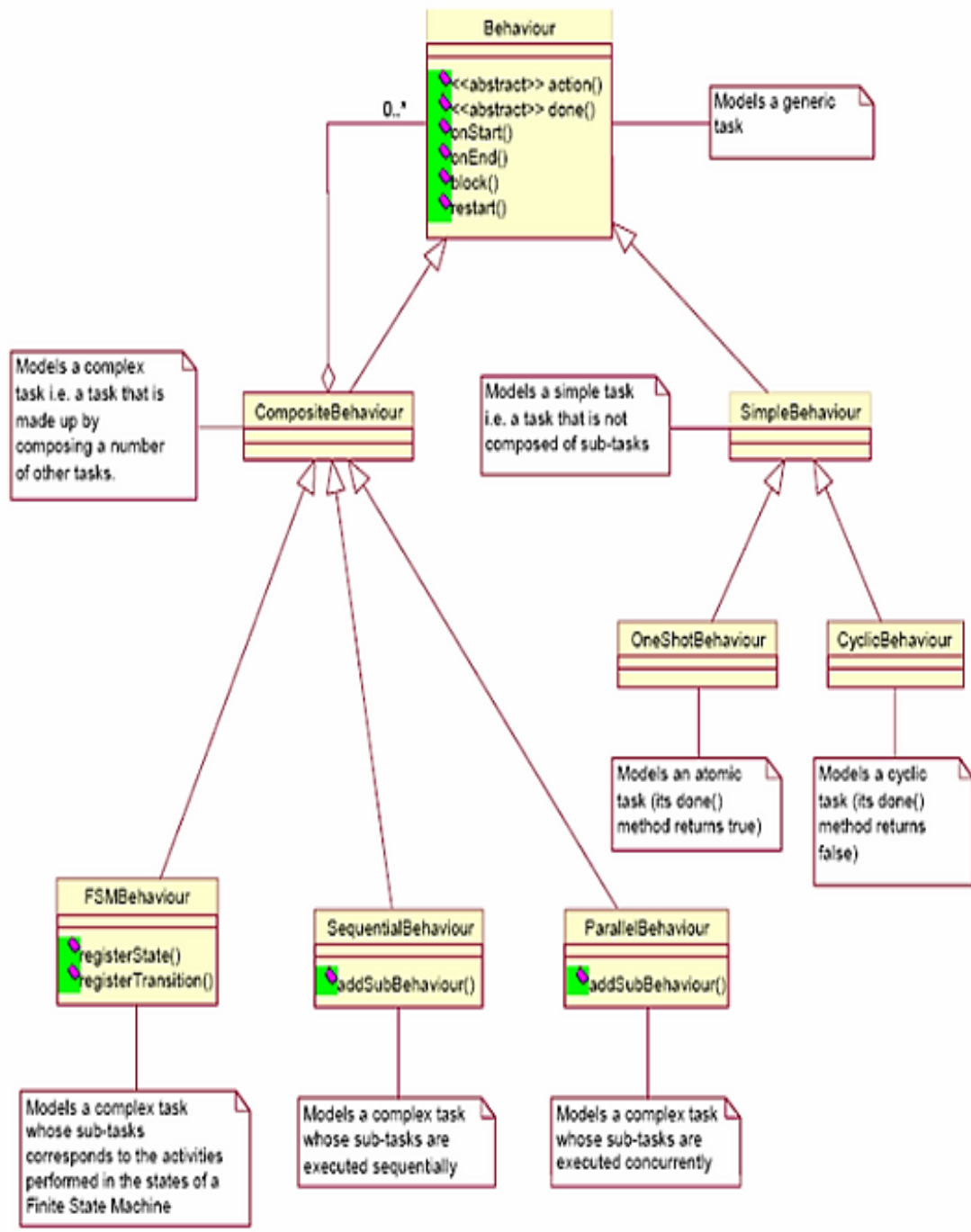


Figure 5.3 : Hiérarchie de quelques classes Behaviour de la plateforme JADE.

2.2.2 Critères de choix

Après avoir présenté le modèle d’agent JADE et la notion de comportement, nous répondons dans cette section à la question :

Pourquoi nous avons choisi la plateforme JADE ?

Nous avons déjà présenté dans le chapitre 3 quelques caractéristiques de la plateforme JADE qui nous ont conduits à la choisir pour la phase d'implémentation. Par rapport à notre problème, JADE possède des caractéristiques d'ordre technique qui ont renforcé notre choix. L'une des caractéristiques principales est l'abstraction Comportement (Behaviour) présentée dans la section précédente. Elle permet d'implémenter les tâches qu'un agent peut exécuter. La notion de Comportement se révèle très importante au moins pour deux raisons :

Elle permet d'implémenter de manière efficace les protocoles d'interaction entre les agents. Ceci est très important dans notre cas pour implémenter les différents protocoles de négociation modélisés par AUML dans le chapitre 4. A titre d'exemple, les classes prédéfinies `ProposeInitiator` et `ProposeResponder` de JADE permettent d'instancier des objets behaviours qui implémentent le protocole d'interaction FIPA-Propose modélisé par le diagramme de séquence AUML de la (figure 5.4). JADE fournit d'autres classes de comportements prêts pour implémenter les protocoles d'interactions FIPA comme FIPA Propose Protocol, FIPA Request Protocol, FIPA Contract Net Protocol, FIPA English Auction Protocol [166]. Ces classes se trouvent dans le package `jade.proto`.

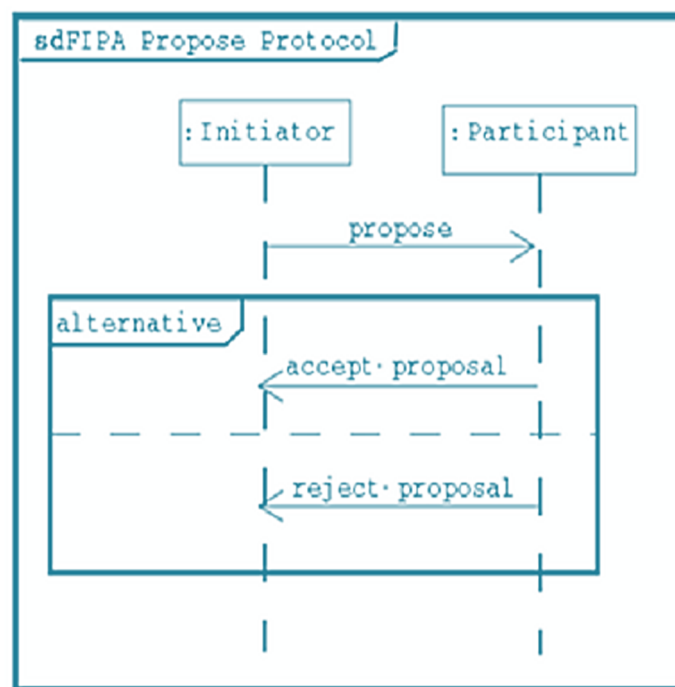


Figure 5.4 : Protocole d'interaction FIPA Propose

La notion de comportement nous permet de générer rapidement le squelette de code d'un agent JADE à partir du diagramme d'états-transitions qui modélise son comportement. Comme nous montrerons dans la section suivante de ce chapitre, nous allons présenter dans ce qui suit le travail qui permet le passage de diagramme d'état transition à l'implémentation des agents.

2.3 Des diagrammes d'états-transitions vers l'implémentation des agents

Dans cette section, nous proposons le passage de la spécification à l'implémentation. Ce passage se présente sous la forme d'un ensemble de transformations entre un diagramme d'états-transitions, modèle de comportement d'un agent et une classe d'agent, conformément à la structure proposée par JADE pour une telle classe.

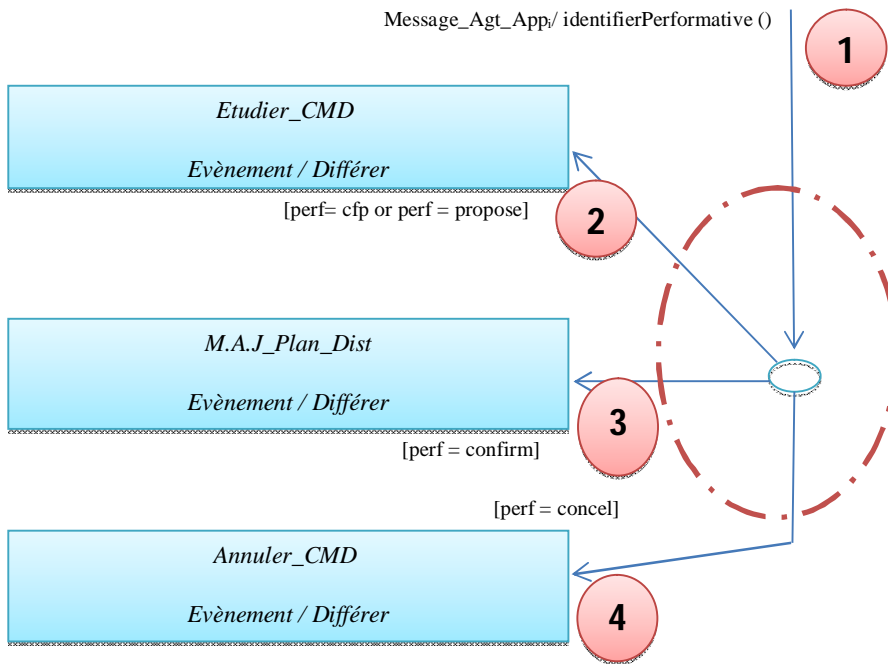
La *figure 5.2* présente les étapes d'exécution d'un thread d'agent JADE. En fait, la *figure 5.2* peut être enrichie par d'autres étapes. Comme le montre cette figure, chaque agent JADE possède une méthode *setup()*. Elle est destinée à contenir les ordres d'initialisations de l'agent et d'ajout de comportements. Il est indispensable d'ajouter au moins un objet comportement (Behaviour) à l'agent, pour qu'il soit capable de faire quoi que ce soit. L'ajout d'un nouveau comportement se fait par la méthode *addBehaviour(Behaviour b)*. La méthode *takeDown()* est exécutée juste avant la terminaison définitive de l'agent.

Chaque objet comportement (behaviour) possède entre autres les méthodes *onStart()*, *onEnd()* et *action()*. La méthode *onStart()* est exécutée une seule fois avant de commencer l'exécution du comportement, c'est-à-dire avant l'exécution de la méthode *action()* qui représente l'activité à exécuter par le comportement. La méthode *onEnd()* est exécutée une seule fois après la fin d'exécution du comportement.

