

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

Université Mohamed KHIDER - BISKRA
Faculté des Sciences exactes et de Sciences
de la nature et de la vie
Département d'Informatique

Ecole Doctorale d'informatique de l'est
Pôle Constantine

N° d'ordre :
Série :

Mémoire

En vue d'obtention du diplôme de Magister en informatique

Option: Intelligence Artificielle

Une approche d'intégration des agents dans l'ERP

Par :

M^{lle}. HOADJLI Hadia

Soutenu le :12/01/2010

Devant le jury composé de :

Président : Dr. BABAHNINI M^{ed} Chaouki, Maître de conférence à l'université Mohamed Khider, Biskra.

Rapporteur : Dr. KAZAR Okba, Maître de conférence à l'université Mohamed Khider, Biskra.

Examineur : Dr. BALLA Amar, Maître de conférence à l'INI, Alger.

Examineur : Dr. CHERIF Foudil, Maître de conférence à l'université Mohamed Khider, Biskra.

Résumé

Les travaux dans le domaine des ERP «Enterprise Resource Planning » ont offert aux sociétés une maîtrise renforcée de ses activités de production grâce à une optimisation de l'utilisation des ressources. Mais ils rencontrent un problème lié à leur implantation, pour ce fait nous avons fait appel à une branche de l'intelligence artificielle qui est les systèmes multi agent afin de manier aux insuffisances des ERP, le résultat de notre recherche est une modélisation d'un système ERP composé d'agents. Notre approche qui est une approche cognitive se base sur la connaissance pour assurer le bon fonctionnement du système et sa fiabilité.

Mots clés: ERP, PGI, Système multi-agents.

ملخص

العمل في مجال تخطيط موارد المؤسسات، مكن الشركات من تعزيز الرقابة على أنشطة الإنتاج عن طريق تحسين استخدام الموارد. ولكنها تواجه مشكلة في تثبيتها و تنفيذها ، لهذا استعنا بفرع من فروع الذكاء الاصطناعي و هو الأنظمة متعددة الأعوان وذلك لمعالجة أوجه القصور في أنظمة تخطيط موارد المؤسسات ، نتيجة بحثنا هي نمذجة لنظام تخطيط موارد مؤسسات يتألف من أعوان، منهجنا منهج إدراكي قائم على المعرفة لضمان الأداء السليم للنظام وموثوقيته .

الكلمات الرئيسية: تخطيط موارد المؤسسات، الأنظمة متعددة الأعوان

REMERCIEMENT

*Nous ne pouvons que nous prosterner devant notre dieu pour tous ce
qu'il nous a accorder.*

*Mes remerciements vont primordialement à mon encadreur le
Docteur KAZAR OKBA qui m'a dirigé, conseillé, et orienté durant la
progression de ce projet, et qui ne mérite qu'admiration, respect, et
reconnaissance.*

*Je remercie aussi les membres du jury pour avoir accepter la
participation*

*Je remercie aussi tous ceux qui m'ont aidé le long de ma recherche, et
principalement Mr. CHAABI MOHAMED TAHAR qui m'a facilité
l'accès aux données de son entreprise.*

*Sans oublier mes enseignants de graduation de l'université de Biskra
ainsi que ceux de l'école doctorale de l'est.*

A TOUS MERCI DU FOND DU COEUR

DEDICACE

Je veux saisir cette occasion pour dédier ce mémoire à ma mère et mon père qui me sont les plus chers au monde, eux qui m'ont tant donné d'amour et de soutien pour que je puisse arriver à ce niveau d'instruction, méritent plus que des remerciements, j'espère que j'arriverais à les satisfaire.

Que dieu me les gardes.

A mes deux adorables petites soeurs LAMIA et ABIR que j'aime tant et qui font la joie de notre petite famille.

A mes grands parents, mes tantes et mes oncles sans exception, et surtout a ma magnifique tante YAMINA car je ne peux oublier son soutien et son encouragement.

A Djazia, Meriem, et Sana mes copines les plus chères qui ont toujours étaient à mes cotés.

A tout mes amis, ceux qui ont été toujours là quand j'avais besoin d'eux.

Hadia

SOMMAIRE

INTRODUCTION GENERALEERREUR ! SIGNET NON DEFINI.

CHAPITRE I : GENERALITES SUR LES ERP

1. INTRODUCTION	4
2. HISTORIQUE	4
3. APPROCHES ET EVOLUTION	5
3.1. L'APPROCHE MRP (MATERIAL REQUIREMENTS PLANNING).....	6
3.2. MRP A BOUCLE FERMEE.....	7
3.3. L'APPROCHE MRPII.....	8
3.4. L'APPROCHE ERP	8
4. DEFINITION DE L'ERP	9
5. POURQUOI L'ERP ?.....	11
6. LES CARACTERISTIQUES D'UN ERP	11
7. L'ARCHITECTURE D'UN ERP	12
7.1. L'AXE HORIZONTAL	12
7.2. L'AXE VERTICAL	14
8. TYPOLOGIE DES ERP	15
9. AVANTAGES.....	18
10. INCONVENIENTS	19
11. LE MARCHE DES ERP.....	20
11.1. LES ERP PROPRIETAIRES.....	21
11.1.1. SAP.....	21
11.1.2. Oracle - People Soft.....	21
11.2. LES ERP OPEN SOURCE	21
11.2.1. Compiere	22
11.2.2. ERP5.....	22
11.2.3. Dolibarr.....	23
11.2.4. SQL Ledger	23
12. COMMENT FAIRE LE CHOIX D'UN ERP	23
13. LES ETAPES DE MISE EN PLACE D'UN PROGIciel DE GESTION.....	25
14. METHODES D'INTEGRATION DES ERP.....	26
14.1. LA METHODE DU BIG BANG	26
14.2. LA METHODE DE FRANCHISE.....	26
14.3. LA METHODE DE MISE EN PLACE PAR MODULE	26
15. L'IMPLANTATION MULTI SITES DE L'ERP	27
15.1. L'AUTONOMIE LOCALE TOTALE	27

15.2. CONTROLE DES SITES SEULEMENT AU NIVEAU FINANCIER.....	27
15.3. COORDINATION DES OPERATIONS ENTRE LES SITES	27
15.4. COORDINATION RESEAUX DES OPERATIONS	28
15.5. LA CENTRALISATION TOTALE.....	28
16. L'ERP ETENDUE VERS UNE NOUVELLE GENERATION	28
17. ERP ET WFMS	30
18. ERP ET L'INTERNET	31
19. LA DEUXIEME GENERATION D'ERP	31
20. CONCLUSION.....	34

CHAPITRE II : LES SYSTEMES MULTI-AGENTS

1. INTRODUCTION	36
1.1. L'INTELLIGENCE ARTIFICIELLE DISTRIBUEE (IAD).....	36
1.2. LES SYSTEMES MULTI-AGENTS	37
1.2.1. Qu'est ce qu'un système multi-agents ?	37
1.2.2. Différence entre S.M.A et Système orienté objets.....	38
1.2.3. Quand utiliser un S.M.A?	39
1.3. CONCEPT D'AGENT	40
1.3.1. Qu'est ce qu'un agent?.....	40
2. L'ERP ET LES SYSTEMES MULTI AGENT	41
2.1. L'APPROCHE MAERP (MULTI AGENT BASED ERP SYSTEM).....	42
2.1.1. L'architecture du MAERP	42
2.2. LE PROJET PABADIS (PLANT AUTOMATION BASED ON DISTRIBUTED SYSTEMS).....	44
2.2.1. Le modèle PABADIS	46
2.2.2. Fonctionnement du système PABADIS.....	48
2.2.3. La communication à l'intérieur du système PABADIS	49
2.3. LE PROJET EXPLANTECH	51
2.4. L'ARCHITECTURE DU SYSTEME PROPLANT.....	52
2.4.1. Le système ExPlanTech.....	54
2.5. SYNTHESE	55
3. CONCLUSION.....	58

CHAPITRE III : MODELISATION

1. INTRODUCTION	61
2. PRESENTATION GENERALE DU MODELE PROPOSE	62
3. PRESENTATION DETAILLEE DES AGENTS DE NOTRE MODELE	63
3.1. AGENT ACCUEIL (AC)	63
3.1.1. Architecture de l'agent accueil	64
3.1.2. Fonctionnement de l'agent accueil	65
3.1.3. Le savoir de l'agent accueil	66
3.2. L'AGENT VENTE	66
3.2.1. Architecture de l'agent vente.....	67
3.2.2. Fonctionnement de l'agent vente	68
3.2.3. Le savoir de l'agent vente.....	68
3.3. AGENT GS.....	69
3.3.1. Architecture de l'agent GS	69
3.3.2. Fonctionnement de l'agent GS	70
3.3.3. Le savoir de l'agent GS	71
3.4. L'AGENT PLANIFICATION	71

3.4.1. Architecture de l'agent planification	71
3.4.2. Fonctionnement de l'agent planification	72
3.4.3. Le savoir de l'agent planification	72
3.5. L'AGENT GRH	73
3.5.1. Architecture de l'agent GRH	73
3.5.2. Fonctionnement de l'agent GRH	74
3.5.3. Le savoir de l'agent GRH.....	75
4. LE MECANISME DE COMMUNICATION ET DE CONTROLE UTILISES	75
5. FONCTIONNEMENT GENERAL DU SYSTEME	76
6. CONCLUSION.....	78

CHAPITRE IV : ETUDE DE CAS ET VALIDATION

1. INTRODUCTION	81
2. ENVIRONNEMENT DE DEVELOPPEMENT.....	81
2.1. PRESENTATION DE LA PLATEFORME JADE.....	81
2.1.1. L'environnement d'exécution JADE	83
2.1.2. La communication entre les agents JADE	83
3. ETUDE DE CAS	85
3.1. PRESENTATION GENERALE DE L'ENTREPRISE	85
3.2. FONCTIONNEMENT DE L'ENTREPRISE ET APPLICATION DE L'APPROCHE	85
3.2.1. Diagramme de classe UML de la base de données.....	85
3.2.2. Les agents de l'entreprise	87
4. RESULTAT	99
4.1. L'AGENT AC	99
4.2. L'AGENT VENTE	101
4.3. L'AGENT GRH	102
4.4. L'AGENT VENTE GS	102
4.5. L'AGENT PLANIFICATION	104
5. EXEMPLE ILLUSTRATIF.....	113
6. L'AGENT SNIFFER	115
7. CONCLUSION.....	116
CONCLUSION GENERALE	119
REFERENCES.....	120



INTRODUCTION
GENERALE

Les systèmes des entreprises sont un ensemble d'outils de système d'information permettant la circulation de l'information à l'intérieur de l'organisation. L'accès des entreprises aux nouvelles technologies et donc l'intégration d'outils basés sur les technologies de l'information et la communication au sein de l'entreprise, ont conduit vers l'apparition des Enterprise Resource Planning (ERP), qui présente un outil permettant une gestion homogène et cohérente du système d'information de l'entreprise, en particulier pour la gestion commerciale de la chaîne de production jusqu'à la vente d'un produit. Ils sont conçus pour contrôler potentiellement les centaines de fonctions dans une grande organisation, Ils sont des outils utiles pour la centralisation des opérations et la prise de décision, et ils facilitent la communication entre les différents modules.

N'empêche qu'ils souffrent de certaines difficultés dans leur mise en œuvre qui nécessite un énorme volume de travail, aussi la plupart exigent un certain niveau de reconfiguration ce qui représente un risque pour la stabilité du système de l'entreprise, tous cela est due à la complexité de l'organisation de l'entreprise et au taux gigantesque d'informations.

Une branche de l'Intelligence Artificielle Distribuée consiste à ce que les composants possédant une certaine autonomie, doivent être dotés de capacités de perception et d'action sur leur environnement, on parle alors d'agents et par conséquent de systèmes multi agents. Ces systèmes deviennent indispensables dans plusieurs domaines d'applications dans le but de résoudre les problèmes de complexité surtout quand il s'agit de grand système tel que ceux des entreprises.

L'objectif de ce travail est d'intégrer la technologie multi agents dans les systèmes ERP pour manier à ces problèmes. Pour ce fait notre mémoire commence par un chapitre qui introduit la notion de l'ERP tout en commençant par un historique expliquant les étapes qui ont conduit à l'apparition de ce genre de logiciel, et les différentes approche antécédentes. En suite nous avons présenté les caractéristiques de ces systèmes, et nous avons illustrer et détailler leurs architecture de base. Aussi nous avons exposé leurs types ainsi que les avantages et les inconvénients qu'ils présentent.

De plus nous avons données une idée sur le marché des ERP, et nous avons pris quelques exemples des ERP les plus populaires, et quelques conseils pour le bon choix. Cela est suivi par les étapes de la mise en œuvre de ce genre de logiciel dans une entreprise et aussi les méthodes d'intégration existantes.

Et nous avons conclu ce chapitre après l'explication des différentes approches récemment parues pour conduire les ERP vers une nouvelle génération.

Le deuxième chapitre est destiné aux systèmes multi agent, il est composé de deux parties ; la première constitue une vue générale sur les systèmes multi agent et le concept d'agent en commençant par une définition de l'intelligence artificielle.

Dans la deuxième partie nous avons expliqué la nécessité et l'avantage de l'adoption de l'approche multi agent pour les systèmes ERP. Nous avons aussi exposé quelques travaux dans le même cadre de recherche, pour conclure avec une synthèse où l'on a expliqué et donné notre avis sur ces travaux et montré la différence entre eux et l'approche que nous avons proposée.

Le troisième chapitre contient les détails du modèle que nous avons proposé pour une approche d'intégration d'agent dans les systèmes ERP, il contient aussi l'architecture que nous avons adoptée pour chaque composant de notre système.

Le quatrième chapitre contient la validation du modèle proposé, et la présentation des différents outils utilisés dans ce but.

Nous finissons notre mémoire par une conclusion générale qui contient les perspectives possibles qui constituent une suite de recherche pour notre projet.



Chapitre 1
Généralités sur les
ERP

1. Introduction

Les changements qui ont affecté l'économie mondiale ces dernières décennies ont permis une concurrence entre les entreprises qui est traduite par une amélioration croissante de la productivité. Cette amélioration à rendu le marché plus exigeant en terme de qualité, de flexibilité, de prix, et de délai. Afin de prendre en considération ces facteurs les entreprises se sont orientées vers de nouvelles techniques en s'appuyant sur les technologies de l'information et de la communication, on ne parle plus de systèmes d'information classiques mais d'ERPs qui sont des outils par excellence pour le pilotage des systèmes opérationnels des entreprises [1] [2].

2. Historique

L'informatique de gestion a subi des bouleversements considérables vers des systèmes d'information qui prennent en charge des niveaux de gestion de plus en plus stratégiques, ce qui s'explique par le passage par plusieurs phases qui sont :

** Informatique d'entreprise (des années 60 aux années 70)*

Les applications informatiques à cette période avaient pour rôle l'automatisation des processus opérationnels et répétitifs, et les systèmes de gestion avaient obtenus une puissance de traitement qui leurs permettait de traiter les données avec grandes quantités.

** Informatique de département (les années 80)*

C'est l'époque d'apparition des minis systèmes, où les entreprises commençaient à s'orienter vers les PME (Petites et Moyennes Entreprises) ce qui se matérialise par des applications permettant l'accès aux données afin de couvrir les besoins les plus spécifiques de l'entreprise, grâce à des interfaces textuelle fournissant une interaction utilisateur-système.