Dans la suite de cette section, nous montrons comment appliquer ces étapes d'exécution d'un thread d'agent JADE sur un exemple d'agents de notre système. Pour cela, nous nous référons au diagramme d'états-transitions de l'agent « Dist » (cf. *chapitre 4, Figure 4.6*). La classe Java de l'agent « Dist » est obtenue à partir de son modèle de comportement, sa structure partielle est présentée dans la *Figure 5.5*.

On va donner dans ce qui suit une petite explication sur le code généré dans la *Figure 5.5*. Pour le Numéro 1 : A la réception du message « Message_Agt_App_i » provenant d'un agent « App » d'un acteur « i », l'agent « Dist » doit identifier sa performative (perf = propose, perf = confirm, perf = cancel, etc). La valeur de cette performative déterminera alors si l'état destination est « Etudier_CMD » ou « M.A.J_Plan_Dist » ou bien « Annuler_CMD ». Ici, on crée une classe interne qui sera utilisée pour connaître si l'émetteur d'un message reçu est un agent « App ».

Pour les numéros 2,3 et 4 : A chaque performative, on associe un objet *MessageTemplate* permettant la sélection de l'état destination de l'agent, Nous notons bien que cette *Template* utilise l'opérateur AND pour coupler deux conditions le type de la performative et l'émetteur du message.



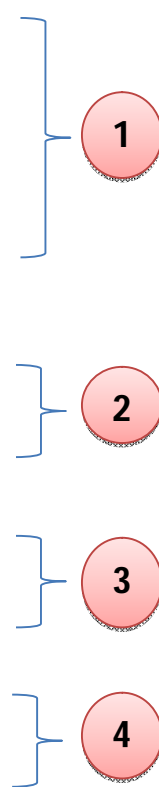
```

publicclass Dist extends Agent {
    //-----
    privateclass MatchAgentName implements MessageTemplate.MatchExpression {
        publicboolean match(ACLMessage msg) {
            String Agname = msg.getSender().getLocalName();
            return( Agname.startsWith("A"));
        } //dans notre cas ce code retourne vrai si le nom de l'agent
        //correspond à un agent App d'un autre acteur.
    } // END of inner class MatchAgentName

    private MessageTemplate EtudierCMD = MessageTemplate.and(
        MessageTemplate.MatchPerformative(ACLMessage.CFP),
        new MessageTemplate(new MatchAgentName()));

    private MessageTemplate MAJPlanDist = MessageTemplate.and(
        MessageTemplate.MatchPerformative(ACLMessage.CONFIRM),
        new MessageTemplate(new MatchAgentName()));

    private MessageTemplate AnnulerCMD = MessageTemplate.and(
        MessageTemplate.MatchPerformative(ACLMessage.CANCEL),
        new MessageTemplate(new MatchAgentName ()));
    
```



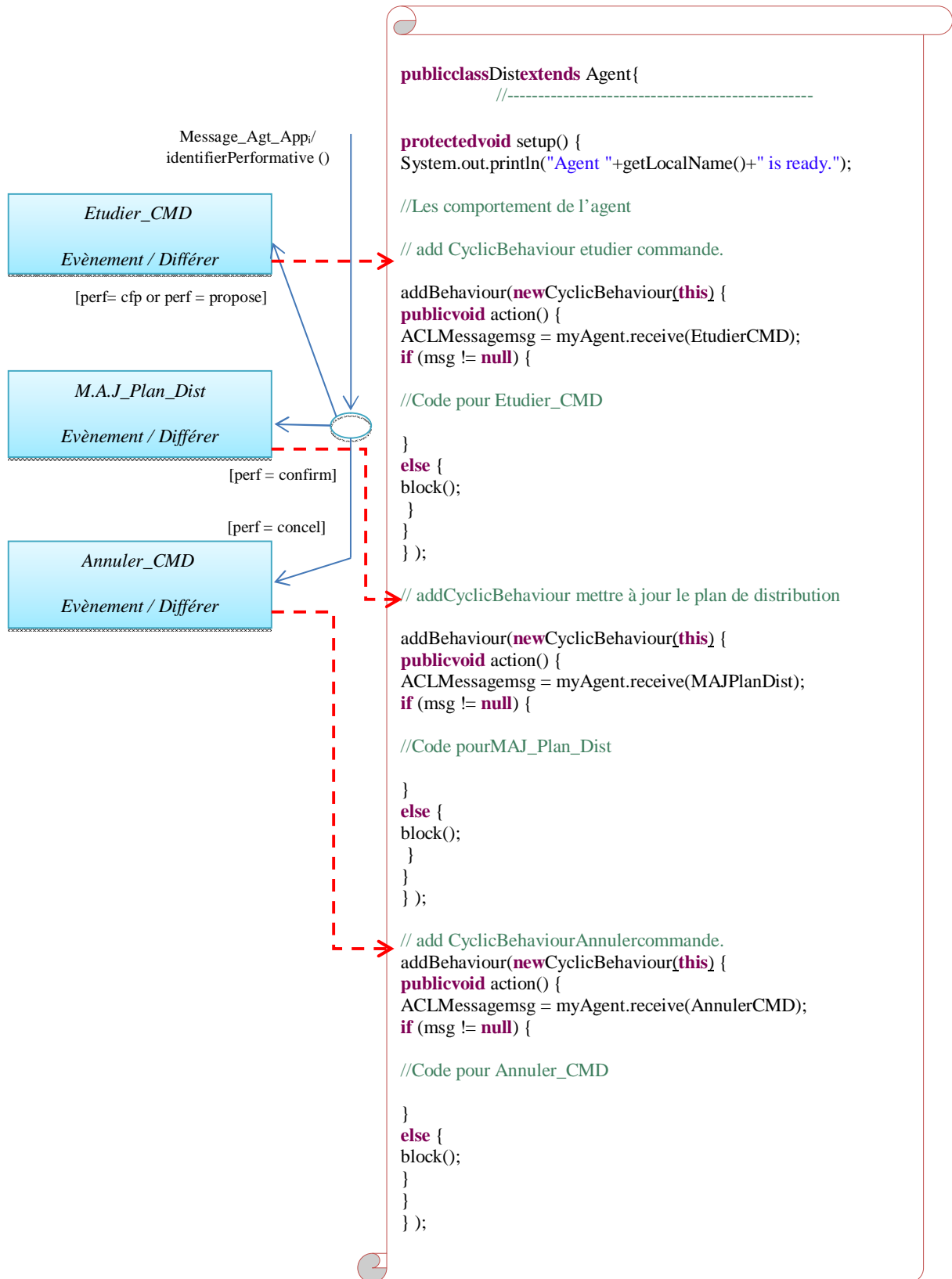


Figure 5.5 : Transformation partielle du modèle de comportement de l'agent « Dist » vers le code Java (Agent JADE).

Dans la deuxième partie de la même *figure 5.5*, pour l'application des règles de transformation, A chaque état, on associe un objet comportement, Puisque l'évènement représenté par un message qui arrive d'un agent approvisionnement d'un autre acteur « *Message Agt_App_i* » est différé dans les trois états « *Etudier_CMD* », « *MAJ_Plan_Dsit* » et « *Annuler_CMD* », alors chaqu'un de ces derniers va correspondre à un comportement (*Behaviour*) de type *CyclicBehaviour*. Ce comportement est bloquant que le message qui le rend actif est absent. Par exemple, l'agent passera à l'état « *MAJ_Plan_Dist* » lorsqu'un message de performative « *CONFIRM* » provenant d'un agent « *App_i* » soit présent. Ceci est traduit par l'instruction : `ACLMessage msg = myAgent.receive(MAJPlanDist);` Le paramètre *MAJPlanDist* est déjà défini plus haut dans la *figure 5.5*

2.4 Architecture fonctionnelle de JADE et cycle de vie d'un agent

Dans cette section, nous allons présenter l'architecture fonctionnelle de JADE et le cycle de vie d'un agent JADE.

2.4.1 Architecture de la plate-forme JADE

L'architecture du logiciel est basée sur la coexistence de plusieurs Machines Virtuelles (VM)Java et la communication se fait par la méthode RMI (Remote Method Invocation) de Java entre machines virtuelles (VMs) différentes. Chaque VM est un réceptacle d'agents qui fournit un environnement d'exécution complet pour l'exécution des agents et permet d'avoir plusieurs agents qui s'exécutent simultanément sur un même hôte. Chaque réceptacle d'agents est un environnement multi-thread d'exécution composé d'un thread d'exécution pour chaque agent et en plus, des threads créés à l'exécution par le système RMI pour envoyer des messages. Un récipient (container) spécial joue le rôle du frontal de la plate-forme (main-container); il contient les agents de gestion et représente la plate-forme toute entière pour le monde extérieur.

Une plate-forme multi-agent JADE est alors composée de plusieurs réceptacles d'agents selon la (*Figure 5.6*). La distribution de ces réceptacles à travers un réseau d'ordinateurs est permise, à condition que la communication RMI entre leurs hôtes soit conservée. Chaque réceptacle d'agents est un objet serveur RMI qui gère localement un ensemble d'agents. Il règle le cycle de vie des agents en les créant, les suspendant, les reprenant et les détruisant. En plus, il traite tous les aspects de la communication : répartition des messages ACL reçus, routage des messages selon le champ de destination (: receiver) et dépôt des messages dans les files de messages privées des agents (*Figure 5.7*). Pour les messages vers l'extérieur, le réceptacle d'agents maintient assez d'information pour chercher l'emplacement de l'agent récepteur et pour choisir une méthode de transport convenable pour expédier le message ACL.

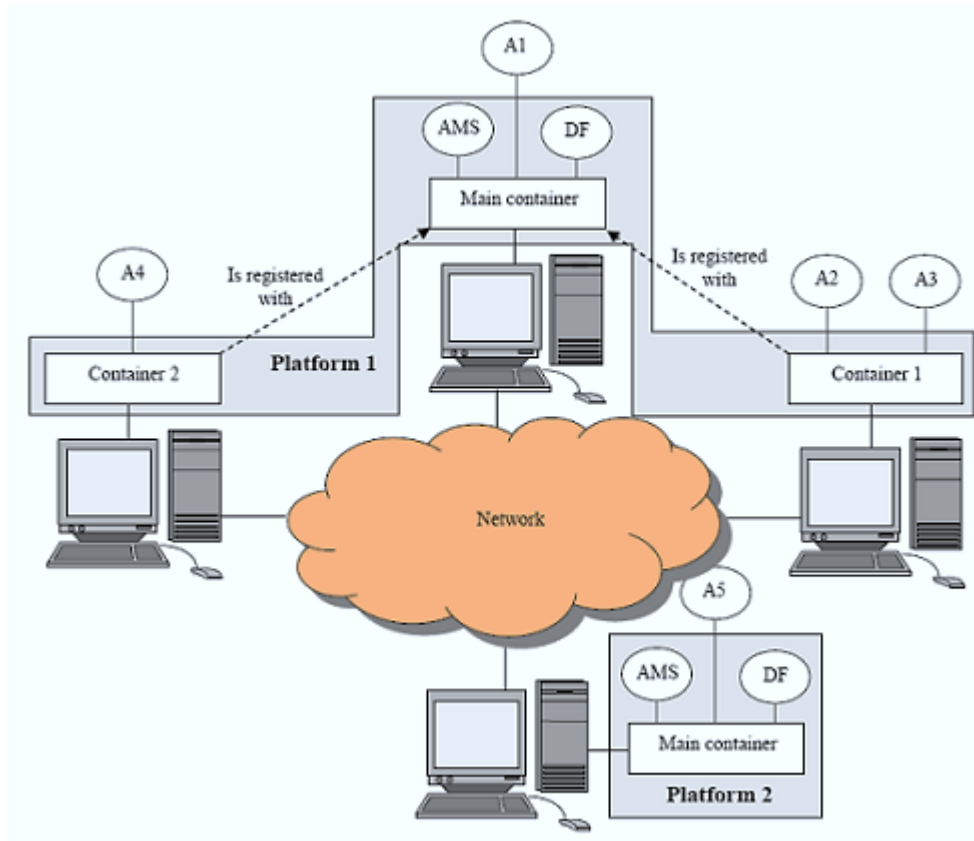


Figure 5.6 : Plateforme et récipiens (container)

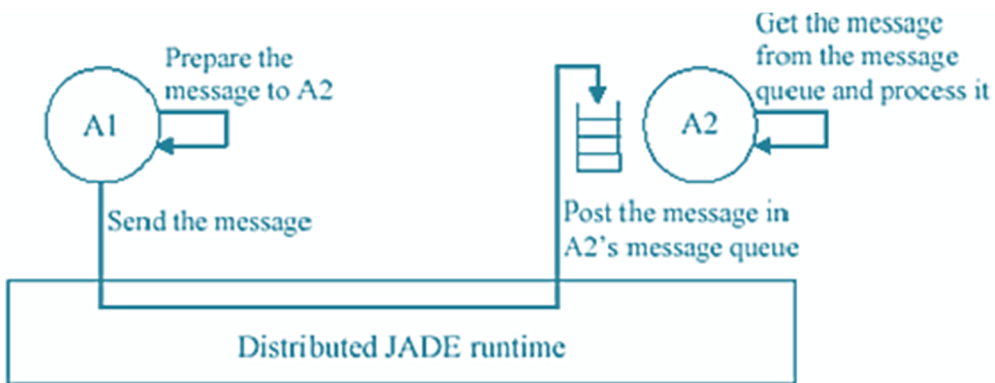


Figure 5.7 : Gestion des messages envoyés et reçus par les agents JADE

2.4.2 Cycle de vie d'un agent

Un agent JADE peut être dans l'une des différents états de son cycle de vie définis par les spécifications FIPA (figure 5.8). Ces états sont représentés par des constantes dans la classe Agent qui figure dans les packages fournis avec la plateforme JADE :

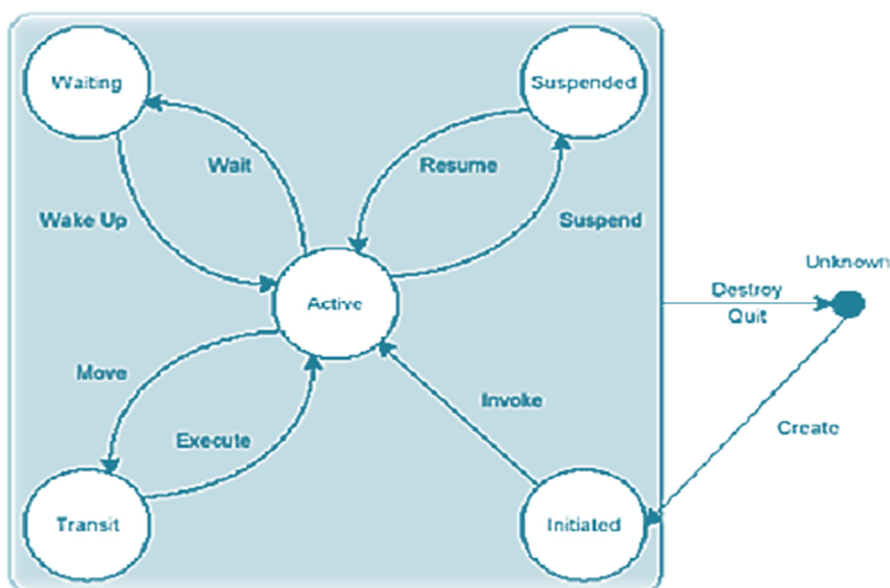


Figure 5.8 : Le cycle de vie d'un agent défini par FIPA

- AP_INITIATED : l'objet Agent est construit, mais il n'est pas encore enregistré avec l'AMS. Il ne possède ni un identificateur ni une adresse et ne peut pas communiquer avec d'autres agents.
- AP_ACTIVE : l'objet Agent est enregistré avec l'AMS, il possède un nom correct et peut accéder à toutes les caractéristiques du JADE.
- AP_SUSPENDED : l'objet Agent est actuellement stoppé. Son thread interne est suspendu et aucun comportement (behaviour) ne peut s'exécuter.
- AP_WAITING : l'objet Agent est bloqué. Son thread interne est en sommeil. Il peut se réveiller quand une condition est vérifiée (typiquement quand un message arrive).
- AP_DELETED : l'exécution du thread interne de l'agent est terminée. L'agent n'est plus enregistré avec l'AMS.
- AP_TRANSIT : un agent mobile entre dans cet état quand il migre à un nouvel emplacement. Le système continue à mémoriser les messages afin de les lui envoyer au nouvel emplacement.
- AP_COPY : cet état est utilisé intérieurement par le JADE pour cloner un agent.
- AP_GONE : cet état est utilisé intérieurement par le JADE quand un agent mobile a migré vers un nouvel emplacement et a pris un état stable.

La classe Agent fournie avec le JADE possède des méthodes publiques qui permettent de

changer l'état d'un objet Agent. Par exemple, la méthode `doWait ()` permet de passer un agent de l'état `AP_ACTIVE` à l'état `AP_WAITING`.

3 Validation de notre approche

Comme nous voyons tout au long de ce mémoire, dans la problématique du management des chaînes logistiques distribuées plusieurs approches sont développées pour assurer la coordination des décisions qui se produisent à travers la chaîne logistique. Cette coordination doit permettre une coopération efficace au sens où elle permet d'assurer la bonne marche des activités entre les partenaires de la chaîne logistique.

La mise en place de ce genre de processus de coordination implique la nécessité de connaître leur l'impact, qu'il soit positif ou négatif, sur la performance globale de la chaîne logistique distribuée. On parle alors d'évaluation de ces performances.

L'enjeu de cette évaluation est de tester l'efficacité du processus de prise de décision que nous avons présenté dans le chapitre 2. Pour mesurer la performance de la négociation dans un tel système, nous proposons un outil de simulation de la chaîne logistique. L'évaluation des performances que nous développons utilise des leviers d'action bien spécifiques. Ces leviers d'actions sont relatifs au nombre de propositions et de scénarios traités conjointement et au nombre des commandes livrées en retard. Les mesures attendues sont relatives aux flux d'informations générés et à la réussite de la coordination mise en place.

3.1 Évaluation des performances dans les chaînes logistiques par simulation d'un système multi agent

La gestion des chaînes logistiques nécessite d'avoir une vision globale sur les performances du système de gestion mis en œuvre. Donc, l'évaluation des performances est préalable à toute application de tels systèmes. Nous présentons dans ce qui suit quelques travaux sur l'évaluation des performances dans le domaine du management des chaînes logistiques qui se fondent sur une simulation d'un système multi agent.

Dans [167], les auteurs proposent une contribution à la réalisation un outil logiciel (Multi-Agents), MATMS (Multi-Agent Tool Management System), qui permet le pilotage d'un réseau d'entreprises. Pour cela, ils proposent un outil de stratégie de management flexible. Les fonctions de cet outil sont fondées sur une méthode de synthèse basée sur un modèle en logique floue. Cet outil permet l'évaluation économique des stratégies de management.

[168], proposent un modèle de sélection de fournisseurs capables d'approvisionner

une entreprise de manière à satisfaire les demandes. L'enjeu de ce modèle est la réduction du nombre de fournisseurs tout en gardant le même niveau d'efficacité dans l'approvisionnement. Pour cela, la méthode proposée utilise l'évaluation des performances des fournisseurs afin de réaliser la sélection.

[169], présente une approche d'évaluation des performances se concentrant spécifiquement sur la collaboration des décisions d'inventaire entre un fournisseur et un distributeur dans une chaîne logistique à deux niveaux. Une comparaison est faite entre un système qui ne dispose pas de mécanismes collaboratifs et un autre système qui en dispose. Les résultats des simulations montrent que la collaboration accroît de manière significative la performance du distributeur.

L'évaluation des performances dans les chaînes logistiques prend de plus en plus d'importance aussi bien dans la conception de la relation interentreprises que pour le dimensionnement des paramètres de pilotage des entreprises.

Comme nous venons de le voir, l'évaluation des performances de la coordination dans une chaîne logistique est peu étudiée. Dans cette partie, nous proposons une démarche par simulation pour l'évaluation d'un système de coordination par négociation d'une chaîne logistique distribuée qui s'appuie sur l'architecture multi agents présentée dans ce chapitre.

Dans le contexte de notre travail, les demandes étudiées sont relatives à la prise de commande ferme et plus particulièrement à celles qui divergent par rapport au planning préalablement établi par le processus prévisionnel à moyen terme.

Cette divergence est relative soit à une modification de commande prévisionnelle en termes de quantité ou de délai, soit à l'occurrence d'une commande non planifiée et à caractère urgent. Bien que marginaux, ces aléas sont la cause majeure des dégradations des performances de la chaîne logistique, car ce sont eux qui engendrent l'effet bullwhip [50].