** Informatique individuelle (les années 90)*

À l'arrivée des ordinateurs personnels et les nouvelles technologies, le défi majeur devenait le partage de l'information et la répartition des applications sur les différents terminaux d'une entreprise. En effet l'architecture en réseaux et le client serveur ont trop apporté aux systèmes de gestion de l'entreprise.

* *Internet (les années 2000)*

Puisqu'elles permettent l'ouverture sur le monde et facilitent la communication inter et intra entreprises, les technologies de l'Internet deviennent des normes pour la mise en place des systèmes informatiques d'entreprise (*intranet*), ainsi que pour les systèmes informatiques interconnectés avec les partenaires (*extranet*). L'Internet a permis la répartition du traitement de l'information, mais aussi le partage de l'information elle-même afin de faciliter la tâche et augmenter la performance des applications de gestion. [3]

Toutes ses étapes ont conduit à l'apparition et au développement de ce que nous appelons aujourd'hui les ERP.

3. Approches et évolution

Afin de répondre aux besoins de la gestion industrielle, les systèmes d'information ont subi de grandes étapes de transformation, ce qui a été matérialisé par l'apparition de plusieurs approches de gestion au fil du temps, la figure qui suit illustre les approches existantes et qui ont précédé la naissance de l'ERP :

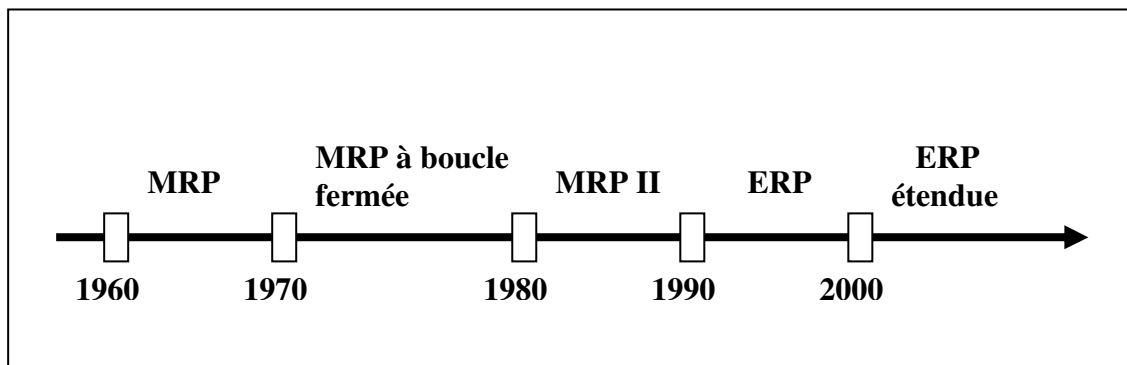


Figure1 : l'évolution des approches de gestion industrielle

Nous commençons alors par les années 60, et la première approche considérée comme antécédente de l'ERP ; nous parlons bien sûr du MRP :

3.1. L'approche MRP (Material Requirements Planning)

Le terme MRP pour *Material Requirements Planning*, ou en français *planification de ressource de production* désigne une méthode de gestion de production et d'approvisionnements informatisée, qui permet d'assurer le calcul des besoins et d'établir une planification des délais dans le but d'aboutir à des dates d'exigibilité correctes et pouvoir les maintenir. En une expression un MRP a pour objectif de transformer les données commerciales relatives aux ventes en données de production. [4]

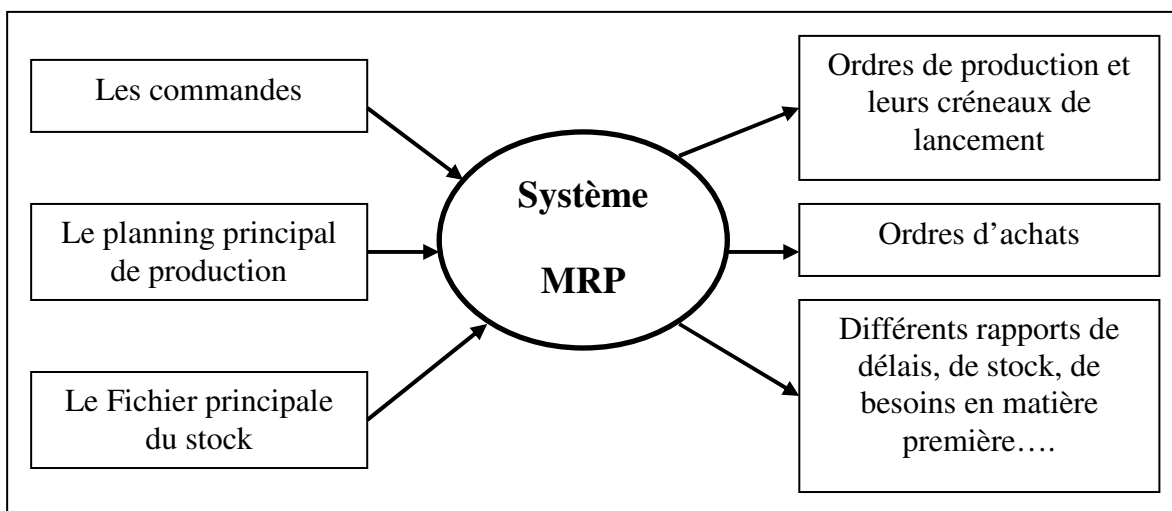


Figure2 : Vue d'ensemble du fonctionnement d'un MRP

Comme le montre la figure ci-dessus, le MRP est une simple automatisation d'un processus de gestion permettant à partir : d'informations sur la manière de production, des commandes des clients et des éléments stockés, de générer des rapports sur les besoins, de bien gérer le stock et de décider des créneaux de lancement des ordres de fabrications des produit et ceux d'achats de matières premières s'il on a besoin.

3.2. MRP à boucle fermée

Le MRP à boucle fermée est un système bâti autour d'un MRP, mais en plus, il implique une gestion de nombreuses fonctions dans le processus de planification, et permet l'emploi de feedbacks pour améliorer la précision du plan. Lorsque le MRP a été utilisé à l'origine, les plans matériaux ont été lancés pour une période déterminée après cette période, un tout nouveau plan sera élaboré et lancé. Ce processus a été répété sans cesse, mais il n'y avait pas de boucle de rétroaction pour pouvoir juger si un plan est réalisable et s'il a été effectivement réalisé. Développer un système en circuit fermé implique la vérification de l'application des plans de production et de la disponibilité des ressources. Par conséquent, la capacité est vérifiée tout au long du processus permettant aux plans proposés d'être révisé en cas de besoin. Le schéma qui suit récapitule le fonctionnement d'un système MRP à boucle fermée : [5]

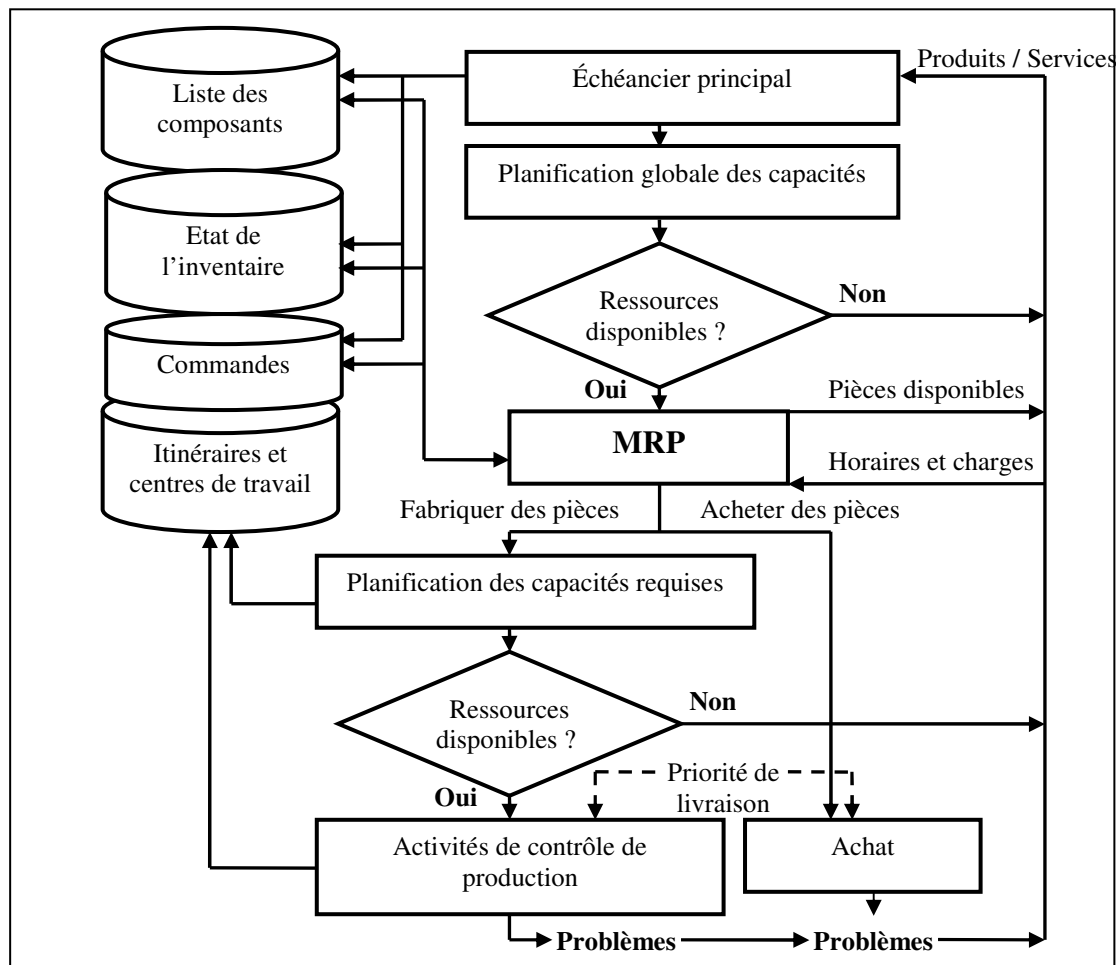


Figure3 : schéma représentatif du fonctionnement général d'un MRP à boucle fermée

3.3. L'approche MRPII

L'approche MRPII acronyme de *Materials Resource Planning*, née dans les années 80, combine le haut niveau de développement auquel la planification en gestion industrielle est arrivée, avec l'approche MRP à boucle fermée, tout en intégrant d'avantage de fonctions ; telles que le suivi de production, le calcul des coûts, et la planification à long terme, moyen terme, court terme, et pourquoi pas en temps réel aussi.

3.4. L'approche ERP

Les approches MRP, MRP à boucle fermée et MRPII ont été matérialisées par des solutions logicielles appelées systèmes de GPAO (Gestion de Production Assistée par Ordinateur), qui sont des outils composés de fonctions importantes du processus global de gestion : la gestion des approvisionnements, la gestion des stocks et la planification. Cette famille d'outils a su évoluer au cours du temps pour introduire des capacités de planification hiérarchisée, puis des liens vers des ressources multiples, aussi il existait d'autres logiciels tournant dans le même cadre, appelés CFAO (Conception et Fabrication Assistée par Ordinateur) [6]. L'évolution de ces outils a conduit à la naissance de nouveaux logiciels, et donc une nouvelle approche qui a fait surface.

L'approche ERP a pu toucher une nouvelle caractéristique celle des processus transversaux dite aussi organisation horizontale, elle est dotée d'une large couverture fonctionnelle, et sa gamme de traitements d'information possède une homogénéité qui lui permet de gérer des ressources de nature différente en maintenant la cohérence.

Cette approche est fondées sur le même principe que l'approche MRPII, mais elle est plus puissante car :

- Elle est basée sur un ensemble unique d'outils de planification à travers une entreprise entière.
- fournit l'intégration en temps réel des données de ventes, d'exploitation, et de finance.
- étend les approches de planification des ressources aux applications qui prend en charge clients et fournisseurs.

La figure 4 illustre l'approche ERP et son fonctionnement : [7]

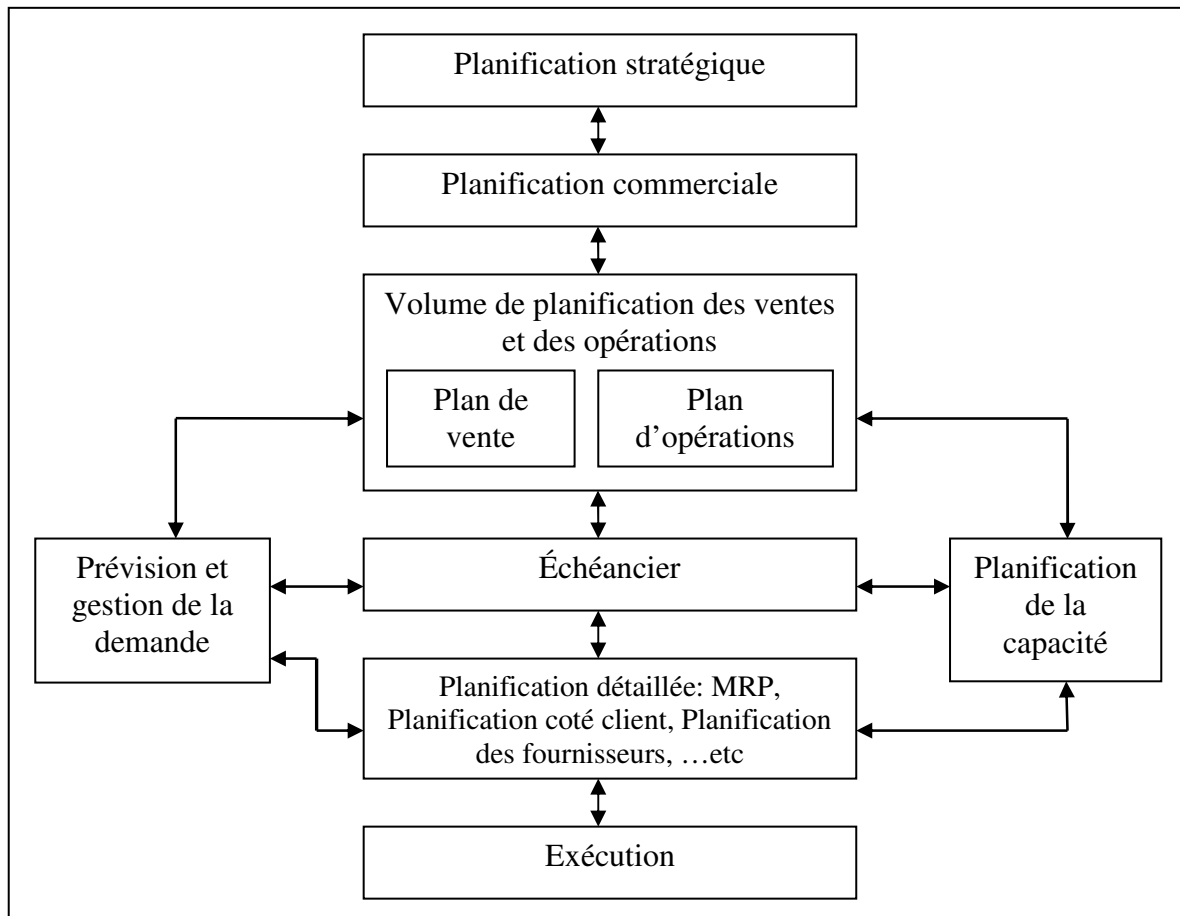


Figure4 : fonctionnement générale d'un système bâtis autour d'une approche ERP.

Dans le reste du chapitre nous allons étudier avec plus de détail les logiciels ERP.

4. Définition de l'ERP

L'**ERP** c'est l'abréviation de l'expression anglaise *Enterprise Resource Planning*, appelée en français *Progiciels de Gestion Intégrés (PGI)*, qui provient du nom de la méthode **MRP** (*Manufacturing Resource Planning*) utilisée par les entreprises manufacturières pour la gestion de la production. Cette expression comme nous le constatons est formée de trois parties :

- Le terme "progiciel" qui est issu de la composition des deux mots « produit » et « logiciel », il désigne un logiciel commercial vendu par un éditeur sous forme d'un produit complet. Ce terme a été créé en 1973 par Jean-Erick Forge, et il a fini par prendre la définition suivante en 1974 : « Ensemble comprenant un programme, les jeux d'essais, la documentation correspondante et susceptible d'être fourni à plusieurs utilisateurs ». [8]
- Le terme "gestion" rend le progiciel en question dédié à l'informatique de gestion, c'est-à-dire que les programmes de ce progiciel remplissent des fonctions de gestion de l'information propre à une entreprise.
- Le terme "intégré" fait allusion à ce que le système soit implémenté au sein d'une organisation ou une structure spécifique.

La composition de ces trois parties désigne un système composé de plusieurs programmes dont chacun effectue une activité parmi celles nécessaires au fonctionnement d'une entreprise, en d'autres termes, il permet de gérer l'ensemble des flux d'informations et des processus propre à une entreprise.

Nous pouvons aussi définir l'ERP comme étant un logiciel ayant pour rôle l'automatisation des fonctions de l'entreprise depuis l'étude de faisabilité jusqu'à la livraison du produit.

En effet, il existe plusieurs définitions de l'ERP celle qui semble la plus complète est la définition donnée par Willis et al en 2003 : « L'ERP est un système intégré qui permet à l'entreprise de standardiser son système d'information pour relier et automatiser ses processus de base. Il fournit aux employés les informations nécessaires pour diriger et contrôler les activités essentielles de l'entreprise le long de la chaîne logistique, de l'approvisionnement à la production ou l'exploitation jusqu'à la vente et à la livraison au client final. Les employés n'entrent qu'une seule fois les informations, qui sont alors mises à la disposition de tous les systèmes de l'entreprise. » [9].

5. Pourquoi l'ERP ?

Les applications informatiques classiques qui avaient pour rôle l'automatisation des entreprises, avaient rencontré plusieurs problèmes d'incohérence, ceci revient à leurs structures qui s'appuyaient sur l'isolement des fonctions de l'entreprise afin d'éviter les conflits d'interfaçage. Or les fonctions d'une entreprise sont complémentaires et agissent sur le même ensemble de données, puisqu'elles communiquent afin d'accomplir un objectif globale qui est celui de l'entreprise elle-même. Les ERP ont relevé se défi en fournissant une structure qui maintient la nature de ces fonctions par le biais d'une approche globale autour d'une base de données centralisée [10].

6. Les caractéristiques d'un ERP

Un ERP regroupe un ensemble de caractéristiques parmi lesquelles nous citons :

- Le fait qu'il soit un progiciel il est :
 - Générique. [10]
 - Possède un très fort niveau de paramétrage ce qui lui permet d'être ouvert sur le monde de l'entreprise. [10]
- L'une de ces caractéristique majeures est le fait qu'il est intégré, par conséquent il:
 - Evite et gère la redondance de données ce qui assure la cohérence. [11]
 - Permet l'adjonction de nouvelles fonctions indépendamment de l'architecture des données. [11]
 - Fournit un certain degré de complémentarité entre les différentes unités du logiciel.
- Un ERP est aussi d'un périmètre fonctionnel large lui permettant de fournir une variété de services de gestion économique.
- Il est dédié à la mise en oeuvre des moyens de l'entreprise comme son nom le montre « entreprise ressource planning ».
- Il focalise plus sur la finalité des processus que sur les modes opératoires de gestion et de planification des ressources. [10]
- Les applications ERP concernent un processus administratif.
- Les applications ERP sont modulaires. [11]

7. L'architecture d'un ERP

Les ERP utilisent une architecture de type client/serveur. Ils comportent un nombre variable de modules fonctionnels partageant les mêmes données, Celles-ci sont contenues au sein d'une base de données centralisée et unique. Les modules communiquent et échangent en utilisant cette même base qui assure la cohérence de l'information. Les modules applicatif d'un ERP recouvrent toutes les fonctionnalités de l'entreprise celle destinée au client comme du côté fournisseur mais aussi celles qui sont en relation avec les partenaires.

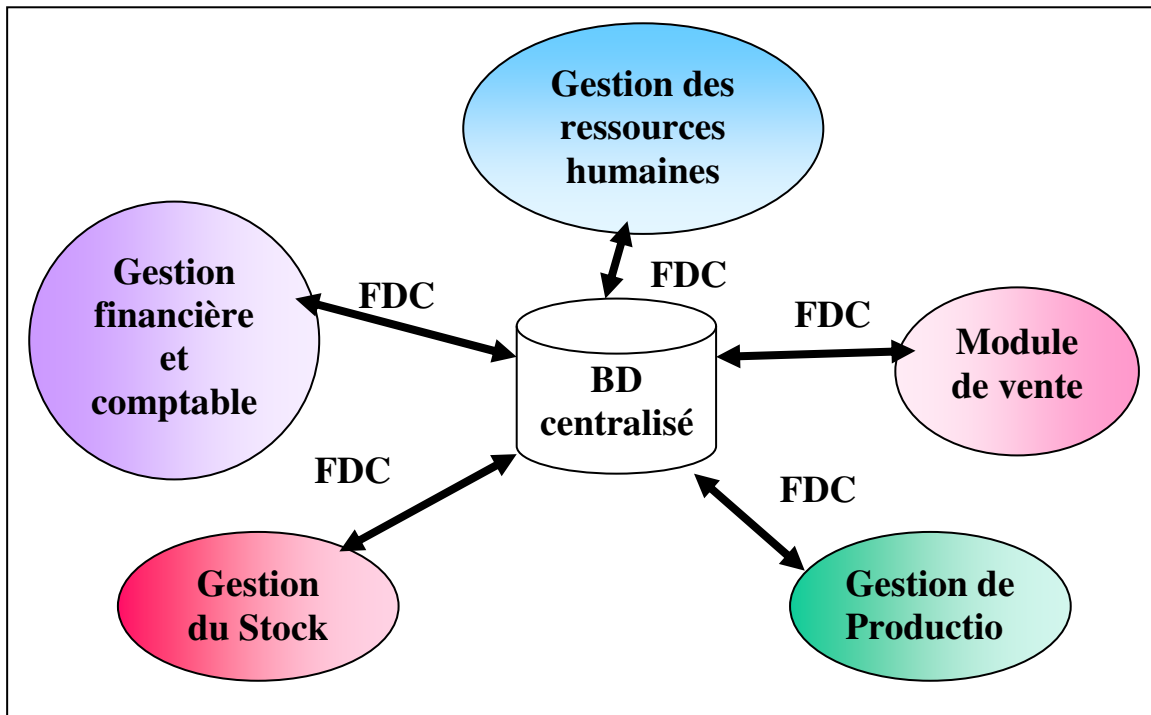
L'ERP possède une structure organisée selon deux axes d'organisation :

7.1. L'axe horizontal

Cet axe représente la classification des modules de l'ERP, selon Jean-Louis Lequeux un ERP est défini comme une application ayant au minimum les modules suivants :

- gestion commerciale.
- GPAO (Gestion de Production Assistée par Ordinateur).
- gestion comptable et financière.
- ressources humaines.
- gestion client.
- gestion planning.

Dans notre recherche bibliographique, nous avons rencontrés plusieurs classifications des modules de l'ERP mais elles sont toutes d'accord que les modules essentiels à tout système ERP sont ceux présentés dans la figure qui suit et qui est une représentation générale de l'architecture d'un ERP :



FDC : Flux de données et de Contrôles

Figure5 : Architecture générale d'un ERP

Cette architecture modulaire enrichi notre système et le rend :

- Évolutif.
- Extensible (nombre d'utilisateurs).
- Et adaptables : Paramètres, Langage propriétaire, Ouverture sur l'extérieur. [10]

Les modules qui sont des sous-système de l'ERP remplissent chacun une fonction de haut niveau, selon cette fonction ils peuvent être classés en trois domaines distincts :

- **Le domaine métier :** regroupe les modules responsables de la réalisation des processus situés sur la chaîne de valeur de l'entreprise, et qui représentent le territoire intérieur de cette dernière où les flux d'information sont directement corrélés aux activités de production et de logistique. tel que les modules de gestion des achats, de gestion de production, ou de gestion des stocks, dans une entreprise manufacturière.

- **Le domaine du pilotage** : les modules de ce domaine sont orientés vers l'analyse et la synthèse du fonctionnement du système de l'entreprise. Ils prennent en charge les fonctions qui facilitent l'accès à des prévisions, et ceux d'évaluation de la performance, ainsi que les activités financières et comptables, la relation client, les tableaux de bord du contrôle de gestion...etc.
- **Le domaine du support** : il s'agit des fonctions chargées de l'administration du système informatique. Ce domaine couvre l'entretien de la structure de données et la modification des paramètres de configuration de la solution logicielle. [6]

7.2. L'axe vertical

Où l'ERP semble organisé en trois couches d'après Baglin et al (2005), qui sont :

- **Le niveau présentation** : il constitue l'interface utilisateur, et donc toute tâche intervenant dans la communication avec ce dernier, afin d'acquérir les données nécessaires au traitement et lui fournir les résultats qu'il espère.
- **Le niveau applications** : il correspond aux fonctions de traitement de l'information.
- **Le niveau Base de données** : contient les procédures qui gèrent les grands volumes de données que peut avoir une entreprise, et les différentes manipulations et opérations à utiliser sur ces données.

Donc chaque module de l'ERP est constitué des trois couches qui sont : la couche présentation, la couche applications et la couche Base de données, ce qui donne la structure représentée dans le schéma suivant :

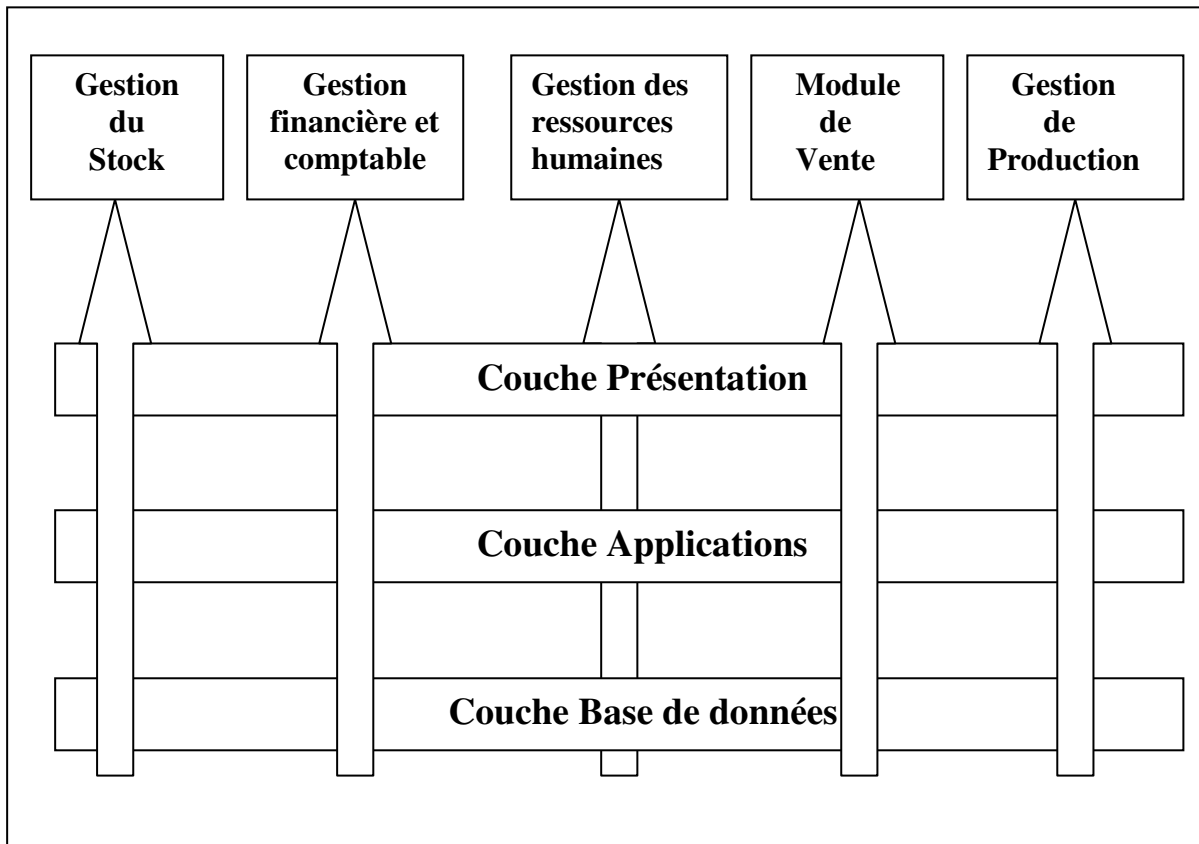


Figure 6 : schéma représentant la structure d'un ERP

8. Typologie des ERP

Une première classification nous permet de distinguer trois catégories d'ERP qui sont :
[12]

- **Les ERPs généralistes** : comme leur nom l'indique les ERPs qui appartiennent à cette catégorie, couvrent en globalité les processus d'une entreprise mais ne permettent pas de couvrir toutes les spécificités métier c'est-à-dire que ces ERPs ne sont pas suffisamment précis pour pouvoir répondre à tous les besoins d'un domaine particulier voire d'une entreprise particulière. Pour bien illustrer prenons une entreprise de produits alimentaires, elle repose évidemment sur beaucoup de contraintes réglementaires et techniques en fonction d'hygiène donc la gestion du stock, les délais de conservation et de livraison ainsi que la manière dont les produits sont gardés doivent être vérifiés soigneusement et avec

précision car ça risque de produire des dégâts au niveau de l'entreprise comme au niveau de l'humanité.

- **Les ERPs spécialistes** : Ils se focalisent sur le métier, et permettent une meilleure évolution et réactivité pour l'entreprise. Ces ERPs sont moins coûteux que ceux de la catégorie précédente qui nécessitent des développements spécifiques, difficiles à maintenir. Cependant les ERPs spécialistes reposent sur des règles métier propre au domaine d'activités de l'entreprise ce qui les rends plus adaptés aux besoins de cette dernière et plus souples à utiliser donc ils fourniront de meilleurs résultats.
- **Les ERPs verticaux** : un ERP vertical appelé aussi « best of breed » n'est pas tout à fait un ERP mais un progiciel de gestion ayant pour rôle de prendre en charge un domaine fonctionnel précis tel que la comptabilité par exemple, il nécessite des interfaces avec le reste des composants du système de l'entreprise ce qui rend son intégration compliquée et coûteuse. Ces progiciels sont considérés comme ERPs du fait qu'ils se comportent d'une manière répondant à une logique semblable à celle d'un ERP.