3.2 Données prises en compte dans le modèle

Pour pouvoir mettre en œuvre une analyse du management de la prise de commande ferme dans la chaîne logistique, la prise en compte de plusieurs paramètres est importante. L'un des plus essentiels est lié à l'architecture industrielle. Elle nécessite de prendre en compte la taille du réseau d'entreprises et le mode de fonctionnement interne des entreprises. Ce dernier peut être de plusieurs types, on différencie les entreprises qui produisent sur stock ou à la commande. La production peut être aussi en mono ou multi produit pour cette chaîne logistique mais chaque entreprise est autonome dans ses prises de décision.

Enfin, il faut aussi prendre en considération le type d'approvisionnement et plus particulièrement les stratégies avec un seul ou plusieurs fournisseurs.

Lorsqu'il se produit une divergence entre deux partenaires d'une chaîne logistique un

processus de négociation est déclenché. Ce dernier est fortement relié aux quantités demandées, aux dates de livraison souhaitées et aussi aux coûts. Une manière d'évaluer la performance d'une négociation réside dans le temps mis en œuvre pour aboutir à un consensus. À travers ce travail, nous cherchons à optimiser les temps de réponse dans la chaîne logistique tout en permettant la satisfaction des commandes client par diminution de nombre des commandes livrer en retard ou annulés.

3.3 Illustration de notre approche de prise d'une demande imprévue

Pour illustrer notre modèle décisionnel présenté dans les chapitres précédents ainsi que la démarche globale de prise d'une nouvelle demande imprévue, nous allons présenter dans cette section un exemple d'une entreprise qui reçoit une demande imprévue et qui se trouve dans la situation décrite par le cas 4 du tableau 2.1 du chapitre 2. Ainsi l'entreprise se retrouve dans une situation où elle doit trouver les composants manquants nécessaires pour la fabrication de produit fini.

La chaîne logistique distribuée utilisée pour cet exemple se compose d'un client, d'une entreprise cible, de quatre fournisseurs. Chaque entité est libre de ses décisions, il n'y a pas de relations hiérarchiques entre les partenaires.

L'entreprise cible (Constitué de l'agent App, l'agent Fab et l'agent Dist) reçoit une demande imprévue de son client (de l'agent App _i de client). Elle est imprévue, car elle diverge de la demande prévisionnelle préalablement établie, dans cette thèse nous ne nous intéressons pas à la phase d'élaboration de demande prévisionnelle, Nous considérons qu'une période correspond à un jour.

Pour pouvoir répondre favorablement à cette demande, l'entreprise doit disposer des quantités excédantes de produits finis. Cela peut être le cas si elle dispose de suffisamment de stock libre de produits finis ou le cas échéant pouvoir produire une partie ou la totalité de ces quantités.

Dans cet exemple, nous considérons que l'entreprise cible ne dispose pas de la totalité des quantités demandées manquantes. Alors en résumé, pour répondre favorablement à la demande imprévue du client l'entreprise doit résoudre le problème d'insuffisance des capacités en matière de composants. Nous allons appliquer les cinq points de l'approche que nous avons présentée dans le chapitre 2 de ce mémoire.

3.3.1 Création de scénarios d'approvisionnement

Afin de proposer plusieurs plannings de livraison à ses fournisseurs, l'entreprise cible vérifie les possibilités de placement de charge de travail dans sa production. Nous supposons dans notre exemple que cette phase peut être réalisée par des outils internes de planification de la production dans le service de Fabrication représenté par l'Agent *Fab*. Dans ce cas, les capacités de production sont réservées pendant la période de négociation et d'examen de la

demande imprévue du client. Nous rappelons que l'entreprise ne traite qu'une seule demande à la fois. À partir des possibilités de placement de la nouvelle charge de production induite par la demande imprévue, nous réalisons les trois scénarios d'approvisionnement (Tableau 5.1)

Scénario	Date1	Date2	Date3	Date4	Date5
Scénario 1	350	320	130	90	100
Scénario 2	290	290	120	100	190
Scénario 3	300	290	200	120	80

Tableau 5.1 : Les scénarios d'approvisionnements de composant1 pour une semaine donnée

Les trois scénarios ainsi constitués forment la base de recherche de partenaires (fournisseurs) pour satisfaire la demande initiale du client.

3.3.2 Recherche des fournisseurs potentiels

Cette phase consiste à rechercher les fournisseurs potentiels pour résoudre le problème, Alors l'Agent *App* recherche dans sa base de données les fournisseurs des composants nécessaires pour la fabrication de produit fini, l'entreprise cible dispose de 4 fournisseurs du composant1.

3.3.3 Envoi de demandes aux fournisseurs potentiels

La demande de composants est constituée pour un seul produit. Dans notre exemple, l'Agent *App* de l'entreprise cible envoie une demandes aux Agents *Dist_i* des fournisseurs contenant tous les scénarios d'approvisionnement pour le composant1 donné.

3.3.4 Réception de propositions d'approvisionnement

L'ensemble des réponses des Agents *Dist_i* (des fournisseurs) aux demandes d'approvisionnement est résumé dans les tableaux suivant.

	Date1	Date2	Date3	Date3	Date5	Coût unitaire
Fournisseur1	350	320	130	90	100	3
Fournisseur2	360	340	140	100	120	4
Fournisseur3	350	300	160	100	100	3
Fournisseur4	400	400	200	100	100	4
Totale	1460	1360	630	390	420	

Tableau 5.2: Réponse fournisseurs pour le composant1 et le scénario S1

	<i>Date1</i>	<i>Date2</i>	<i>Date3</i>	<i>Date3</i>	<i>Date5</i>	<i>Coût unitaire</i>
<i>Fournisseur1</i>	280	300	120	100	190	3
<i>Fournisseur2</i>	300	300	150	100	190	4
<i>Fournisseur3</i>	290	290	120	100	190	3
<i>Fournisseur4</i>	300	300	200	100	190	4
<i>Totale</i>	1170	1190	590	400	760	

Tableau 5.3: Réponse fournisseurs pour le composant1 et le scénario S2

	<i>Date1</i>	<i>Date2</i>	<i>Date3</i>	<i>Date3</i>	<i>Date5</i>	<i>Coût unitaire</i>
<i>Fournisseur1</i>	300	290	200	120	80	3
<i>Fournisseur2</i>	300	290	200	120	80	4
<i>Fournisseur3</i>	200	200	200	200	200	3
<i>Fournisseur4</i>	300	300	300	120	100	4
<i>Totale</i>	1100	1080	900	560	460	

Tableau 5.4 : Réponse fournisseurs pour le composant1 et le scénario S3

A partir de ces réponses le processus de sélection peut commencer. L'interface ci-dessous (*Figure 5.9*) de l'Agent Distribution ***Dist*** contient une partie pour la saisie des informations concernant les réponses ou les propositions des fournisseurs aux scénarios envoyer par l'entreprise, de plus une zone d'échange de message entre l'agent Distribution de fournisseur et les autres agents.

3.3.5 Sélection des propositions des fournisseurs

Dans cette partie, Nous appliquons l'équation E1 pour vérifier la faisabilité d'un scénario d'approvisionnement pour un composant donné. Nous exécutons l'algorithme présenté dans le chapitre 2, aussi nous utilisons un outil de résolution des programmes linéaires CPLEX pour résoudre le modèle et nous présentons les résultats de l'application de notre méthode sur le composant1.

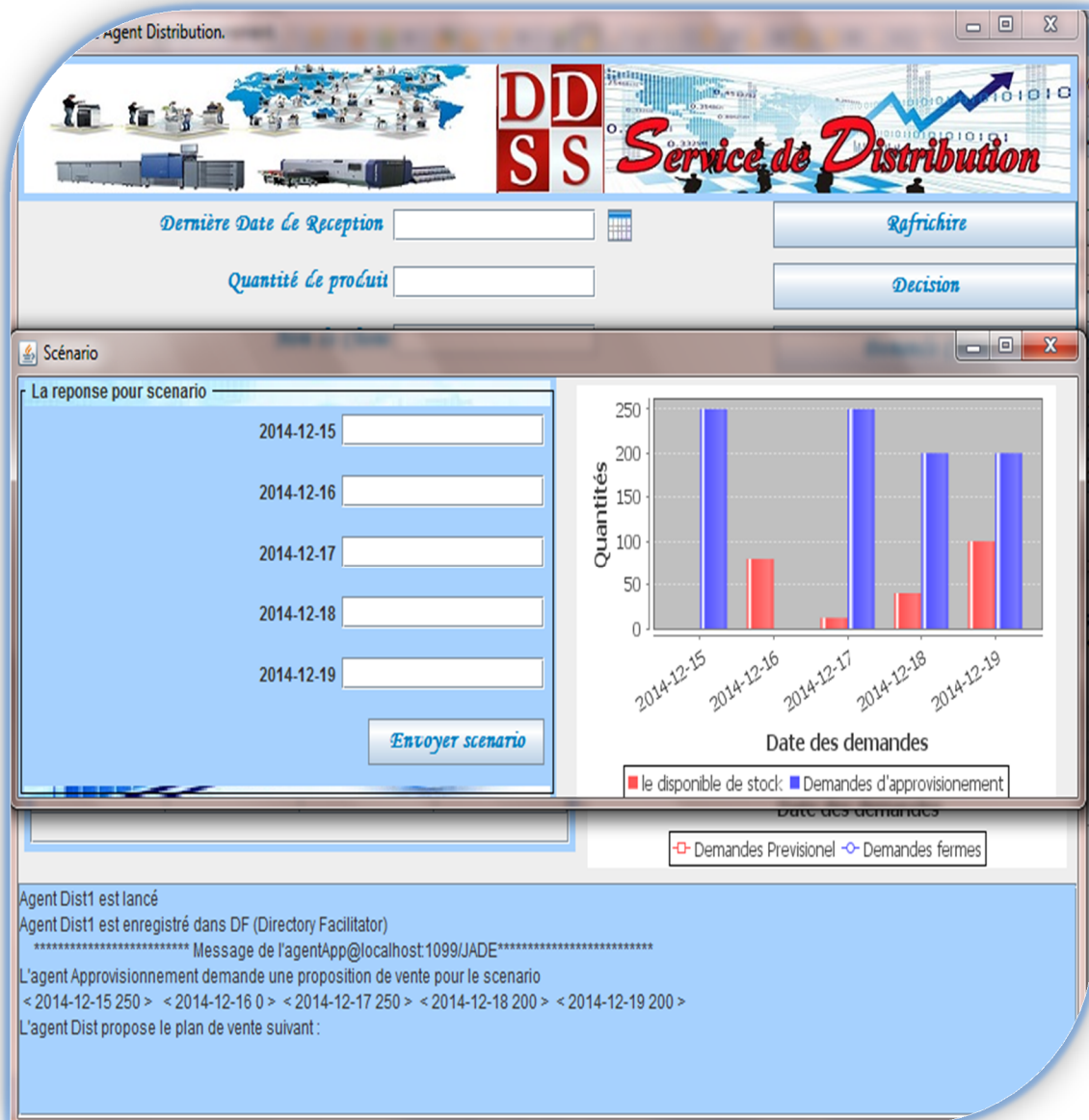


Figure 5.9 : Fenêtre principale de l'Agent Distribution (Dist)

L'interface ci-dessous de l'Agent Approvisionnement contient une partie pour la saisie des informations concernant les scénarios qui vont être ensuite envoyés aux fournisseurs, une zone de réciprocité de message entre l'agent Approvisionnement et les autres agents de système et qui affiche aussi les résolutions du système pour l'aide à la prise de décision après l'enclenchement d'une partie de code CPLEX représenté dans la (figure 5.11), encore une dernière partie qui permet l'élection de bon fournisseur par l'administrateur conséquemment au proposition de système.



Figure 5.10: Fenêtre de l'Agent Approvisionnement

	Scénario1	Scénario2	Scénario3
Quantité totale demandée	990	990	990
Coût total d'acquisition	3180.0	2970.0	3000.0
Nombre des fournisseurs	4	4	4

Tableau 5.5 : Résultats de la sélection de fournisseur pour le scénario 1, 2 et 3

vue la différence dans le cout total d'acquisition et l'abaissement de prix avec un pourcentage de 6,60% dans le scénario 2 par rapport au scénario 1 et la diminution de prix de 5,66% dans le scénario 3 par rapport au scénario 1, alors on peut dire que plus le nombre de scénario augmente la probabilité d'avoir un prix d'approvisionnement optimal est stimulé aussi pour l'entreprise.

```

//-----
import ilog.concert.IloLinearNumExpr;
import ilog.concert.IloNumExpr;
import ilog.concert.IloNumVar;
import ilog.cplex.IloCplex;

//-----

try {

    IloCplex cplex = new IloCplex();
    IloNumVar[] x = cplex.boolVarArray(conteur);
    int[] objvals = {prix[0],prix[1],prix[2],prix[3]};
    cplex.addMinimize(cplex.scalProd(x, objvals));
    for (int j = 0; j < 5; j++) {
        cplex.addGe(cplex.scalProd(fournniss[j], x), cinar[j]);    }

    if ( cplex.solve() ) {
        if (cplex.getStatus().equals(IloCplex.Status.Infeasible ) ) {
            //-----
        }
        jTextArea.append(" Etat de la solution= " + cplex.getCplexStatus()+ "\n");
        jTextArea.append(" Le Temps d'exécution Cplex= " + cplex.getCplexTime()+ "\n");

        double[] val = cplex.getValues(x);
        int ncols = cplex.getNcols();
        int j = 0;
        while (it1.hasNext() || j < ncols ) {

            if (val[j] == 0 || (tableteste[4][j] < cumcinar[4])) {

                // Le code si la condition est vrai

            }

            else {

                // Le code si la condition n'est pas vérifiée

            }

            j++;

        }

        cplex.end();

    } catch (IloException e) {
        System.err.println("Concert exception " + e + " caught");
    }
}

```

Figure 5.11 : Code d'exécution avec CPLEX



Figure 5.12: fenêtre d'identification

La fenêtre ci-dessus est la fenêtre principale des Agents *App*, *Fab* et *Dist* et c'est la première qui apparaît lors du lancement des Agents car l'accès aux comptes est protégé et donc nécessite une identification par un nom d'utilisateur et un mot de passe ce qui est illustré dans la première fenêtre du système.

3.4 Simulation

Le but de ce simulateur est de fournir des indicateurs de performances des échanges d'informations en vue de répondre à des commandes fermes ou urgente. Il est à noter qu'une solution d'une négociation n'est pas nécessairement un succès et que la qualité du résultat réside dans le fait de ne pas refuser une commande qui était en réalité acceptable et aussi ne pas livrer une commande en retard.

Comme nous l'avons dit, il existe plusieurs manières d'insérer une demande dans le planning initial. À l'aide de la simulation, nous cherchons à déterminer le nombre adéquat de scénarios d'approvisionnement nécessaire à l'aboutissement d'une négociation au regard des performances attendues de la coordination.

Dans cette partie, la simulation consiste à évaluer l'impact du nombre de scénarios sur les performances des prises de décision. Pour cela, une série de tests est réalisée. Ils illustrent différentes configurations de chaînes logistiques.

Les spécifications, prises en compte, des entités qui composent les chaînes logistiques étudiées sont relatives à une fabrication mono produit et à une stratégie d'approvisionnement mono ou multi fournisseurs.

Nous avons aussi fait la simulation de deux entreprise (chacune est représentés en fonctions des trois agents App, Dist, Fab) qui appartiennent à la même chaine logistique la première fait recours au fournisseurs de produit fini pour satisfaire ses clients dans le pire des cas et une autre entreprise qui préfère tous simplement répondre par le refus de la commande client si elle n'arrive pas à faire la production dans les délai prédéfinis, Au cours de la simulation, nous avons calculé l'indicateur de performance qui représente le nombre des commandes livrées en retard par les deux entreprise.

3.5 Résultats des simulations

Le test fait sur un échantillon de 100 demandes client (les demandes client sont généré aléatoirement) avec 4 fournisseurs, a abouti aux résultats suivants, l'utilisation d'un seule scenario à donner le pourcentage de 80% de demande client accepté pour les autres 20% des commandes, l'entreprise doit entamer une négociation avec les clients avant les refusés, l'utilisation de deux scenarios a parvenu à une augmentation de 7,2 % des demandes accepté, enfin l'utilisation de trois scenarios à accéder de plus le nombre de demande client agréé par apport a un seule scenario avec un pourcentage de 11,25% et diminuer automatiquement le temps de négociation avec les clients en plus d'une optimisation dans le coût globale d'approvisionnements.

Les résultats obtenus démontrent clairement l'impact de l'utilisation des scénarios dans la convergence et les temps des négociations. et nous déduisons aussi que le recours au fournisseurs de produit fini a réduit considérablement le nombre des commandes livrées en retard. En conséquence, ceci se traduira par un bon niveau de service dans l'optique de la fidélisation du client final.

4 Conclusion

Ce chapitre a été consacré à la phase d'implémentation et de validation de l'architecture proposée sous l'environnement JADE. Nous avons présenté en détail la notion de comportement et le modèle d'agent JADE. Ensuite nous avons montré que ces deux caractéristiques technique sont renforcé beaucoup notre choix de la plateforme JADE, Puis à titre d'exemple, on a pu générer la structure partielle de l'agent « Dist » à partir de son diagramme d'états-transitions.

Nous utilisons aussi dans ce chapitre l'architecture multi agents élaborée préalablement afin de réaliser un simulateur de chaîne logistique distribuée. Le but de ce simulateur est de permettre une évaluation des performances, Le simulateur nous permet la réalisation de deux séries de tests, le premier concerne le dimensionnement du nombre de scénarios nécessaires pour répondre favorablement à un maximum de demandes, La seconde série de tests concerne De faire le recoure au fournisseur de produit fini ce qui réduit considérablement le nombre des commandes livrées en retard. Ceci constitue ainsi en quelque sorte une deuxième validation expérimentale de notre approche.

Conclusion Générale Et Perspectives

Le travail abordé dans ce mémoire de thèse est relatif à la prise de commande imprévue dans une chaîne logistique distribuée. Les enjeux principaux de ces travaux sont d'une part la description de mécanismes de prises de décisions coopératifs en vue de la confirmation ou non de la prise d'une commande imprévue. Le second objectif de ce mémoire est de classer et de mettre en lumière les différents problèmes liés à ce genre de décision. Du point de vue d'un acteur de décision, ces problèmes sont considérés soit comme internes, soit comme externes, La gestion de la chaîne logistique apparaît comme un ensemble de processus soutenus par des applications logicielles. La mise en œuvre de tels processus implique que les différents acteurs de la chaîne communiquent et collaborent pour améliorer les performances industrielles.

La chaîne logistique considérée dans ce travail est dite distribuée. Cela signifie que chaque entreprise qui la compose est libre de prendre ses propres décisions. Nous avons aussi pris l'hypothèse que le contexte de la prise de décision pour la prise de commande imprévue est opérationnel. Nous sommes au niveau opérationnel au sens où nous réservons des ressources pour assurer la prise des commandes imprévues que nous souhaitons accepter.

...Problématique étudiée

La problématique que nous abordons dans ce mémoire réside dans la satisfaction des demandes des clients par une entreprise. Ces demandes sont particulièrement celles qui divergent fortement des prévisions préalablement établies. Dans cette situation, le processus décisionnel peut être complexe. Il est étroitement lié aux flexibilités dont disposent l'entreprise et ses partenaires pour faire face à ces variations importantes non prévues. Ces flexibilités sont relatives aux disponibilités des ressources nécessaires à l'acceptation des demandes. Elles sont de nature à influencer directement la bonne marche de la prise de décision. Ces ressources sont liées aux quantités de stock des produits finis, aux capacités de production, aux quantités de stock de composants nécessaire à l'achèvement d'une production.

La prise de décision dans le contexte opérationnel se complique lorsque les délais pour répondre à une commande imprévue sont réduits. Cela est essentiellement dû au manque de temps dont dispose l'entreprise pour acquérir les flexibilités manquantes pour satisfaire la

commande imprévue.

La question qui se pose alors réside dans le fait de proposer un processus apte à satisfaire une commande imprévue dans une structure de décision distribuée et pour lequel le temps alloué pour le choix est restreint.

...Bilan de notre démarche et des résultats obtenus

Pour répondre à la problématique dégagée dans l'introduction générale et le chapitre1, nous avons commencé par définir explicitement les concepts clés de gestion des chaînes logistiques et leurs composantes coopératives. Nous avons présenté une synthèse sur les outils et les pratiques permettant d'améliorer la collaboration entre les acteurs d'une chaîne logistique. Nous avons ainsi pu témoigner la présence de quelques faiblesses dans ces derniers, surtout au niveau de la robustesse et la gestion des situations d'urgences causées par les exceptions pouvant altérer le fonctionnement normal de la chaîne et dévier ainsi son état planifié. Ensuite, Nous avons relevé le manque de méthodologies et de modèles génériques pour les chaînes logistiques en utilisant les systèmes multi-agent. Nous avons ainsi témoigné du faible nombre de travaux relatifs à ces sujets.