De ce fait une question se pose: *alors quelle catégorie est la meilleure ?*

La démarche best of breed est meilleure si l'on veut privilégier la performance de l'entreprise du fait qu'elle consiste à assembler les meilleurs composants, fonction par fonction (chaque composant est un produit best of bread ie : ERP vertical), de plus un produit spécialisé est toujours plus performant qu'un produit généraliste, oui un produit best of bread est plus spécialisé qu'un ERP spécialiste puisque la spécialisation chez le best of bread est au niveau de la fonction. Or cette démarche souffre de la difficulté lors de l'interfaçage, ça nécessite un grand effort et un coût élevé, elle souffre en plus de problèmes dues à l'isolement des fonctions, aussi l'optimisation du rapport efficacité/variété n'est pas évidente, ce qui rend la solution intégrée (ERP spécialiste) plus sûre, plus économique et plus efficace.

Mais la question qui se pose aussi; peut on apporter le même niveau de performance des best of breed aux ERPs spécialistes ? Oui c'est faisable grâce à l'organisation verticale de l'ERP que nous avons déjà décrit de plus l'approche que nous proposons et qui a pour objectif l'intégration de l'agent dans l'ERP fait en sorte de renforcer la performance de l'ERP et de prendre en considération chaque fonction en détaille et de la mettre en valeur, tout en facilitant les communication inter et intra fonctions.

Une deuxième classification qui s'appuie sur le critère de la taille et la nature de l'entreprise à qui le logiciel est destiné, donne les types d'ERPs suivants :

- **ERP Grand groupe** : un ERP grand groupe est un ERP doté en plus de modules périphériques, offrant la possibilité de paramétrage des processus et des règles de gestion ce qui permet une organisation de haut niveau, mais nécessitant des ressources d'assistance à Maîtrise d'ouvrage. Du fait qu'un grand groupe est un ensemble d'entreprises qui interagissent entre eux et qui sont complémentaires les uns les autres, il a besoin d'un ERP possédant des fonctionnalités multi site étendues pouvant partager les données de base. Ce type d'ERP possède une interface utilisateur multi langue, peut utiliser différents systèmes de gestion de bases de données, et il est compatible avec plusieurs plateformes.
- **ERP** : Progiciel de Gestion Intégré disposant d'une intégration financière automatique de tous les flux, mais ne disposant pas de modules périphériques, ses processus et ses règles de gestion sont le plus souvent câblés. Il possède des fonctionnalités multi site mais qui sont limitées et partagent des données limitées, il est multi langue, multi plateforme, et multi SGBD.
- **ERP PME/PMI** : Progiciel de Gestion Intégré (PGI) doté d'une GPAO et d'une comptabilité intégrées disposant au moins du journal des achats et des ventes automatique, ses processus et ses procédures sont figés, aussi il est mono langue, mono plateforme, et mono SGBD. Il n'a pas de fonctionnalités multi sites et donc nécessite une duplication de base de données [13].

Nous constatons alors, que la réalisation d'un progiciel ERP doit prendre en considération la nature de l'entreprise, ses besoins et le type de ses services, sans oublier sa taille ce qui veut dire le nombre de ses établissements et les relations qui les lie .

9. Avantages

Les progiciels de gestion intégrés permettent à l'entreprise une meilleure maîtrise de ses activités de production, tout en se basant sur une méthode stratégique garantissant l'optimisation de l'utilisation des ressources humaines et matérielles à la fois. Les objectifs d'un ERP se focalisent sur la réduction des coûts afin d'aboutir à la croissance de l'entreprise. Ainsi les ERPs présentent plusieurs avantages :

- ☑ la facilité de la gestion des flux d'informations dans une entreprise.
- ☑ La cohérence et l'homogénéité des informations.
- ☑ La réduction des coûts d'exploitation et des goulots d'étranglement.
- ☑ L'intégrité et l'unicité du système d'information.
- ☑ Différents niveaux de sécurité pour éviter l'utilisation non autorisée des données.
- ☑ l'optimisation des processus de gestion.
- ☑ Fournit un outil de configuration ayant pour rôle l'adaptation des services aux besoins spécifiques de chaque utilisateur
- ☑ la réduction des erreurs et des délais coûteux.
- ☑ Une communication interne et externe facile.
- ☑ minimisation des coûts d'interfaçage puisqu'ils ne contiennent pas d'interfaces entre les composants du progiciel.
- ☑ minimisation des coûts en temps d'exécution grâce à la synchronisation des traitements.
- ☑ minimisation des coûts de maintenance évolutive comme l'amélioration des fonctionnalités et l'évolution des règles de gestion, de même pour la maintenance corrective.
- ☑ l'augmentation de l'efficacité opérationnelle.
- ☑ La simplification de la planification stratégique.
- ☑ l'amélioration du service à la clientèle.
- ☑ Permet la portée d'une entreprise hors ses murs, jusqu'aux fournisseurs, aux clients et aux partenaires [14] [15] [16].

Malgré le taux d'avantages que fournit un ERP, n'empêche qu'il souffre de quelques inconvénients que nous allons présenter par la suite.

10. Inconvénients

Comme tout projet, les ERP souffrent de quelques problèmes du fait qu'ils sont difficiles et longs à mettre en œuvre; ils sont aussi relativement rigides et délicats à modifier, Par conséquent les inconvénients qui surgissent sont les suivants :

- ☒ Il est plus une boîte à outil puisqu'il ne peut fonctionner sans qu'il soit personnalisé
- ☒ coût élevé quoi qu'il existe quelques uns qui sont des logiciels libres, les coûts dans ce cas se réduisent dans le coût de formation des utilisateurs et le service assuré par le fournisseur.
- ☒ le progiciel est parfois sous-utilisé
- ☒ lourdeur et rigidité de mise en œuvre
- ☒ difficultés d'adaptation du personnel de l'entreprise à l'utilisation de l'ERP.
- ☒ nécessité d'une bonne connaissance des processus de l'entreprise
- ☒ nécessité d'une maintenance continue ce qui rend l'entreprise dépendante du fournisseur.
- ☒ le choix d'une solution ERP est souvent structurant pour l'entreprise et un changement vers un autre ERP peut être extrêmement lourd à gérer.

Donc, les systèmes ERP sont toutefois difficiles à gérer et à mettre en œuvre et peuvent même avoir des effets perturbateurs sur le fonctionnement d'une entreprise. Si le déploiement n'est pas proprement planifié et implémenté avec la vision et l'aide de professionnels, de telles initiatives peuvent entraîner une augmentation des coûts et un mécontentement général sans réels avantages pour l'organisation.

11. Le marché des ERP

Le marché mondial des ERP est dominé en 2006 par SAP, qui représente 43 % de la couverture en ERP. Oracle, ayant acquis PeopleSoft, représente 23 % du marché mondial [17].

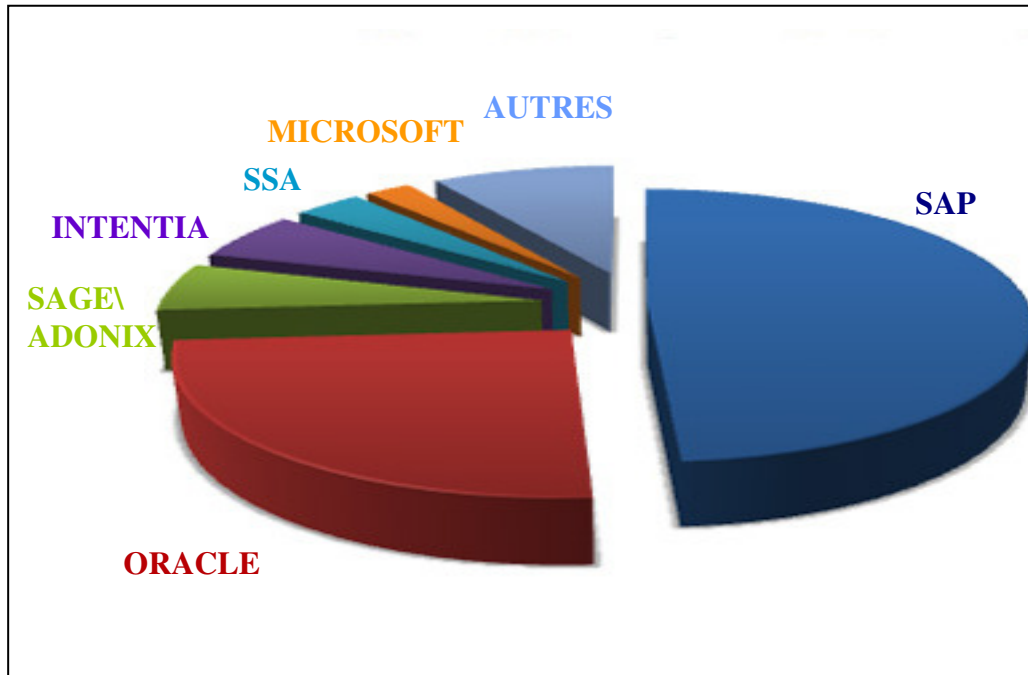


Figure 7 : les parts des éditeurs d'ERP dans le marché mondiale

Les ERP ont généré en Europe, Moyen-Orient et Afrique un chiffre d'affaires de 8,1 milliards de dollars en 2006, et l'estimation à l'horizon 2011 est de 11,6 milliards de dollars [17].

Donc, le marché des ERP repart à la hausse après une baisse relative. Il y a au moins 2 raisons à cela :

- le besoin de cohérence des systèmes d'information qui n'a jamais été si grand,
- les ERP ont beaucoup progressé et sont désormais, plus souples, plus facile à installer et plus performants.

Nous pouvons distinguer sur le marché deux types d'ERP ; les ERP propriétaires et les ERP open source :

11.1. Les ERP propriétaires

Nécessitent l'achat d'une licence, les principaux ERP de ce type sont les suivant : SAP (leader mondial), Oracle/Peoplesoft, SAGE ADONIX, Microsoft, SSA Global, GEAC, Intenia/Lawson, Infor Global Solutions.

11.1.1.SAP

SAP est le leader mondial du monde des ERP. Ce progiciel a remporté rapidement un succès important auprès des grandes entreprises en proposant un progiciel multilingue et multidevises. SAP s'intéresse aussi au marché des PME, en pleine croissance en proposant sa suite BusinessOne.

SAP est une application client-serveur. Ses modules couvrent l'ensemble des fonctions de gestion de l'entreprise et chaque module couvre des besoins complets de gestion. Les entreprises peuvent implémenter tous les modules fonctionnels de SAP, ou seulement quelques-uns. Ces modules sont de plusieurs sortes: des modules orientés logistique, finance, et ressources humaines.

11.1.2.Oracle - People Soft

La particularité de peoplesoft est qu'il utilise un serveur web pour faire le relais entre le client et le serveur d'application. De ce fait les postes clients nécessitent uniquement un navigateur Internet pour accéder aux applications. [18]

11.2. Les ERP open source

Sont gratuits, leur implémentation revient moins cher, puisqu'il n'y a pas de coût de licence. Mais les frais de maintenance et de l'assistance technique sont inclus dans le calcul du coût d'acquisition total. Les principaux ERP open source sur le marché sont : Aria, Compiere, ERP5, OFBiz (Open for Business), Tiny ERP,Dolibarr, SQL Ledger .

11.2.1. Compiere

C'est l'ERP open source le plus connu développé par Jorg Janke et all en 2001 et destiné au secteurs de distribution, et négociation. Il repose sur une architecture J2EE et utilise le système de gestion de base de données open source PostgreSQL. Cet ERP intègre les nouvelles technologies comme le workflow, la gestion des alertes, la messagerie électronique avec une gestion de pièces jointes, et une interface web.

Compiere est une solution Horizontale pour PME/PMI dotée de plusieurs modules, parmi lesquels un module de gestion des ventes (tarification, facturation, CRM), de gestion des achats, de gestion de stock et de logistique. La gestion comptable et financière est intégrée (comptabilité générale, analytique et budgétaire) et quelques fonctions de production existent déjà (gestion des ordres de fabrication, des nomenclatures...), et possède une interface multi langue [19].

11.2.2.ERP5

C'est un projet développé par la société Nexedi, fondée sur le serveur d'applications Zope. Il est développé dans le langage de programmation Python, basé sur la technologie web et une architecture d'ensemble objet-relationnel. Il intègre les fonctions de gestion de la relation client (CRM), gestion de production (MRP), gestion d'approvisionnement, gestion du processus de design (PDM), comptabilité, gestion du personnel et commerce électronique. [20]

L'architecture de l'ERP5 est une architecture d'ensemble objet-relationnel qui repose sur cinq concepts de base d'où le '5' dans son nom, ces concepts sont [15]:

- 1) **ressource** : décrit une ressource nécessaire pour la réalisation d'un processus d'affaires, tels que les compétences individuelles, les produits, les machines, ...etc.
- 2) **Nœud** : une entité qui reçoit et envoie des ressources. Elle peut être liée à une entité physique (comme les installations industrielles) ou à une entité abstraite (comme un compte bancaire). Il existe aussi les Méta-nœud qui sont des noeuds contenant d'autres noeuds.
- 3) **Chemin** : décrit comment un noeud accède à une ressource dont il a besoin d'un autre noeud. Par exemple, un chemin peut être une procédure de commerce qui définit la façon dont un client obtient un produit provenant d'un fournisseur.

- 4) **Mouvement** : décrit un mouvement de ressources entre les nœuds à un moment donné et pour une période bien déterminée. Par exemple, un tel mouvement peut être l'expédition des matières premières provenant de l'entrepôt à l'usine. Les mouvements sont des réalisations de Chemins.
- 5) **Item** : une instance unique d'une ressource. Par exemple, une RAM est une ressource pour l'assemblage d'un ordinateur, tandis que la RAM numéro 56982 est un item [15].

11.2.3. Dolibarr

C'est un ERP destiné aux petites et moyennes entreprises, ses principaux secteurs sont la gestion des commandes, la livraison et la facturation, il est construit avec la technique de développement PHP et le système de gestion de base de données MySQL [20], formant une application Web, et donc peut être utilisés de n'importe quel endroit avec un accès Internet.

Dolibarr combine l'approche ERP avec l'approche CRM (Customer Relationship Management), et il est le premier ERP / CRM qui maintient les règles suivantes :

- 1) Un logiciel doit être simple à mettre au point.
- 2) un logiciel doit être simple à installer.
- 3) un logiciel doit être simple à utiliser. [15]

11.2.4. SQL Ledger

C'est un progiciel destiné à la comptabilité, il fonctionne grâce à l'outil open source Perl et le système de gestion de base de données PostgreSQL. Il est multi-sociétés, multi-utilisateurs, et multi-langues. [20]

12. Comment faire le choix d'un ERP

Comme nous l'avons présenté précédemment ils existent sur le marché une variation de produits ERP, donc quel est le meilleur ?

Evidemment on ne peut pas vous dire qu'une solution logicielle est meilleure que d'autres, car tout dépendra de la nature de l'entreprise qui va l'utiliser. Le choix de la solution

logicielle est vraiment un facteur critique qui va affecter la réussite d'un projet ERP, c'est pour ça qu'il faut faire très attention. Voici les étapes qu'il faut suivre pour pouvoir choisir l'ERP le mieux adapté à l'entreprise :

1) Identifier le contexte

- Identifier la vision de la direction.
- Définir les rôles et les responsabilités de chacun des acteurs.
- Définir les priorités du projet et effectuer un planning des étapes à réaliser.
- Définir les objectifs du projet.
- Analyser le budget alloué.