De là, nous avons proposé un modèle fondé sur les concepts d'agents et d'interactions, la spécificité de ce modèle est qu'il traite les situations imprévues en s'appuyant sur des outils mathématiques d'aide à la décision aptes à trouver les ressources manquantes. Ces ressources peuvent être des ressources matérielles. C'est le cas pour des matières premières ou des composants manquants. Elles peuvent être aussi des produits finis. Nous avons aussi proposé une approche méthodologique à suivre afin de délimiter la structure et le périmètre de la chaîne logistique à modéliser. Ceci a conduit à l'identification de 3 agents (App, Fab, Dist) nécessaires pour modéliser chaque acteur. Chacun de ces agents, cherche à coopérer avec les autres pour augmenter le nombre de scénarios possibles pour répondre à une demande client imprévue.

Nous utilisons enfin cette architecture pour la réalisation d'un outil d'évaluation de performance basé sur la simulation de chaîne logistique. Cela nous permet d'évaluer, notamment, les performances des négociations induites par les prises de décisions. Nous avons validé l'architecture proposée par le biais d'une autre simulation, cette dernière a concerné l'évaluation de l'importance du faire recours au fournisseur de produit fini pour satisfaire la demande client dans les délais. Les différents cas étudiés ont montré l'apport des agents dans la coopération entre acteurs de la chaîne et l'intérêt qu'ils apportent au niveau de l'aide à la décision distribuée. Les agents de notre architecture facilitent et automatisent les négociations entre clients/fournisseurs. Ceci contribue à la propagation de l'impact d'une nouvelle donnée sur l'ensemble de la chaîne logistique et de procéder rapidement à une replanification lorsqu'un aléa survient.

En conclusion de ces travaux, nous pouvons dire que le problème de prise de commande imprévue, dans le contexte de chaîne logistique distribuée, est très complexe. Nous avons fait le choix du traitement de ce problème par la formalisation d'un mécanisme de prise de décision et la réalisation des outils d'aide à la décision. L'architecture multi agents que nous avons proposée permet l'implantation de ces outils ainsi que de simuler le fonctionnement de la chaîne logistique en vue d'évaluer les performances des mécanismes de négociations.

...Perspectives

Concernant la continuité du déploiement de l'architecture proposée, nous identifions plusieurs thèmes :

- Le développement d'un outil d'aide à la re-planification serait une solution qui pourrait être adaptée et implantée dans l'architecture multi agents que nous proposons.
- Développer l'architecture pour assurer la connexion des agents avec un grand nombre de systèmes d'informations utilisés par les industriels. Nous envisageons pour cela de travailler parallèlement avec des éditeurs de systèmes d'informations et des entreprises pour lesquelles la problématique de la gestion des commandes client imprévue et des exceptions est importante.
- Poursuivre le travail sur l'adaptation et l'apprentissage des agents.

Références Bibliographiques

- [1] Berkrane, Mottay, Link-Pezet ,« *Intelligence économique et décision* » , 2002 .
- [2] Scott.M.S, « *Management decision systems, computer based support for decision making* », Harvard University, Boston, MA, USA, 1971.
- [3] Keen P. G. W et Scott Morton M. S, « *Decision support systems:an organizational perspective*», Addison-Wesley Publishing Company, 1978.
- [4] Cohen E, Colasse B, Lepen C, «*L'aide à la décision et la recherche opérationnelle aujourd'hui*», dans : "*25 ans de sciences des organisations s*», Édition Economica, 1994.
- [5] ChristopherM ,« *Logistics and supply chain management* », Pitman Publishing, London , 1992.
- [6] Lummus R, Vokurka R& Albert K, «*Strategic supply chain planning*» ,Production and Inventory Management Journal , 1998.
- [7] Poirier C, &Reiter S, « *La Supply Chain - Optimiser la chaîne logistique et le réseau interentreprises* » , Dunod, 2001.
- [8] ChopraS, MeindilP, « *Supply chain management: Strategy, planning and operations*», New Jersey, 2007.
- [9] Mentzer J, De Witt W, Keebler J, Min, S, Nix N, Smith C, et al, « *Defining supply chain management*», *Journal of business logistics*, 2001.
- [10] Croom S, Romano P, &Giannakis M, « *Supply chain management: an Analytical framework for critical literature review* », *European Journal of Purchasing & Supply Management*, 2000.
- [11] Vakharia A, «*e-Business and supply chain management* », *A journal of the Decision Sciences Institute*, 2002.
- [12] Simchi-Levi D, &Kaminsky P, «*Designing and Managing the Supply Chain: Concepts, Strategies and Case Studies*», Boston, MA: McGraw Hill Professional, 2003.
- [13] Rota K, «*Coordination temporelle de centres gérant de façon autonome des ressources – Application aux chaînes logistiques intégrées en aéronautique*»,

- Thèse soutenue à Toulouse, Université Paul Sabatier et ONERA, 1998.
- [14] Claver J. F, Gelinier J, PITT D, « *Gestion de flux en entreprise : modélisation et simulation* », Edition Hermès, 1997.
- [15] Simchi-Levi D, Kaminsky P & Simchi-Levi E, « *Designing and managing the supply chain: Concepts, strategies and case studies* », Irwin McGraw-Hill, 1999.
- [16] Despontin-Monsarrat E, « *Aide A La Décision Pour Une Coopération Inter-Entreprises Dans Le Cadre De La Production A La Commande* », Thèse de doctorat, Université de Toulouse 3 Paul Sabatier, Toulouse France, décembre 2004.
- [17] Statler H et Kilger C, « *Supply Chain Management and Advanced Planning : concepts, models, software and case studies* », Editions Springer Verlag, 2000.
- [18] Hétreux G, « *Structures de décision multi-niveaux pour la planification de la production : robustesse et cohérence des décisions* ». Thèse de doctorat de l'Institut National des Sciences Appliquées de Toulouse, 1996.
- [19] Lecompte-Alix T, « *Un cadre formel de décision pour la planification multi-niveau des systèmes de production réticulaires* », Thèse de doctorat de l'Université Bordeaux 1, 2001.
- [20] Hatchuel A, Saidi-Kabeche D, Sardas J. C, « *Towards a new planning and scheduling approach for multistage production systems* », International Journal of Production Research, 1997.
- [21] Fontan. G, Mercé. C, Hennes. J.C, Lasserre. J.B , « *Hierarchical scheduling for decision support* », Journal of Intelligent Manufacturing, 2005.
- [22] Ganeshan R, Jack E, Magazine M & Stephens P, « *A Taxonomic Review of Supply Chain Management Research* », in Quantitative Models for Supply Chain Management, Boston, 1998.
- [23] Vincent L, Neubert G, Llerena D et Pellegrin C, « *Synthèse des approches SCM Existantes* », 2004.
- [24] Botta-Genoulaz V, « *Principes et Méthodes pour l'Intégration et l'Optimisation du pilotage des Systèmes de Production et des Chaînes Logistiques* », Institut National des Sciences Appliquées de Lyon et Université Claude Bernard de Lyon 1, 2005.
- [25] Croom S, Romano P, & Giannakis M, « *Supply chain management: an Analytical framework for critical literature review* », European Journal of Purchasing & Supply Management, 2000.

Références Bibliographiques

- [26] Fargier H, Thierry C, « *The use of possibilistic decision theory in manufacturing planning and control; Recent Results in Fuzzy Master Production Scheduling* », Springer-Verlag, 2002.
- [27] Hapke M, Slowinski R, « *Fuzzy set approach to multi-objective and multi-mode project scheduling under uncertainty* », Springer-Verlag, 2000.
- [28] Geneste L, Grabot B et Moutarlier P, « *Scheduling of heterogeneous data using fuzzy logic in a customer-subcontractor context, Scheduling under fuzziness* », Springer-Verlag, 2000.
- [29] Letouzey A, Geneste L et Grabot B, « *Production planning forecasting under uncertainty using possibility theory* », CASYS 2001, 5th International Conference on Computing Anticipatory Systems, Liege Belgique, Aout 2001.
- [30] Reynoso C, « *Prise En Compte De Commandes Incertaines Et Imprecises Dans Une Logique MRP2* », Montpellier – France, Octobre 2002.
- [31] Giannoccaro I, Pontrandolfo P, et Scozzi B, « *A fuzzy echelon approach for inventory management in supply chains* », European Journal of Operational Research, 2003.
- [40] Davis T, « *Effective supply chain management* », Sloan Management Review, 1993.
- [41] Lee H.L, Billington C, « *Material management in decentralized supply chain* », Operations Research, 1993.
- [42] Arnold J, « *Production Planning and control within Supply Chains* », IT and Manufacturing Partnerships, 1996.
- [43] Rota K, « *Coordination Temporelle de Centres Gérant de Façon Autonomes des Ressources: Application aux Chaînes Logistiques Intégrées en Aéronautique* », Thèse de doctorat. Université de Toulouse, France, 1998.
- [44] Rohde J, Meyr H & Wagner M, « *Die supply chain planning matrix, in : PPS Management* », Berlin, 2000.
- [45] Thomas D.J, Griffin P.M, « *Coordinated supply chain management* », European Journal of Operational Research, 1996.
- [46] Kearney A.T, « *Management approach to supply chain integration* », Rapport aux membres de l'équipe de recherche A.T. Kearney, Chicago, 1994.
- [47] Poirier C, et Reiter S.E, « *La Supply Chain* », Dunod, 2001.
- [48] SCC Supply chain operations reference model: overview of SCOR version 7.0; Supply chain council Inc. available on <http://www.supplychain.org>; 2006.