2) Etudier les exigences

- Analyser les forces et faiblesses du système existant.
- Analyser les modes de fonctionnement de l'entreprise.
- Analyser le processus business (organisation, flux de données).
- Définir précisément les besoins.
- Traduire les besoins en fonctionnalités et performance.
- Appréhender les risques en cas de compromis.

3) Explorer le marché

- Comparer et évaluer les divers ERP envisageables.
- Effectuer une présélection sur la base de critères correspondant aux besoins jugés prioritaires (fonctionnels, techniques, budgétaires).

4) Réaliser des appels d'offres

- Rédiger un cahier des charges.
- Rédiger une grille d'analyse des offres.

5) Sélectionner :

- Analyser les offres.
- Evaluer le logiciel en utilisant des critères comparatifs.
- Etablir un scénario d'essai.

13. Les étapes de mise en place d'un progiciel de gestion

Le choix d'un ERP débute par une analyse comparative des offres du marché, qui répondent aux besoins de l'entreprise. Une fois le choix effectué, les principales étapes de mise en place sont les suivantes :

- 1) Analyse des fonctionnalités et des avantages qu'offre cet ERP par rapport au système d'information existant de l'entreprise, ce que va engendrer
 - des adaptations organisationnelles des modes de gestion de l'entreprise pour se conformer aux spécifications du progiciel
 - des demandes d'évolution ou d'aménagements du progiciel formulées à l'éditeur (*adapter* le progiciel) interne,
 - des développements d'interfaces entre le progiciel et des applications périphériques.
- 2) Définition du paramétrage nécessaire aux différents modules du progiciel.
- 3) spécifications des évolutions et spécifications des interfaces.
- 4) spécifications des règles de reprise et de transcodification des données.
- 5) réalisation puis tests des traitements.
- 6) préparation des actions de conduite du changement :
 - formation des personnels,
 - communication auprès du personnel, des clients, des partenaires ou fournisseurs,
 - révision des processus et de l'organisation ;
- 7) passage définitif sur le nouveau progiciel [8].

14. Méthodes d'intégration des ERP

Il existe trois méthodes permettant d'intégrer un ERP au sein d'une entreprise, afin de garantir le passage définitif de l'ancien système d'information vers le nouveau progiciel. Ces méthodes sont [21]:

14.1. La méthode du Big Bang

C'est la plus ambitieuse et la plus difficile des méthodes d'intégration d'ERP, elle consiste à basculer en une fois, tous le système vers l'ERP qui va être installer sur toute l'organisation.

- ✓ *L'avantage de cette approche:* temps d'implémentation réduit.
- ✗ *L'inconvénient de cette approche:* est le risque d'un blocage de l'entreprise, et la nécessité de nombreuses corrections.

14.2. La méthode de Franchise

Elle est destinée aux grandes entreprises composées de plusieurs établissements qui ne partagent pas de nombreux processus d'affaires. Cette méthode consiste à choisir l'un de ces établissement pour y installer l'ERP, une fois rassuré que le système est stable nous passerons à un autre établissement de l'entreprise et ainsi de suite jusqu'à ce que toute l'entreprise soit migrée vers l'ERP, tout en liant les processus communs des différent établissements.

- ✓ *L'avantage de cette approche:* la stabilité du système avant de mettre en place l'ERP dans toute l'entreprise, ce qui minimise le taux des correction par rapport à la méthode précédente.
- ✗ *L'inconvénient de cette approche:* elle prend beaucoup de temps, et l'existence de plusieurs systèmes au sein de l'entreprise jusqu'à ce que la bascule soit complète.

14.3. La méthode de mise en place par module

Cette méthode est destiné généralement aux petites entreprises, dans ce cas on intègre les modules un par un en commençant par le module le plus nécessaire càd celui qui est relatif aux fonctions pour les quelles l'entreprise à fait appel à l'ERP.

- ✓ *L'avantage de cette approche:* peux de corrections à effectuer.
- ✗ *L'inconvénient de cette approche:* nécessite de multiples interfaces avec les logiciels encore en cours.

15. L'implantation multi sites de l'ERP

L'implémentation multi sites d'un ERP peut prendre des formes très différentes, parmi les quelles nous citons :

15.1. L'autonomie locale totale

Se base sur le principe de doter les Sous unités de l'entreprise d'une autonomie de décision presque totale, cette forme permet le pilotage de différents systèmes ERP pour les différentes plans donc unités de la même entreprise.

✓ *Avantages :*

- évite les conflits associés aux changements établis au niveau des sous unités de l'entreprise.
- permet aux entreprises de poursuivre leurs futures acquisitions et cessions loin des complications systèmes.
- réduit le risque de défaillance d'implantation d'un projet ERP.

✗ *Inconvénients :*

- ne parvient pas à capter le potentiel des systèmes ERP pour intégrer les données, les systèmes et les processus à travers les lieux et les unités d'affaires.

15.2. Contrôle des sites seulement au niveau financier

Les sous unités de l'entreprise ont toute une autonomie dans tous les domaines, à l'exception du domaine de la comptabilité financière, ce qui nécessite un contrôle à ce niveau.

L'efficacité de cette méthode est majeur dans lorsque les sous unité de l'entreprise effectuent des tâches différentes.

15.3. Coordination des opérations entres les sites

Cette forme est caractérisé par un degré élevé d'autonomie locale dans les opérations, mais les différents sites conservent la capacité de gérer la chaîne d'approvisionnement globale grâce à l'accès aux informations locales concernant : les achats, les inventaires, et les échéanciers de production.

Cette forme est plus efficace quand il y a des avantages à obtenir à partir d'achats communs, ou bien si l'entreprise possède des clients aussi bien globaux que régionaux.

15.4. Coordination réseaux des opérations

Les opérations locales peuvent accéder à informations des autres sites, Pas un degré élevé de centraliser, mais

15.5. La centralisation totale

Un haut degré de centralisation dans cette méthode le tout est implémenté localement [7].

16. L'ERP étendue vers une nouvelle génération

Au début des années 2000, L'ERP s'est évolué et s'est étendu pour inclure de nouveaux modules relatifs à deux nouvelles approches : l'approche CRM (Customer Relation Management) et l'approche SCM (Supply Chain Management).

▪ **l'approche CRM**

Le Customer Relationship Management ou la Gestion de la Relation Client est un module tourné vers le client qui vise à améliorer la communication entre l'entreprise et ses clients, en automatisant les différentes composantes de la relation client. Il assure l'analyse des informations collectées sur le client pour permettre à l'entreprise de revoir sa gamme de produits afin de répondre au mieux à ses attentes. Il fournit en plus des outils de pilotage qui permettent une assistance dans la gestion des prises de contact, l'élaboration des propositions commerciales, et aux clients.

Mais, le CRM présente une difficulté d'intégration fonctionnelle avec les autres modules ainsi que la difficulté d'obtenir une coopération entre les différents services de l'entreprise vu que tous ces services sont concernés par le client.

▪ **La SCM**

La Supply Chain Management ou la gestion globale de la chaîne logistique est un mode d'organisation qui permet à l'entreprise selon Deixonne (2001) de :

- comprendre la demande du marché et de sélectionner les produits et les services de distribution.

- gérer l'équilibre entre la demande que l'entreprise veut satisfaire et sa capacité d'y répondre
- Maîtriser la logistique fournisseur, les achats, la production, les ventes, le stockage, la logistique client, la distribution, le service après vente afin de maîtriser l'ensemble des processus.

La SCM est basée sur une série de technologies comme l'EDI (échange de données informatisées) et l'APS (Advanced Planning and Scheduling) :

- **L'EDI** : est la transmission informatisée de documents commerciaux entre des systèmes d'application de partenaires d'affaires indépendants, à travers les réseaux de communication sous un format conventionnel standardisé .
- **L'APS** : est un système informatique qui permet de planifier à l'avance l'ensemble des flux physiques et financiers de l'entreprise [22].

Aussi avec l'évolution des nouvelles technologies, et avec l'apparition de la notion de processus métier, l'ERP doit se transformer. Il ne doit pas seulement être cohérent, il doit devenir réactif. Pour cela l'ERP adopte donc parfois :

- soit, une structure de type EAI, organisée autour d'un middleware assurant la circulation entre les modules,
- soit, se dote d'outils de supervision et d'optimisation horizontaux, de type workflow.

D'autre part, l'utilisation de l'Internet, et les modes d'interconnexion dans un contexte de commerce collaboratif offre d'avantages de services à l'ERP.

Les applications d'ERP, trouvent aujourd'hui une nouvelle valeur ajoutée dans leur capacité de gérer l'ensemble des processus de gestion entre l'entreprise, ses partenaires et ses clients. A cette fin, les ERP ont évolué vers des suites eBusiness intégrées en s'enrichissant de nouvelles fonctionnalités complémentaires telles que la Business Intelligence, ou encore plus récemment, le e-Procurement, le e-Commerce, e-MarketPlace.

Marketplace (place de marché d'Internet) : désigne un site où les entreprises peuvent échanger des produits et services. On peut distinguer les places de marché verticales consacrées à un secteur d'activité ou à une filière, les places de marché horizontales dédiées à une fonction par exemple le transport, et les sites d'appels d'offres généralistes traitant surtout des achats généraux [26].

E-PROCUREMENT : ce terme générique qualifie différents aspects de la gestion administrative des achats de l'entreprise, lorsqu'elle recourt de diverses façons au commerce électronique sur Internet, aux services d'appels d'offres en ligne, aux places de marché virtuelles, etc [23].

Cette évolution fonctionnelle accompagnée du développement technologique a transformé les systèmes ERP, initialement bâties sur le mode client serveur, en une offre applicative qui évolue aujourd'hui vers une logique de portail d'entreprise, intégrant des technologies de l'Internet complexes reliant entre elles des bases de données, applications transactionnelles et collaboratives très sophistiquées, ce qui conduit les ERP vers une nouvelle génération « les ERP II ».

17. ERP et WFMS

Les ERP et les WFMS (WorkFlow Managment Systems) sont deux classes de systèmes qui se focalisent sur le traitement des processus d'affaires, mais chacune d'entre elle adopte une approche différente.

Un WFMS est basé sur le paradigme de la spécification et l'exécution de processus. Donc en premier lieu, un modèle WorkFlow est élaboré afin de préciser l'organisation des processus de l'entreprise, puis des instances de ce WorkFlow seront créés pour effectuer les étapes décrites dans le modèle. Pendant l'exécution du workflow, l'instance peut accéder aux bases de données, aux applications, et peut interagir avec les utilisateurs.

Pendant qu'un ERP est mis en œuvre autour de l'idée d'applications préfabriquées, dans ce cas, Le plus de paramètres une application ERP possède, invoquera plus de flexibilité dans la configuration des processus de l'entreprise. Toutefois, le modèle de WorkFlow dans les systèmes ERP n'est pas explicitement spécifié parce qu'il est intégré dans les applications.

La différence principale entre le WFMS et l'ERP est ; dans les systèmes ERP, *la logique de flux* et la *logique de fonction* sont à la fois intégrés dans des applications. En revanche, un WFMS sépare les deux explicitement, *la logique de flux* est prise en compte par un modèle de

workflow, et la *logique de fonction* est capturée dans les applications, des données et du modèle que les personnes invoquent. Ainsi, un WFMS permet aux développeurs de séparer les flux entre les composants d'un système (applications, données, personnes) à partir du modèle de workflow.

Les Systèmes WFMS sont centrés sur les processus, l'accent étant mis sur la gestion des flux logique. D'autre part, les systèmes ERP sont centrés sur les données, ils se focalisent sur la gestion d'une *logique de fonction* via une infrastructure de données homogène à travers l'organisation, pour supporter de multiples applications.

Mais qu'elle est la différence entre une *logique de flux* et une *logique de fonction* ?

La logique de fonction : traite une tâche spécifique, comme la mise à jour d'un dossier client ou le calcul de l'ordre.

La logique de flux : combine de nombreuses fonctions dans une séquence afin de résoudre des problèmes plus complexes tels que le traitement d'une commande.

Les nouveaux ERP tendent vers l'intégration d'un WFMS au sein du système ERP pour améliorer ce dernier, cette approche a pour objectif de prendre les points forts des deux technologies, tout en intégrant des modules responsables du pilotage des processus d'affaires nous parlons ici de moteur workflow [24].

18. ERP et l'Internet

En tant que plate-forme d'intégration, Internet offre un potentiel considérable de services, en plus l'accès est largement disponible et les technologies Internet reposent sur des standards ouverts, ce qui a permis aux systèmes ERP de s'y adapter et alors profiter des bénéfices qu'elle peut offrir. En conséquence, les ERP deviennent basés sur le protocole HTTP, des serveurs Web, des serveurs d'applications, et les langages Java et XML, et utiliseront des interfaces basées sur un navigateur afin que les utilisateurs puissent accéder au système à partir de n'importe quel ordinateur connecté à Internet de n'importe quel endroit du monde.

19. La deuxième génération d'ERP

La nouvelle tendance qui est en train d'émerger et qui est connue sous le nom de la deuxième génération des ERP ou encore l'ERP II, est une re-mise en œuvre des ERP ; une nouvelle implémentation, qui a provoqué un éloignement des architectures client-serveur vers une nouvelle architecture basée Internet, où de nouvelles technologies y sont intégrées comme :

- E-Commerce (Electronic Commerce)
- M-Commerce (Mobile & Wireless Technologies)
- C-Commerce (Collaborative Commerce)
- Middleware
- Enterprise Portal Technologies
- Web Services
- Analytical Capabilities (Data Warehousing & Data Mining)
- CRM, SCM, SRM
- Knowledge Management
- Business Intelligence

Le passage de l'ERP à l'EPR II est due à un changement stratégique d'un système qui est totalement centrée sur l'entreprise et préoccupé par l'optimisation des ressources internes et de traitement transactionnel à un système reposant sur l'intégration des processus et la collaboration externe.

Les ERP de deuxième génération ont pour objectif de couvrir l'ensemble des grands processus de l'entreprise de bout en bout, ils se distinguent de leurs prédécesseurs par deux évolutions majeures :

- une couverture fonctionnelle élargie, et une nouvelle architecture technique orientée services qui repose sur des technologies EAI (Enterprise Application Integration) qui est un ensemble d'outils informatique permettant de faire communiquer les application de l'entreprise entre elles, et avec l'extérieur.
- les ERP II prennent désormais en charge de nombreuses nouvelles fonctions liées au Web services.

Du point de vue technique un saut c'est produit, mettant fin aux architectures monolithiques, les ERP II reposent sur une architecture orientée service, qui est composé de trois couches distinctes :

- Les modules qui composent le système ERP comme le module de finance, ou celui de ressource humaine ...etc, regroupent des services accessibles via une interface de programmation (API) standard telle que JCA ou SOAP. Avec la possibilité pour l'entreprise de rajouter ses propres applications grâce aux connecteurs EAI.

CHAPITRE I : GENERALITES SUR LES ERP

- La synchronisation des échanges entre les services s'appuie en effet sur un « bus EAI » traditionnel composé d'un courtier de messages et/ou d'un middleware orienté message.
- Un moteur d'orchestration chargé de lier l'ensemble des services selon des processus métier (achat, commande, etc.) [25].

Et voilà un tableau qui récapitule les principales différences entre l'ERP traditionnel et celui de la deuxième génération :

	ERP	ERP II
Rôle	ERP traditionnel a pour rôle l'optimisation interne de l'entreprise	permet d'optimiser la chaîne d'approvisionnement grâce à la collaboration avec les partenaires commerciaux.
Domaine	Elle est concentrée sur la fabrication et la distribution	traverse tous les secteurs d'affaire en relation avec l'entreprise.
processus	Dans les systèmes ERP, les processus ont été axés sur les quatre murs de l'entreprise	Systèmes ERP II se connecter avec les partenaires commerciaux, pour prendre ces processus au-delà des frontières de l'entreprise.
Architecture	Les anciens systèmes ERP ont été monolithique et fermée	Systèmes ERP II sera basé sur le Web, ouvert à intégrer et à interagir avec d'autres systèmes qui permettent aux utilisateurs de choisir seulement les fonctionnalités dont ils ont besoin.
données	L'information dans les systèmes ERP est produite et consommée au sein de l'entreprise	Dans un système ERP II, cette même information sera disponible dans toute la chaîne d'approvisionnement pour les participants autorisés

Cette nouvelle génération d'ERP facilite l'intégration d'applications existantes, et apporte surtout plus de réactivité à l'entreprise en lui permettant de faire évoluer ses processus.

20. Conclusion

Les travaux dans le domaine des ERP «Enterprise Resource Planning » ont offert aux sociétés une maîtrise renforcée de ses activités de production grâce à une optimisation de l'utilisation des ressources. Mais ils rencontrent un problème lié à leur implantation ce qui a causé un pourcentage significatif d'échec partiel et parfois total. Ces échecs sont provoqués par des facteurs techniques et organisationnels tel que :

- la sélection du logiciel approprié à l'entreprise.
- l'intégration d'un nouveau système avec les systèmes existants.
- Difficulté de modification du progiciel.
- La résistance au changement enregistrée lors de la mise en place de ces progiciels.
- l'adaptation et l'adéquation organisationnelle.

D'autre part, une entreprise est vue comme un ensemble de sous systèmes inter opérables et complexes qui doivent coordonner leurs taches pour satisfaire les besoins des utilisateurs en temps et en qualité de réponse.

Les limites des ERP ne font pas arrêter la migration des entreprises vers ce type de gestion, mais au contraire et au fur et à mesure des progrès technologiques on assiste à une croissance de besoin et d'utilisation de ces progiciels. Ce qui nous mène à repenser à développer de nouvelles approches d'implémentation de ces ERP qui deviennent de plus en plus indispensables.

Le but de ce travail est de résoudre le problème d'implantation des ERP la solution doit prendre en considération les facteurs déjà cités c'est le cas pour un système multi agent. En d'autres termes nous nous intéressons dans ce travail à intégrer le paradigme agent pour la modélisation d'un ERP sachant que les fonctions d'une entreprise sont complexes et nécessitent une coopération entre ses différentes unités avec une certaine autonomie des unités elle-même.

Dans le chapitre qui suit nous allons illustrer les systèmes multi agent est montrer comment ils peuvent résoudre les problèmes déjà cité.



Chapitre 2
***Les systèmes multi-
agents***

1. Introduction

L'informatique est en train de changer de manière assez profonde. Tout d'abord, l'informatique devient ubiquitaire. Au départ confinée dans les ordinateurs, elle est en train d'investir les objets de la vie courante : téléphones portables, assistants personnels, maison, etc. Elle devient ainsi de plus en plus diffusée et distribuée dans de multiples objets et fonctionnalités qui sont amenés à coopérer. La décentralisation est donc la règle et une organisation coopérative entre modules logiciels est un besoin. De plus, la taille, la complexité et l'évolutivité croissantes de ces nouvelles applications informatiques font qu'une vision centralisée, rigide et passive (contrôlée explicitement par le programmeur) atteint ses limites. On est ainsi naturellement conduit à chercher à donner plus d'autonomie et d'initiative aux différents modules logiciels.

Le concept de système multi-agents propose un cadre de réponse à ces deux enjeux complémentaires (et à première vue contradictoires) : autonomie et organisation [27].

L'approche multi-agents provient, en partie, d'une critique de l'intelligence artificielle classique qui considère qu'un problème doit être traité par un superviseur qui décompose la tâche à réaliser en tâches plus simples, puis en itérant le procédé jusqu'à atteindre un niveau de description atomique, soluble par le superviseur. A l'inverse, dans les systèmes multi-agents, la résolution de problèmes est le fruit d'interaction entre entités relativement autonomes appelées *agents*. Ces interactions sont plus ou moins complexes (coopération, conflits, compétitions...) et peuvent faire *émerger* des comportements qui n'étaient pas spécifiquement comprise dans la description exhaustive des agents [28].

Donc l'approche multi-agents est plus une branche de l' intelligence artificielle distribuée que nous allons présenter si dessous :

1.1. L'Intelligence Artificielle distribuée (IAD)

L'*intelligence artificielle* est une science venant résoudre les problèmes qui ne peuvent être résolus par l'informatique traditionnelle, elle réunit beaucoup de sciences tel que l'informatique, la psychologie, et la philosophie afin de produire des machines (ordinateurs) qui peuvent être qualifié d'intelligentes.

L'approche classique de l'I.A, qui est relative à la conception de système à base de connaissances, ne convenait pas aux applications progressives qui exigent une taille importante et une diversité des connaissances. Aussi ils existent des problèmes qui nécessitent soit :

- * **Une résolution distribuée de problèmes** : dans ce cas l'expertise est distribuée mais le domaine ne l'est pas, c'est le cas de la résolution d'une tâche complexe par un ensemble de spécialistes ayant des compétences complémentaires.
- * **Une résolution de problèmes distribués** : ici le domaine est distribué, c'est le cas de systèmes distribués physiquement.
- * **Une résolution par coordination** : plusieurs experts doivent coordonner leurs efforts donc expertises pour résoudre le problème.

De même dans la réalité, les individus travaillent généralement en groupe. Cela a conduit à l'apparition d'une nouvelle approche d'IA appelée « *intelligence artificielle distribuée* (I.A.D) » qui repose sur le principe de la distribution de l'intelligence entre un ensemble d'entités qui coopèrent pour atteindre un objectif global que chacune d'elles ne peut le réaliser individuellement.

L'I.A.D se décompose en trois différentes branches qui sont :

- RDP : Résolution Distribuée de Problème
- IAP : Intelligence Artificielle Parallèle
- Et les SMA : Systèmes Multi-Agents que nous allons étudier avec plus de détails par la suite.

1.2. Les systèmes multi-agents

1.2.1. Qu'est ce qu'un système multi-agents ?

La définition d'un système multi-agents (avec son acronyme S.M.A, et MAS pour « multi-agent system » en anglais) est : « un système multi-agents est un ensemble organisé d'agents ». Cela signifie que dans un système multi-agents, il existe une ou plusieurs organisations qui structurent les règles de cohabitation et de travail collectif entre agents (définition des différents rôles, partages de ressources, dépendances entre tâches, protocoles de coordination, et de résolution de conflits, etc.).

Dans un même système, il existe en général plusieurs organisations et un même agent peut appartenir à plusieurs simultanément. Des exemples d'organisations d'agents dans le monde réel sont une organisation économique telle qu'une entreprise, mais aussi une organisation animale telle qu'une fourmilière. Suivant les cas, les comportements des agents sont plus ou moins complexes et rationnels et l'organisation est plus ou moins adaptative. L'élément central aux systèmes multi-agents est l'équilibre (et la complémentarité) entre autonomie et organisation.

Les agents sont en général situés dans un environnement contenant également des entités passives, manipulées par les agents (Par exemple des ressources, des données, des objets physiques, etc) et communément appelées objets. Chaque agent n'a qu'une connaissance partielle de son environnement et des autres agents. Un système multi-agents est donc intrinsèquement décentralisé [27].

Et d'après Jacques Ferber : Un S.M.A est défini comme:

- ▶ Un ensemble B d'entités plongées dans un environnement E (E est caractérisé par l'ensemble des états de l'environnement S).
- ▶ Un ensemble A d'agents avec $A \subseteq B$.
- ▶ Un système d'actions (opérations) permettant à des agents d'agir dans E (une opération est une fonction de $S \Rightarrow S$).
- ▶ Un système de communication entre agents (envoi de messages, diffusion de signaux,...)
- ▶ Une organisation O structurant l'ensemble des agents et définissant les fonctions remplies par les agents (notion de rôle et éventuellement de groupes)
- ▶ Eventuellement, une relation à des utilisateurs U qui agissent dans ce SMA via des agents interfaces U est inclus ou égale à A [29].

1.2.2. Différence entre S.M.A et Système orienté objets

IL a eu toujours une confusion entre le concept d'agent et celui d'objet voir un système multi agent et un système orienté objet, certainement on ne parle pas de la même chose mais de deux système qui se ressemblent un peut mais les différences qui existes entre eux sont crucial, la ressemblance entre ces deux types de système réside dans :

- Tout deux l'agents et l'objets encapsulent leur état interne.
- Ils peuvent également poser des actions sur cet état grâce à leurs méthodes

- Ils communiquent en s'envoyant des requêtes par des messages par exemple.
- Ils permettent de distribuer la traçabilité.
- Ils fournissent des mécanismes d'interaction entre les composants du système. [30]

Tandis que les différences majeures entre un système multi agent et un système orienté objet sont les suivantes : [31]

- **Le degré d'autonomie** : un objet peut invoquer une méthode qui doit être exécutée et qui appartient à un autre objet pour pouvoir accomplir son travail donc il dépend des autres objets du système. Par contre un agent, recevra une requête et décidera de son propre gré s'il doit exécuter ou non une action.
- **La flexibilité** : à l'inverse de l'objet un agent possède un comportement flexible.
- **La source de contrôle** : un système orienté objet, contient une seule source de contrôle, pendant que chaque agent dans un système multi agent est considéré comme une source de contrôle.

1.2.3. Quand utiliser un S.M.A?

L'approche SMA est recommandée quand:

- ▶ le problème est trop complexe mais peut être décomposé.
- ▶ il n'y a pas de solution générale ou lorsque celle-ci est trop coûteuse en CPU.
- ▶ on peut paralléliser le problème (gain de temps).
- ▶ on veut une certaine robustesse (redondance).
- ▶ l'expertise provient de différentes sources.
- ▶ les données, contrôles, ressources sont distribués.
- ▶ le système doit être adaptatif.
- ▶ A des fins de modélisation (populations, structures moléculaires, tas de sable...).

1.3. Concept d'agent

1.3.1. Qu'est ce qu'un agent?

Il n'existe pas encore un consensus sur la définition d'un agent. En plus de la relative jeunesse du domaine, une des raisons est que diverses communautés revendiquent ce terme avec des problématiques parfois au départ assez différentes (par exemple en ce qui concerne les agents mobiles) même si ces différentes problématiques sont complémentaires et conduites à se rencontrer à terme.

Nous proposons une première définition qui est la suivante : « un agent est une entité logicielle ou physique à qui est attribuée à une certaine mission qu'elle est capable d'accomplir de manière autonome et en coopération avec d'autres agents » [27].

Un agent est un système :

- ▶ 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 forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser)
- ▶ qui possède des ressources propres.
- ▶ qui est capable de percevoir (mais de manière limitée) son environnement.
- ▶ qui ne dispose que d'une représentation partielle de cet environnement (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 reçoit [28].

2. L'ERP et les systèmes multi agent

Dans un contexte international de plus en plus compétitif, les entreprises ont constamment besoin d'adapter et d'optimiser leurs outils industriels en vue d'augmenter leur productivité. En particulier, le pilotage et l'ordonnancement des lignes de production sont évalués en fonction de paramètres de coût et de délais, qui contribuent à définir le rendement et donc la compétitivité d'une entreprise, c'est pour cela que les systèmes ERP ont paru, mais il reste encore le besoin accru de flexibilité, d'agilité et d'efficacité des systèmes de production qui se traduit par une complexité grandissante qu'il faut savoir maîtriser. Les systèmes multi agents de leur côté permettent d'acquérir les caractéristiques voulues et en même temps résoudre le problème de complexité, grâce aux propriétés suivantes :

a) La modularité : un système ERP est composé de différents modules partageant une base de données unique, d'un autre côté un système multi agent est destiné pour l'implémentation de solution modulaire, ce qui fait de lui un bon outil pour les ERP.

b) L'autonomie : est la propriété principale d'un agent logiciel, donc l'utilisation d'un système multi agent dans un ERP lui garantit cette caractéristique qui devient de plus en plus nécessaire, pour l'optimisation des applications actuelles qui sont destinées à résoudre des problèmes nécessitant une solution distribuée, ces applications sont composées d'unités capable d'agir sans intervention externe dans le but de manier à la complexité due à ce type de problème.

c) L'adaptabilité : un système multi agent est adaptable ce qui veut dire qu'il a une capacité de répondre aux changements de l'environnement, nous parlons de l'auto-organisation des agents. Cette caractéristique permet à l'entreprise de s'adapter aux changements du marché.

d) L'intégration d'expertise: dans le cas de l'ERP il a besoin de connaissances de domaines permettant la gestion de l'entreprise, les systèmes multi agent lui facilitent la tâche.

Dans le but de profiter des bénéfices que peut apporter les caractéristiques déjà citées, les systèmes ERP se dirigent vers une approche multi agent. Dans la partie qui suit nous allons présenter quelques travaux dans ce sens.

2.1. L'approche MAERP (Multi Agent based ERP system)

Avec la mise en œuvre de l'ERP, une entreprise aura besoin de remplacer les systèmes actuels, avec de nouveaux modules ERP et donc transférer ses données de l'ancien système d'information vers le nouveau logiciel ERP. Par conséquent plusieurs difficultés techniques se posent due par exemple à la différence des architectures de données entre l'ancien et le nouveau système, donc le transfert des anciennes données pourrait entraîner de graves pertes de données, c'est la problématique principal qui motive l'approche MAERP. [35]

L'approche MAERP présente un prototype d'un système ERP basé sur les systèmes multi agent, cette architecture utilise des agents logiciels intelligents comme des représentants de la gestion des systèmes d'information, ces agents peuvent communiquer par le biais du réseau de l'entreprise tout en servant d'interprète de la base de données du système actuel de l'entreprise, donc l'ancien système seraient conservées et la communication et le partage d'informations sont disponibles par les interactions entre les différents agents intelligents.

L'avantage de cette approche réside dans deux points essentiels :

- 1) Elle ne nécessite pas une re-conception de l'architecture de données, en plus le système utilise l'expertise existante au sein de l'entreprise.
- 2) Diminue le coût de la mise en œuvre du logiciel ERP, grâce à ces agents qui fournissent une alternative à faible coût pour atteindre le même objectif d'intégration entre les sous-systèmes d'un système ERP classique.

2.1.1. L'architecture du MAERP

L'architecture du système MAERP, contient quatre types d'agents: les agents de coordination, ceux d'interface, les agents de collecte de données, et les agents de tâche.

a) L'agent de coordination : c'est l'agent principal du système MAERP, chaque service de l'entreprise possède un ou plusieurs agents de coordination en fonction de la complexité de sa tâche. Un agent de ce type représente un service de l'entreprise ou le système d'information lors de la communication avec les autres agents de coordination, et contrôle le reste des agents du sous-système réservé au service dont il appartient.