Références Bibliographiques

- [49] Ouzizi L, «*Planification de la production par co-décision et négociation de l'entreprise virtuelle*», Thèse de doctorat, Université de Metz, January 2005.
- [50] Moyaux, «*Design, Simulation and analysis of collaborative strategies in multi agent systems : the case of supply chain management*», Thèse de doctorat, Université Laval, 2004.
- [51] Carlsson C, Fullér R, «*Reducing the bullwhip effect by means of intelligent, soft computing methods*», Dans les actes de 34th Hawaii international conference on system sciences, 2001.
- [52] Taylor D, «*An Approach to the identification and elimination of demand amplification across the supply chain*», Thomson Learning Press, 2001.
- [53] Moyaux T, «*Techniques Multi Agents pour la Réduction de l'Amplification de la Demande dans une Chaîne Logistique : Application à l'Industrie Forrestière*», Résumé de Thèse, Juillet 2002, Université de Laval, Laval Canada, 2002.
- [54] Barnard C.J, «*The functions of the executive*», Harvard University Press, Cambridge, 1938.
- [55] Campagne J.P & Sénéchal O, «*Les nouvelles exigences de la coopération*», Dans *Coopération et connaissance dans les systèmes industriels* sous la direction de R. Soënen et J. Perrin, Lavoisier, Hermes Science, 2002.
- [56] Camalot J.P, «*Aide à la décision et à la coopération en gestion du temps et des ressources*», Thèse de Doctorat, Institut National des Sciences Appliquées de Toulouse, Toulouse, France, Mai 2000.
- [57] Monteiro T, «*Conduite distribuée d'une coopération entre entreprises : le cas de la relation donneurs d'ordres- fournisseurs*», Thèse de Doctorat, Institut National Polytechnique de Grenoble, Grenoble, France, Octobre 2001.
- [58] Quéré M, «*Coopération ou coopérations: quels enjeux économiques ?* », Dans *Coopération et connaissance dans les systèmes industriels* sous la direction de R. Soënen et J. Perrin, Lavoisier, Hermes Science, 2002.
- [59] Boujut J.F, Cavaillé J.B & Jeantet A, «*Instrumentation de la coopération*», Dans *Coopération et connaissance dans les systèmes industriels*, Lavoisier, Hermes Science, 2002.
- [60] Botta-Genoulaz V, «*Les systèmes d'information supports à la chaîne logistique*», Ecole d'été d'automatique – Gestion de la Chaîne Logistique. Session 24, Grenoble, France, Septembre 2003.
- [61] Weber C.A, Current J.R, Benton W.C, «*Vendor Selection Criteria and Methods*», European Journal of Operational Research, August, 1991.

Références Bibliographiques

- [62] Bouchriha Hanen, «*Système d'aide à la décision dans l'établissement d'un réseau d'entreprises*», INP grenoble, 2002.
- [63] Nydick R.I, Hill R.P, «*Using the Analytic Hierarchy Process to Structure the Supplier Selection Procedure*», International Journal of Purchasing and Materials Management, Spring 1992.
- [64] Current J, Weber C, «*Application of Facility Location Modeling Constructs to Vendor Selection Problem*», European journal of operational research, 1994.
- [65] Ittner C.D, Larcker D.F, Nagar V, Rajan M.V, «*Supplier selection, monitoring practices, and firm performance*». Journal of Accounting and Public Policy 1999.
- [66] Shin H, Collier D.A, Wilson D.D, «*Supply management orientation and supplier/buyer performance*», Journal of Operations Management, 2000.
- [67] Boer L, van der Wegen, Telgen L, «*Outranking methods in support of supplier selection*». European Journal of Purchasing & Supply Management , 1998.
- [68] Monczka R, Trent R, Handfield R, «*Purchasing and Supply Chain Management*», South-Western College Publishing, New York, 1998.
- [69] Ferber J, «*Les systèmes multi-agents. Vers une intelligence collective* ». Inter Editions, Paris, 1995.
- [70] Wooldridge M, «*Intelligent agents*», The MIT Press, 1999.
- [71] Briot J.P, Demazeau Y, «*Systèmes Multi-Agents. Principes et architectures*», Collection IC2, Hermes Science.
- [72] Hunhns M, Singh M.P, «*A Multiagent Treatment of Agenthood*», en Applied Artificial Intelligence: An International Journal, January-March 1999.
- [73] Sycara K, «*Multi-Agent Systems*», AI Magazine, Summer 1998
- [74] Franklin S, Graesser A, «*Is it an Agent, or just a Program? A Taxonomy for Autonomous Agents*». Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages. Springer-Verlag, 1996
- [75] Hussein Joumaa, Yves Demazeau, and Jean-Marc Vincent, «*Evaluation of multi-agent systems : The case of interaction*», In ICTTA'08 : Proceedings of the 3rd International Conference on Information and Communication Technologies : from Theory to Applications, Damascus, Syria, april 2008. IEEE, IEEE Computer Society.
- [76] Foundation for Intelligent Physical Agents. FIPA ACL Message Structure Specification. (On <http://www.fipa.org/specs/fipa00061/SC00061G.html>), 2002

- [77] Verwijmeren, «*Software component architecture in supply chain management*»Computers in Industry, 2004.
- [78] OunnarF, Pujop, Mekaouchel, Giambiasin, «*Modélisation d'un système isoarchique et simulation via un environnement distribue* », HLA, Mosim 08, Paris, 2008.
- [79] Cloutier L, Frayret J-M, D'Amours S, Espinasse B et Montreuil B,«*A commitment-oriented framework for networked manufacturing coordination*». International journal of computer integrated manufacturing, 2001.
- [80] Wooldridge M, Jennings N. R, «*Intelligent agents: Theory and practice*», The Knowledge Engineering review, 1995.
- [81] Parunak H. V. D,«*Applications of distributed artificial intelligence in industry*». In O'Hare, G. M. P. And Jennings, N. R., editors, Foundations of Distributed Artificial Intelligence, John Wiley et Sons,1996.
- [82] Maturana F, Shen W et Norrie DH, «*MetaMorph: an adaptive agent-based architecture for intelligent manufacturing*», International Journal of Production Research, 1999.
- [83] Govindu R., Chinnam R.B,«*A generic process-centered methodological framework for analysis and design of multi-agent supply chain systems*» Computers & Industrial Engineering, 2007
- [84] Chehbi, S. «*Modélisation et Simulation des Processus Collaboratifs dans les Chaînes Logistiques: Une Approche Multi Agents*», Thèse de doctorat, Université LUMIERE-LYON II, 2007
- [85] Chehbi S., Derrouiche R., Ouzrout Y. et Bouras A. «*Multi-Agent Supply Chain Architecture to Optimize Distributed Decision Making*», Proceedings of the 7th World Multi-conference on Systemics, Cybernetics and informatics, SCI, 16, Orlando, USA, 2003.
- [86] Telle O, Thierry C. et et Bel G. «*Simulation d'une relation client/fournisseur au sein d'une chaîne logistique intégrée: Mise en œuvre industrielle*»,2003.
- [87] Bournez C, Gutknecht O. «*Modèle de contrôle par émergence de coordination dans un réseau de contrats multiagents*», In Journées Francophones de l'Intelligence Artificielle Distribuée et des Systèmes MultiAgents, Montréal, Québec, Canada, 2001.
- [88] Jiao J, You X. et Kumar, A. «*An agent-based framework for collaborative negotiation in the global manufacturing supply chain network*», Robotics and Computer-Integrated Manufacturing, 2006.

- [89] Barbuceanu M, Fox M, «*Cool: A language for describing coordination in multi-agents systems*», In Proceedings of the first International Conference on Multi-agent Systems, 1995.
- [90] Sadeh N, Hildum DW, Kjenstad D. et Tseng A. «*MASCOT: an agent-based architecture for dynamic supply chain creation and coordination in the internet economy*», Prod Plan Control, 1999.
- [91] Baumgaertel H, Brueckner S, Parunak V, Vanderbok R., et Wilke, J, «*Agent Models of Supply Network Dynamics*», The Practice of Supply Chain Management, 2001.
- [92] Parunak H. V. D., VanderBok, R. «*Modeling the extended supply network*» In ISA-Tech'98 (Houston), Industrial Technology Institute, 1998.
- [93] Terzi S, Cavalieri S, «*Simulation in the supply chain context: a survey. Computers in Industry*», 2003.
- [94] Swaminathan J.M., Smith S.F. et Sadeh N.M, «*Modeling Supply Chain Dynamics: A Multiagent Approach*», Decision Sciences, 1998.
- [95] Govindu R., Chinnam R.B. «*A generic process-centered methodological framework for analysis and design of multi-agent supply chain systems*». Computers & Industrial Engineering, 2007.
- [96] Fox M. S., Barbuceanu M., et Teigen R, «*Agent-oriented supply chain management*», The International Journal of Flexible Manufacturing Systems, 2000
- [97] Barbuceanu M, Fox M. «*Coordinating multiple agents in the supply chain*». In Proceedings of the Fifth Workshops on Enabling Technology for Collaborative Entreprises, WET ICE'96, IEEE Computer Society Press, 1996.
- [98] Beck J., Fox M, «*Supply chain coordination via mediated constraint relaxation*», In Proceedings of the First Canadian Workshop on Distributed Artificial Intelligence, Banff, Canada, 1994.
- [99] Teigen R. et Barbuceanu M. «*The supply chain demonstrator*». Rapport technique et guide de l'utilisateur, 1996.
- [100] Liang W.Y, Huang C.C, «*Agent-based demand forecast in multi-echelon supply chain*», Decision Support Systems, 2006.
- [101] Moyaux T, Chaib-draa B, et D'Amours S, «*An agent simulation model for the québec forest supply chain*», Proceedings of the 8th International Workshop on Cooperative Information Agents (CIA), Erfurt, Germany, 2004.
- [102] Kimbrough S.O., Wu D.J. et Zhong, F, «*Computers play the beer game: can artificial agents manage supply chains?*», Decision Support Systems, 2002.

- [103] Deloach S., Wood M. et Sparkman, C, «*Multiagent Systems Engineering – MaSE*», The International Journal of Software Engineering and Knowledge Engineering, 2001.
- [104] DeLoach, S., Wood, M, «*Developing Multiagent Systems with agentTool*», en Y. Lesperance & C. Castelfranchi, editores, Intelligent Agents VII - Proceedings of the 7th International Workshop on Agent Theories, Architectures, and Languages (ATAL'2000), 2000
- [105] Pavón, J «*INGENIAS : Développement Dirigé par Modèles des Systèmes Multi-Agents*», Dossier d'Habilitation à Diriger des Recherches de l'Université Pierre et Marie Curie Spécialité: Informatique, 2006
- [106] Zambonelli, F., Jennings, N. et Wooldridge, M «*Developing Multiagent Systems: The Gaia Methodology*. ACM Trans. Software Engineering and Methodology, 2003.
- [107] Ferber J, Gutknecht O, «*A Meta-Model for the Analysis and Design of Organizations in Multi-Agent Systems*», Proceedings of the Third International Conference on Multi- Agent Systems (ICMAS98), IEEE CS Press, 1998.
- [108] Gutknecht, O., Ferber, J. «*The MADKIT Agent Platform Architecture*», Workshop on Infrastructure for Scalable Multi-Agent Systems, 2000 Revised Papers. LNCS 1887, Springer Verlag, 2001.
- [109] Picard, G., Bernon, C., Gleizes, M., et Peyruqueou, S, «*ADELFE: a methodology for adaptive multi-agent systems engineerin*». In Petta, P., Toks Dorf, R., and Zambonelli, F., editors, Engineering Societies in the Agents World III: Third International Workshop (ESAW 2002), LNCS 2577, Springer Verlag, 2002.
- [110] Rao, A. et Georgeff, M , «*Modeling Rational Agents within a BDI Architecture*». Proceedings of the 2nd international conference on Principles of Knowledge Representation and Reasoning, 1991.
- [111] Kendall, E, «*A Methodology for Developing Agent Based Systems for Enterprise Integration*», IFIP Working Conference of TC5 SIG on Architectures for Enterprise Integration, Queensland, Australia, November 1995.
- [112] Lind, J, «*Iterative Software Engineering for Multiagent Systems: The MASSIVE Method*», Springer 2001.
- [113] Glaser, N, «*Contribution to Knowledge Modelling in a Multi-Agent Framework (the CoMoMAS Approach)* », Thèse de l'Université Henri Poincaré, Nancy I, 1996
- [114] Iglesias, C., Garijo, M., Gonzalez J. et Velasco J, «*Analysis and design of*