Grâce à leurs connaissances du domaine, les agents de coordination ont la capacité de suivre, de communiquer et de collaborer avec d'autres agents, et de réagir aux demandes ainsi qu'assigner des tâches aux agents de tâche et aux agents de collecte de données. Les tâches effectuées par un agent de coordination peuvent être résumées dans les points suivants :

- La réception de l'instruction et l'établissement des rapports pour les utilisateurs via une interface agent
- L'attribution de la collecte de données et la réception de données provenant d'un agent de collecte de données
- Relais de données.
- L'attribution des tâches et de recevoir les résultats et les réactions de la tâche des agents.
- La communication avec d'autres agents de coordination [36].

b) L'agent de collecte de données : le rôle d'un agent de collecte de données est d'effectuer le stockage des données et fournir les données répondants aux demandes des agents de coordination, en recherchant dans la base de données. L'intelligence des agents de collecte de données réside dans l'identification des données non valides et des valeurs manquantes afin que les données soient complètes au moment de leur envoi à l'agent de coordination.

c) L'agent de tâche : la fonction d'un agent de tâche varie d'un service à un autre en fonction de ce que ce dernier doit accomplir, aussi le nombre d'agents de tâche varie selon le nombre et la complexité des tâches. En général, cet agent reçoit des données de l'agent de coordination, effectue l'analyse de ces données, et exécute les programmes spécifiques, et puis retourner les résultats à l'agent de coordination.

d) L'agent d'interface : l'agent interface garantit la communication entre les utilisateurs et l'agent de coordination et leur prépare des rapports, aussi il interprète les résultats. Il possède la capacité d'apprendre et de stocker des utilisateurs et leurs préférences, ainsi que la capacité de surveiller et d'informer les utilisateurs lorsque les tâches sont accomplies [35].

La figure suivante illustre l'architecture du système MAERP : [36]

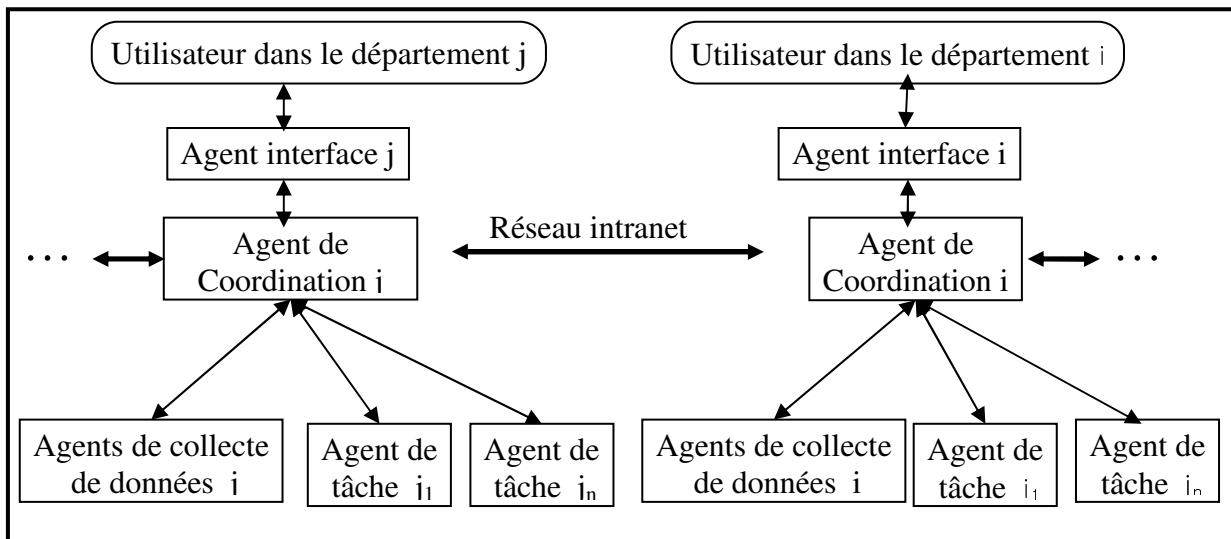


Figure 1: Architecture du système MAERP

D'après le schéma, il est clair que l'architecture du MAERP est composée d'agents de coordination desquels sont des représentants du système d'information et qui communiquent les uns avec les autres sur le réseau de l'entreprise, chacun étant connecté à un agent interface qui agit comme un outil de communication entre l'utilisateur et le système MAERP, en plus plusieurs groupes d'agents de collecte de données et qui effectuent des tâches spécifiques.

2.2. Le projet PABADIS (Plant Automation Based on Distributed Systems)

C'est un projet européen qui a pour objectif l'automatisation de la planification de production en utilisant les systèmes distribués tout en faisant appel à la technologie d'agents logiciels et la technologie Java. Ce projet a débuté en décembre 2000 avec la contribution de plusieurs organismes dont :

1) des organismes académiques : tel que

- L'IAF Fraunhofer Institut, de Magdebourg, en Allemagne
- La PUM Université de Marburg, en Allemagne
- L'ICT Université de Vienne en Autriche
- L'EMA-LGI2P, Nîmes, en France

2) des organismes Industriels: comme

- Le Phoenix Contact, le Jetter, et aussi l'IMS qui sont des organismes allemands
- UNISOFT, et Hatzopoulos qui sont Grèques
- Et enfin le P2I Français

3) et des Groupes IMS (Intelligent Manufacturing Systems) : ces groupes ont pour objectif la construction des systèmes industriels intelligents, nous pouvons citer :

- Les groupes Sun Microsystems, et Ajile des Etats-Unis
- Centor, Prevost Motor Coach de Canada
- ABB, CUI Genève PebbleAge, de Suisse

[37]

Le projet vise principalement à répondre aux besoins d'une production variable, et des demandes incertaines des clients, mais aussi à cerner le processus de fabrication qui doit être flexible, et garantir la distribution des ressources dans les réseaux de machines, d'ateliers, ou d'entreprises. Ces besoins rendent le contexte de développement du projet complexe et nécessitant une prise en charge des aspects dynamiques et c'est le cas pour tout ERP.

Il vise aussi à créer un environnement plug-and-participate qui permet aux sociétés de production de faire des contrôles plus souple sur le travail, et facilite l'intégration de nouveaux composants au réseau ce qui va permettre de brancher une nouvelle machine et l'utiliser sans des changements majeurs au sein de l'héritage.

Cela dans le but d'apporter les bénéfices suivants :

- Une réduction des opérations d'ordonnancement, et par conséquent une réduction de la complexité.
- Un certain degré de tolérance aux pannes
- La reconfiguration automatique
- La résolution de problèmes du parallélisme
- L'auto-organisation des systèmes

2.2.1. Le modèle PABADIS

Le modèle PABADIS est basé sur plusieurs composants qui sont :

- Un service d'actions et de contrôle appelé aussi agence.
- La CMU (Co-operative Manufacturing Unit) : c'est l'union d'une unité manufacturière et un agent résidentiel, cet agent collecte les informations du réseaux via une interface et les fournit à l'unité manufacturière associée. il existe plusieurs types de CMU qui sont :
 - CMU logique : qui fournit les services informatiques, comme les algorithmes complexes d'échéancier, les recherche d'informations dans les bases de données...etc, une CMU de ce type peut être des réseaux informatique, Internet, des bases de données
 - CMU manufacturière : utilisée pour les scénarios physique de production, elle peut être des machines individuelles, la planification des ventes, ou bien l'ensembles des instructions complète pour la production d'un produit.
 - CMU hybride : c'est une combinaison des deux types précédents, elle est utilisée dans le cas des machines complexes où l'on peut difficilement distinguer les fonctionnalités logique des fonctionnalités physiques.
 - CMU SCADA (Supervisory Control and Data Acquisition) : cette CMU sert à fournir au système PABADIS une interface qui le lie aux fonctionnalités SCADA qui permettent un contrôle supervisé de la planification ainsi que l'acquisition des données.
- Plus un ensemble d'agents dont : un agent produit, un agent résidentiel, et un agent de gestion.
 - L'agent Produit : accompagnent le produit donc il est doté d'une certaine mobilité, et recherchent et utilisent des services. Cet agent représente la requête venue du système ERP centralisé, il est composé de :
 - a- un ensemble de méthodes comme ceux responsables des taches de calcul, ou de sécurité, ou encore des mécanismes de communication.

b- et aussi des données du produit tel que les demandes, les plans d'exécution ... etc.

Grâce à ses composants cet agent garantit l'exécution correcte des tâches et un control réussi.

- L'agent Résidentiel : représente les ressources (CMU) et offre des services, en fournissant la connexion entre les composants du réseau dont les autres agents et les ressources de la CMU.
- L'agent de gestion de produit : fournit une interface facilitant le contrôle humain qui est nécessaire au niveau de certaines fonctions, et permet aussi l'observation du système au cours d'exécution. Par conséquent, l'agent de gestion de produit est responsables de la collecte d'informations dans le système et les servir à l'ERP, MES ou SCADA selon leurs besoins, c'est pour cela qu'il est appelé aussi agent d'information.

Tous ces composants travaillent ensemble et collabores pour fournir une architecture qui garantie la distribution du contrôle manufacturier dans une entreprise, ce qui est illustré par la suite :

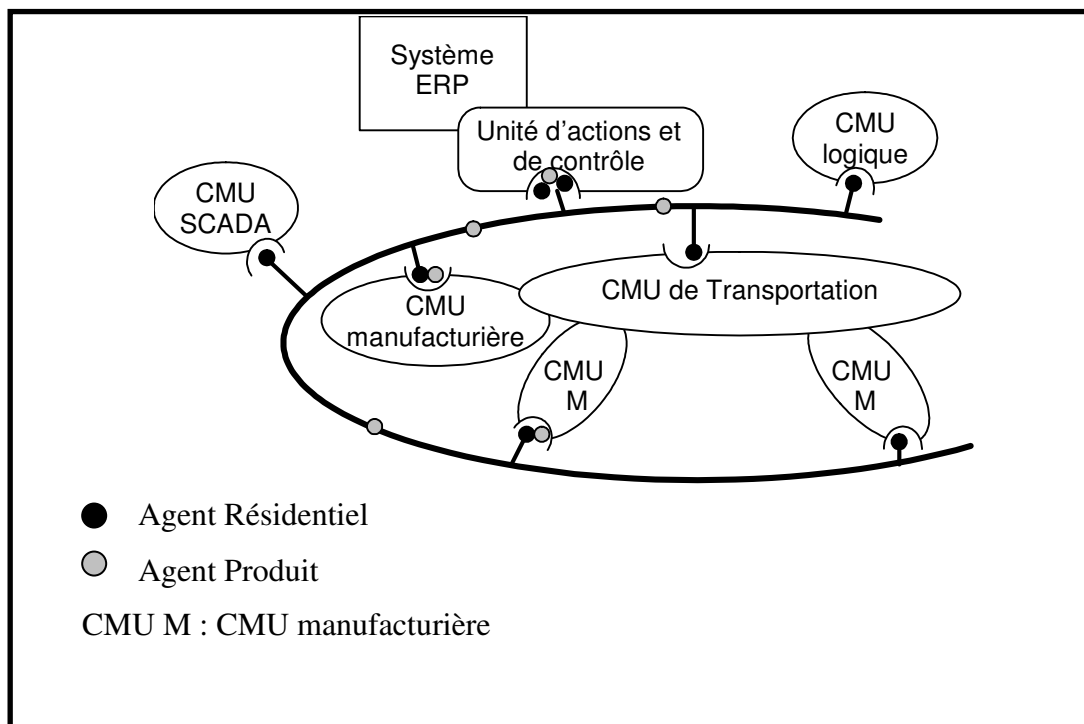


Figure 2 : Schéma représentatif du modèle PABADIS

Ce modèle se base sur deux points essentiels :

- a- La décentralisation des fonctions MES (Manufacturing Execution System) et SCADA qui sont prises en charge par des agents, cette caractéristique permet une réduction de la complexité de l'allocation des ressources ce qui assure la *flexibilité verticale*.
- b- La mise en réseau de toutes les ressources, ce qui est garantie par l'utilisation de la technologie plug-and-participate pour doter le système d'une *flexibilité horizontale*.

2.2.2. Fonctionnement du système PABADIS

Comme mentionné auparavant, les CMUs dans PABADIS sont composés d'un module de fabrication et un module d'intelligence. Ce dernier est en fait l'agent résidentiel qui représente la CMU dans le réseau ; Il fournit au réseau une interface pour accéder aux ressources physiques et aux fonctionnalités qu'offre la CMU. L'interface est utilisée par les agents produits, qui sont des agents mobiles, ces agents là guident le produit dans son processus de fabrication, pour cela ils possèdent toute l'information nécessaire pour l'optimisation du développement du produit. Par conséquent, quand un agent produit arrive à une CMU, il fournit à l'agent résidentiel les paramètres exacts et nécessaires pour effectuer les traitements dont il est chargé.

Les agents produits sont créés dans une entité appelée Agence, qui est l'interface entre la communauté des agents PABADIS et le système ERP. Elle traite la demande et la transforme en documents XML. L'Agence analyse les documents et crée un agent produit qui sera responsable de la création de la pièce correspondante, elle recueille également les rapports des agents produits.

Sans oublier le rôle important que joue le Look-up Service (LUS), qui est l'un des éléments clés du système. Le LUS comme son nom l'indique est un service de recherche qui agit selon un mécanisme semblable à celui adopté par les pages blanches qui permettent d'accéder au numéro de téléphone d'une personne si l'on connaît son nom, même principe pour le LUS qui fournit un accès à des informations concernant les agents produits, afin qu'ils sachent à quel destination doivent-ils migrer ou quand ont-ils besoin d'une fonction précise par exemple. Chaque CMU informe le LUS de son adhésion au réseau, et lui fournit des données sur les fonctionnalités qu'elle peut offrir, pour cela Le LUS doit mettre à jour ses données périodiquement.

Le processus de fonctionnement du système PABADIS est le suivant :

- Le système ERP formule sa commande industrielle et l'envoie accompagnée par une description des tâches nécessaire à la préparation du produit.
- L'agence, qui est appelée aussi unité d'action et de contrôle, reçoit la commande et la transforme en une suite de requêtes que les agents peuvent comprendre.
- L'agent fabricant de l'agence fabrique un produit pour chaque requête
- L'agent produit communique avec l'agent résidentiel adéquat afin de définir la démarche que doit suivre le produit en question, ensuite il exécute la requête tout en se mobilisant pour garantir le contrôle.
- Par la suite, l'agent produit effectue un rapport rendu à l'agence, ce rapport précisera si l'exécution de la requête est accomplie ou bien qu'elle ait échouée.
- L'agence va mettre fin à l'agent produit, ou bien il va se terminer de lui-même.
- A la fin l'agence envoie un rapport détaillé à l'ERP où le cycle de production se déroulera.