- multiagent systems using MAS-CommonKADS*», In: M. P. Singh, A. Rao, and M. J. Wooldridge (ed.), *Intelligent Agents IV*, LNAI 1365, Springer Verlag, 1998.
- [115] Pavón J., Gómez-Sanz J. «*Agent Oriented Software Engineering with INGENIAS*», In: Proc. 3rd International Central and Eastern European Conference on Multi-Agent Systems (CEEMAS 2003), V. Marik, J. Müller, M. Pechoucek (Eds.), 2003.
- [116] Demazeau, Y. «*From Interactions To Collective Behaviour In Agent-Based Systems*», Proceedings of the First European Conference on Cognitive Science, Saint Malo, France, Avril 1995.
- [117] Gervais, M, «*ODAC: An Agent Oriented Methodology Based on ODP*», *Autonomous Agents and Multi-Agent Systems*, 2003.
- [118] Moulin, B., Brassard, M. «*A scenario-based design method and environment for the development of multiagent systems*», En Proceedings of the 1st Australian Workshop on Distributed Artificial Intelligence, LNAI 1087 – Springer Verlag, 1996.
- [119] Caire, G., Leal, F., Chainho, P., Evans, R., Garijo, F., Gomez-Sanz, J. J., Pavon, J., Kerney, P., Stark, J., &Massonet, P, «*Agent Oriented Analysis using MESSAGE/UML*», Second International Workshop, AOSE 2001.
- [120] Bauer B, Bergenti F, Massonet, Ph. et Odell J. «*Agents and the UML: a unified notation for agents and multi-agent systems*», In Proceeding of the AOSE 2001, Springer, Montreal, 2001
- [121] Luck, M., McBurney, P. et Preist, C. «*A Manifesto for Agent Technology: Towards Next Generation Computing*», *Journal of Autonomous Agents and Multi-Agent Systems*, 2001.
- [122] Bauer, B., Bergenti, F., Massonet, Ph. et Odell, J, «*Agents and the UML: a unified notation for agents and multi-agent systems*», In Proceeding of the AOSE 2001, Springer, Montreal, 2001
- [123] Burkhart, R. «*The SWARM multi-agent simulation system*», In *Object Oriented Programming Systems, Languages and Applications (OOPSLA)*, 1994.
- [124] Bellifemine F., Poggi A., Rimassa G., «*JADE – A FIPA compliant agent framework*», CSELT internal technical report. Part of this report has been also published in Proceedings of PAAM'99, London, April 1999.
- [125] Bellifemine F., Giovanni C., TizianaT., Rimassa G., «*Jade Programmer's Guide*», Jade version. 2000.
- [126] Fabio Bellifemine, FedericoBergenti, Giovanni Caire, et AgostinoPoggi.

- «*JADE - A Java Agent Development Framework*», chapter 5. Bordini et al. 2005.
- [127] Bellifemine F, Caire G, A. Poggi, G., « *JADE : A software framework for developing multi-agent applications* », Lessons learned. In *Information and Software Technology*, 2008.
- [128] Guessoum Z, « *Modèles et Architectures d'agents et de systèmes multi-agents adaptatifs* », Thèse d'habilitation, Université Pierre et Marie Curie, 2003.
- [129] MADKIT. <http://www.madkit.org>.
- [130] <http://www.objectspace.com/products/voyager/>
- [131] Daniels M, « *Integrating simulation technologies with swarm* », In *Agent Simulation: Application, Models, and Tools*, University of Chicago. 1999.
- [132] Daniels M, « *An open framework for agent-based modeling* », In *Applications of Multi-Agent Systems in Defence Analysis*, Los Alamos National Labs. 2000.
- [133] <http://www.swarm.org/>
- [134] Gutknecht O et Ferber J, « *Vers une méthodologie organisationnelle pour les Systèmes multiagents* », Actes des 7èmes journées Francophones D'Intelligence Artificielle et SMA. Eds Hermès, 1999.
- [135] Nwama H et al, « *Zeus: A toolkit for building distributed multi-agent systems* ». *Applied Artificial Intelligence Journal*, 1999
- [136] Lopez P et Roubellat F, « *Ordonnancement de la production* ». Série productique, Editions Hermès, *Traité IC2 Informatique –Commande – Communication*, 2001
- [137] Pinedo M, Chao X, « *Operations Scheduling with applications in manufacturing and services* », Editions Mc. Graw-Hill, 1999
- [138] Huet M.P, « *An Application of Agent UML to Supply Chain Management* », In *Proceedings of Agent Oriented Information System (AOIS-02)*, Paolo Giorgini and Yves Lespérance and Gerd Wagner and Eric Yu (eds.), Bologne, Italie, 2002.
- [139] Russell S and Norvig P, « *Artificial Intelligence – A modern approach* », Englewood Cliffs, NJ: Prentice-Hall, 2^e édition, 2003.
- [140] Giard V, « *Gestion de la production et des flux* », Economica, 3^e édition, 2003.
- [141] Monteiro T, « *Conduite distribuée d'une coopération entre entreprises – le cas*

Références Bibliographiques

- de relation donneurs d'ordres- fournisseurs* », Thèse de doctorat, institut national polytechnique de Grenoble, 2001.
- [142] Berkrane, Mottay, Link-Pezet, « *Intelligence économique et décision* », <http://www.urfist.cict.fr/sompub9.html>, 2002.
- [143] Lemoigne, « *Les systèmes de décision dans les organisations* », PUF, 1994.
- [144] MintzbergH, « *Structure et dynamique des organisations* », 1982.
- [145] Levine P, Pomerol J.C, « *Systèmes interactifs d'aide à la décision et systèmes experts* », Edition Hermes, 1990.
- [146] Simon H.A, « *The new science of management decision* », New-Jersey, 1977.
- [147] Roy B, « *Science de la décision ou science de l'aide à la décision ?* », Cahier Lamsade, Université de paris Dauphine, 1992.
- [148] Tahon et TrentesauxD, « *Aide à la décision en gestion De production* », ouvrage Organisation et gestion de la production, 2002.
- [149] www.cxp.fr
- [150] Botta-Genoulaz V, Millet P. A, et GrabotB, « *A survey on the recent research literature on ERP systems* », Computers in Industry, 2005.
- [151] Lamouri S, « *Synchronisation des prises de décisions dans une chaîne logistique: robustesse et stabilité* », Mémoire d'Habilitation à Diriger des Recherches, École Doctorale d'Informatique et Électronique de Paris, Institut Supérieur de Mécanique de Paris (Supméca Paris), 2006.
- [152] Hakanson B, WondergemJ. ajouter le titre. <http://www.supplychain.org/Chapters/Europe/French/10.htm>.
- [153] Doche F, Djorno Y et Douilly N, « *L'art de l'entreprise mondiale, Module 8 : Fidéliser le client mondial* », Financial Times Limited et Editions Village Mondial, Paris, 1999.
- [154] <http://www.ILOG.com>
- [155] <http://lpsolve.sourceforge.net/5.5/>
- [156] Weber C.A, Current J.R, « *A Multi objective Approach to Vendor Selection* ». European Journal of Operational Research 68, 1993.
- [157] Nydick R.I, Hill R.P, « *Using the Analytic Hierarchy Process to Structure the Supplier Selection Procedure* ». International Journal of Purchasing and Materials Management, Spring 1992.

- [158] Bauer B, Müller J.P,«*Methodologies and modeling languages*»,Agent-Based Software Development, Boston, London, 2004.
- [159] Odell J, Van Dyke Parunak H, et Bauer B,«*Representing agent interaction protocols in UML*», Proceedings of the First International Workshop , Springer-Verlag, 2001.
- [160] Bauer, B., Odell, J. «*UML 2.0 and agents: how to build agent-based systems with the new UML standard*». Engineering Applications of Artificial Intelligence, 2005.
- [161] Saleh K., El-Morr C,«*M-UML: an extension to UML for the modeling of mobile agent-based software systems*» Information and Software Technology, 2004.
- [162] Wooldridge M, Jennings N et Kinny D, «*The Gaia Methodology for Agent-Oriented Analysis and Design*»Journal of Autonomous Agents and Multi-Agent System, 2000.
- [162] Huget M-P ,«*Agent UML Class Diagrams Revisited*», In Proceedings of Agent Technology and Software Engineering (AgeS), Germany, 2002.
- [163] AdelFerguen, «*La plate-forme JADE*», « http://www.limsi.fr/~jps/enseignement/examsma/2005/1.plateformes_3/index-Ferguen.html », 2007.
- [164] Ramdani L, « *JADE, Développement et Implémentation de systèmes Multi-Agents* », Master de recherche en Informatique, Université Paris Sud XI, 2007.
- [165] Saidna S, « *Plates-formes des systèmes multi-agents* », Master de recherche en Informatique, Université Paris Sud XI, 2007.
- [166] FIPA. Foundation for Intelligent Physical Agents.<http://www.fipa.org/repository/ips.php3>
- [167] Teti R and D'Addona D, « *Emergent synthesis in supply network tool management*», Advanced Engineering Informatics, 2006.
- [168] Ashutosh S and Pratap K.J. Mohapatra, «*Evaluation of supplier capability and performance: A method for supply base reduction*», Journal of Purchasing & Supply Management, 2006.
- [169] Yonghui F and Rajesh P, «*Supply-side collaboration and its value in supply chains*», European Journal of Operational Research, 2004.
- [170] Florea A,« *Using Utility Values in Argument-based Negotiation*», In Proceedings of IC-AI'02, the International Conference on Artificial Intelligence, LasVegas, 2002.