2.2.3. La communication à l'intérieur du système PABADIS

L'approche PABADIS fournit un système multi agent orienté produit, ce qui fait de l'agent produit l'élément le plus actif et sur le quel se focalise la communication à l'intérieur du système PABADIS, il existe plusieurs types de communications dont l'agent produit joue un rôle, les plus importantes sont :

a) communication entre agents produit

En cas de conflit sur l'utilisation des fonctions fournies par une CMU par plusieurs agents produit, ces derniers se mettent à négocier afin de décider les priorités d'utilisation des fonctions.

b) communication entre l'agent produit et l'agent résidentiel

Ce type de communication ne nécessite ni négociation ni une re-planification il utilise plutôt un simple mécanisme, qui est indispensable pour l'allocation des ressources, la planification, mais aussi pour d'autres fonctions du MES.

c) communication entre l'agent produit et le LUS

Cette communication est établie dans le but d'acquérir les informations nécessaires à la migration des agents produit, pour cela chaque agent produit envoie une requête au service LUS, de son côté ce dernier fait sa propre recherche et rend la réponse qui définit toute CMU permettant de fournir le service dont l'agent produit a besoin.

d) communication entre l'agent produit et l'agent de gestion de produit

Dans ce cas, c'est l'agent de gestion de produit qui prend toujours l'initiative vu qu'il est responsable de la collecte d'informations, et bien sûr l'agent produit ne fait que répondre à la requête. Cette communication est invoquée généralement par les fonctions MES nécessitant une distribution.

La figure qui suit est une illustration de la communication dans le système PABADIS.

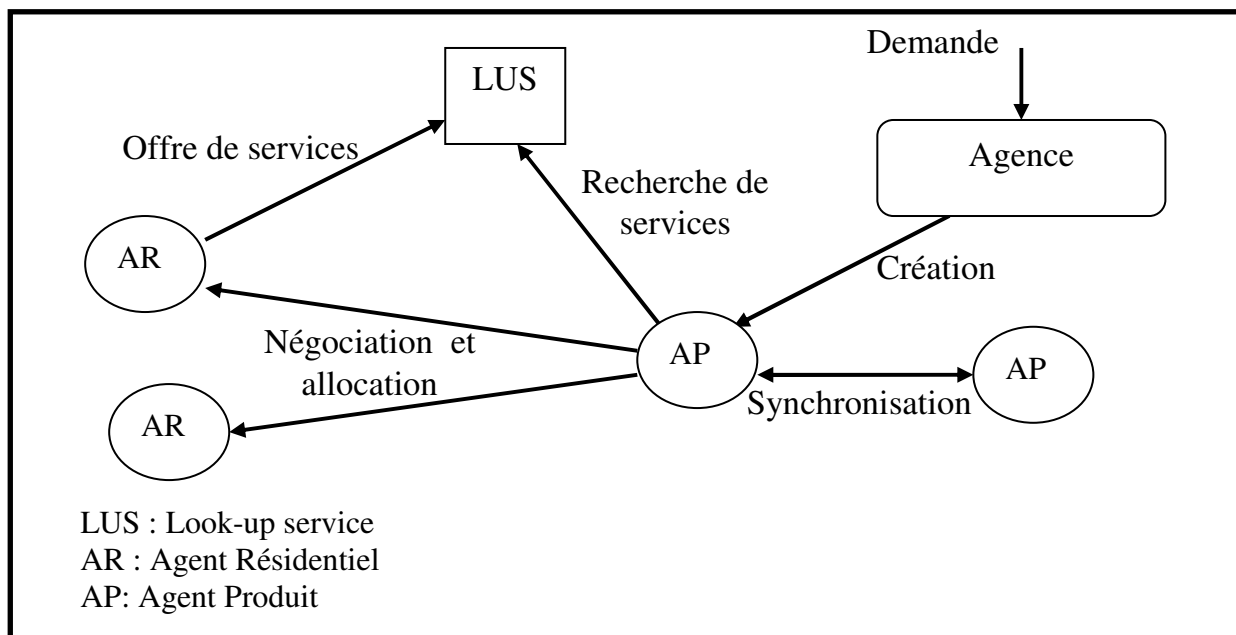


Fig 3 : Schéma représentatif de la communication PABADIS

Comme nous l'avons déjà cité, tout agent résidentiel communique soit avec un agent produit ou bien le LUS pour qu'il puisse compléter les tâches dont il a la charge, or les agents résidentiels ne communiquent pas entre eux dans le but d'assurer leur indépendance et c'est la caractéristique majeur du système PABADIS, du fait que la production se focalise sur le produit, qui utilise les ressources disponibles, et non pas sur les machines, qui ne font qu'exécuter les tâches demandés par les agents produit.

2.3. Le projet ExPlanTech

Le projet ExPlanTech est un projet de l' IST (Information Society Technologies) qui est une organisation européenne destinée à la recherche dans le but d'adapter et d'intégrer les technologies prometteuses dans les applications industrielles [38]. Ce projet a débuté le premier octobre de l'an 2000 avec la coopération des organismes suivants :

- L'université technique tchèque en PRAGUE dans la république tchèque
- MODELARNA LIAZ SPOL. S R.O. dans la république tchèque
- CERTICON A.S. dans la république tchèque
- HATZOPOULOS S.A. en GRÈCE

L'objectif de l'ExPlanTech est d'introduire et d'adapter une technologie multi-agent pour la planification de la production dans deux cas industriels en Grèce et en République tchèque il s'agit de la technologie *Proplant*. Dans d'autres termes, il vise à intégrer le système Proplant dans l'actuel Enterprise Resource Planning (ERP) et atteindre une amélioration de la performance de cette dernière.

Cette approche apporte plusieurs avantages dont :

- l'amélioration de l'allocation des ressources.
- la flexibilité du système.
- l'amélioration de la relation fournisseur \ consommateur [39].

Pour atteindre son objectif le projet doit comporter les deux étapes qui suivent:

- a) L'adaptation et la personnalisation de la plate-forme technologique Proplant, y compris les interfaces de communication, selon les besoins des utilisateurs.
- b) L'installation du prototype multi-agent personnalisé de la planification de production dans les systèmes ERP existants.

2.4. L'architecture du système Proplant

La technologie Proplant aborde le problème des projets de production, en mettant l'accent sur les besoins spécifiques du client, la nécessité d'une aide à la décision qui soit intelligente dans le but d'améliorer la relation fournisseur\consommateur, l'environnement de travail et la confiance des clients [39].

Dans un système Proplant l'activité traditionnelle de planification de la production utilisée dans la majorité des entreprises industrielles, est remplacé par un agent de service centré sur la négociation et une décision fondé sur des connaissances basées sur les exigences des tâches, ce qui donne un aspect intelligent. Cette nouvelle technologie augmente le taux d'affaires et la compétitivité des PME par l'optimisation de leurs capacités opérationnelles.

Le système ExPlanTech contient deux catégories d'agents: des agents intra-entreprises (IAE), et des agents inter-entreprises (IEE) dont nous allons décrire par la suite.

Commençant par la catégorie des agents IAE, et qui est propre au système Proplant, où nous pouvons distinguer les agents suivants :

- 1) ***L'agent de production*** : représente l'unité de production du niveau le plus bas du système, il effectue les échéanciers des machines parallèles et la distribution des tâches, aussi il gère l'allocation des ressources par l'intermédiaire d'un type particulier d'agents de base de données
- 2) ***L'agent de planification de production*** : qui est chargé de la planification du projet. Son objectif est de construire un plan exhaustif, partiellement classé de l'ensemble de tâches qui doivent être menées en vue d'accomplir le projet.
- 3) ***L'agent de gestion de la production*** : il contrôle le travail d'un groupe d'agents produits pour une tâche considérée, ou bien il partage les tâches dont il a la charge avec un autre agent de son type.
il a pour objectif d'augmenter le degré de performance de la gestion projet, donc son rôle est de choisir parmi les agents produits le meilleur en termes de coûts opérationnels, du délai de livraison et de sa disponibilité.
- 4) ***L'agent client*** : c'est une autre instance d'un agent de la catégorie IEE qui fournit une interface entre les clients et le système.
- 5) ***Le méta agent*** : est un agent de surveillance qui permet de visualiser des informations, des documents ou encore les flux de travail du système, en plus il fournit

des conseils sur l'efficacité optimale du système. Un tel agent permet l'aide à la décision distribuée en ce qui concerne la planification des processus de fabrication [40].

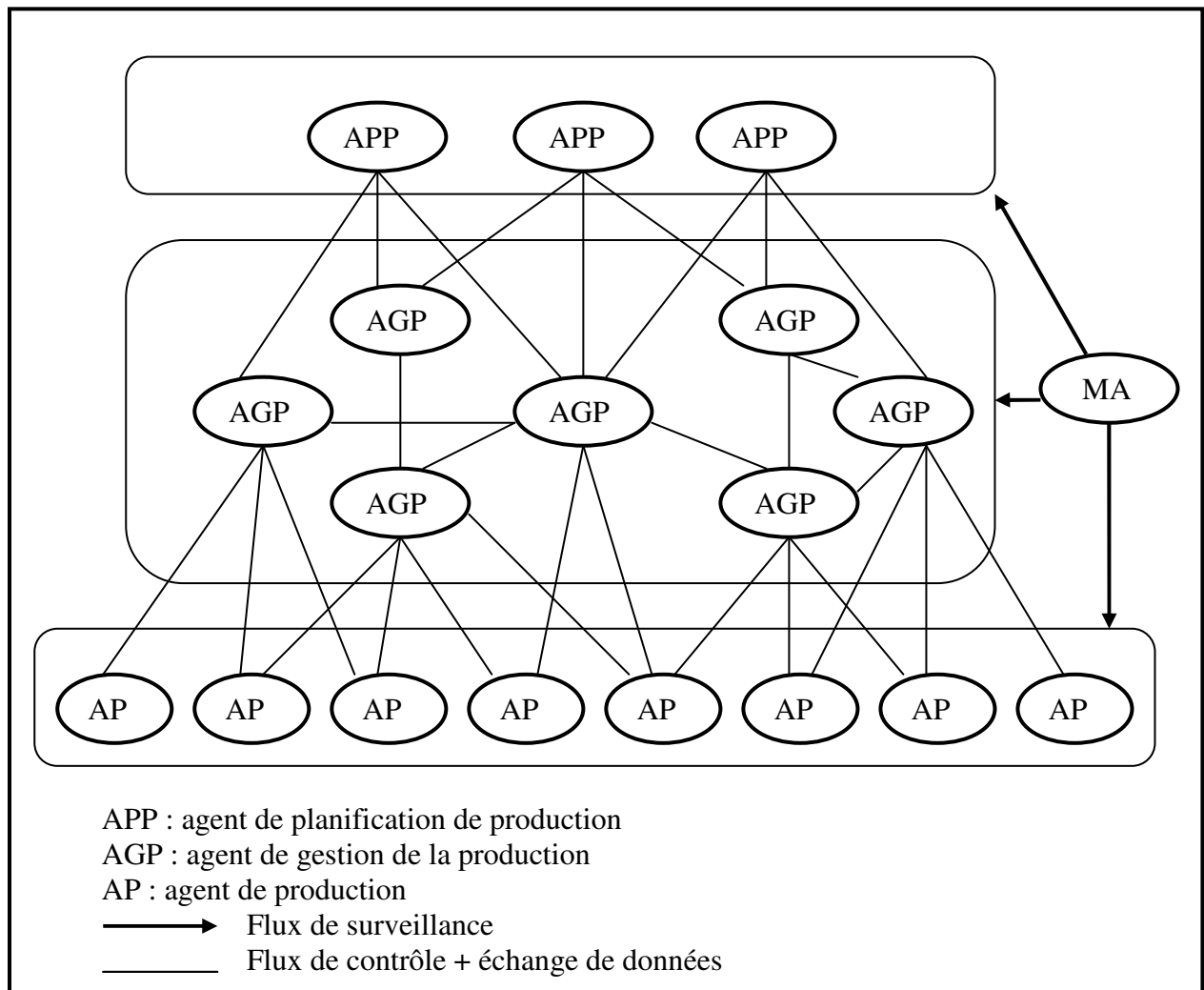


Figure 4: Architecture générale du Proplant

Comme nous pouvons le constater depuis le schéma si dessus, les agents du système Proplant sont organisés d'une manière hiérarchique, les agent du niveau le plus haut contrôlent et gèrent ceux du niveau qui vient juste au dessous et ainsi de suite, ce contrôle sera accompagné de données nécessaires au charges à accomplir, de leur cotés les agent contrôlés envoient des données nécessaires au déroulement des tâches au agents qui leurs contrôlent ou ceux qui ont besoins d'informations. De plus les agents du même niveau peuvent communiquer soit pour échange de données, soit pour un contrôle dans le cas où un agent veut déléguer un autre pour une tâche.

Donc seuls les agents de catégorie IAE ont été mis en œuvre dans le système multi agent Proplant, l'autre catégorie qui est celle des agents IEE a fait l'objet du projet ExPlanTech.

2.4.1. Le système ExPlanTech

Le système ExPlanTech repose sur l'intégration de l'architecture du Proplant expliquée dans la section précédente, mais il contient aussi de nouveaux agents pour des fonctions spécifiques ce type d'agents est les agents inter-entreprises (IEE) parmi les quels nous citons :

- 1) *L'agent Extra-Enterprise* : permet à un utilisateur distant -qui se trouve à l'extérieur de l'entreprise- d'accéder au système, bien sûr seulement dans le cas où cet utilisateur possède le droit d'y accéder, ce service est réalisé à l'aide d'une technologie de client léger, qui est flexible et indépendante de la plateforme, chose nécessaire pour répondre à un environnement variable tel qu'il est celui en dehors de l'entreprise. Grâce à cet agent le système ExPlanTech est accessible via le navigateur Web tout en maintenant la sécurité d'accès.
- 2) *L'agent entreprise vers entreprise* : cet agent rend le système ExPlanTech accessible par un autre système qui appartient à une entreprise différente, ou encore avec des agents du même type qui appartiennent à d'autres entreprises, cette accès permet la communication entre deux entreprises et donc l'échange d'informations qui peuvent être utilisées dans la décision par exemple, ou encore pour l'externalisation des tâches dans le cas d'épuisement de ressource.
- 3) *L'agent cockpit* : Cet agent est destiné à la gestion des conflits dans le cas où plusieurs utilisateurs ont besoin d'interagir avec l'agent de planification en même temps, il permet de présenter l'utilisateur de manière conviviale avec l'état des processus de production, les plans, les charges de ressources acquises etc, cette présentation lui facilite la tâche de trouver des solutions aux conflits, il fournit également une possibilité d'interagir avec le système, ou encore de changer les paramètres des plans ou des ressources en fonction des droits d'accès.
- 4) *L'agent de simulation* : représente une interface entre le système de planification et une communauté d'agents d'émulation qui simulent le comportement des machines physiques ou des ressources humaines dans le but de connaître les délais exacts de

fabrication des produits ou le taux d'échec et donc permettre un feedback et alors une correction des plans de production par l'agent de simulation bien sûr [41].

Ces agents représentent l'extension du système ExPlanTech intra-entreprise, ils fournissent différentes fonctions qui permettent l'ouverture de l'entreprise vers son environnement extérieur et donc le prendre en compte dans sa planification, pour aboutir à des résultats meilleurs.

2.5. Synthèse

Les projets que nous avons présentés, sont une intégration des systèmes multi agents dans les ERP dans le but d'apporter à ces systèmes de la flexibilité et pour les rendre plus performants vu qu'ils sont des systèmes complexes, mais chacun d'eux à utiliser une manière différente.

Le projet PABADIS a adopté une approche qui lui a permis de passer de la solution fonctionnelle vers l'intégration des systèmes multi agent ce qui est expliqué dans la figure qui suit :

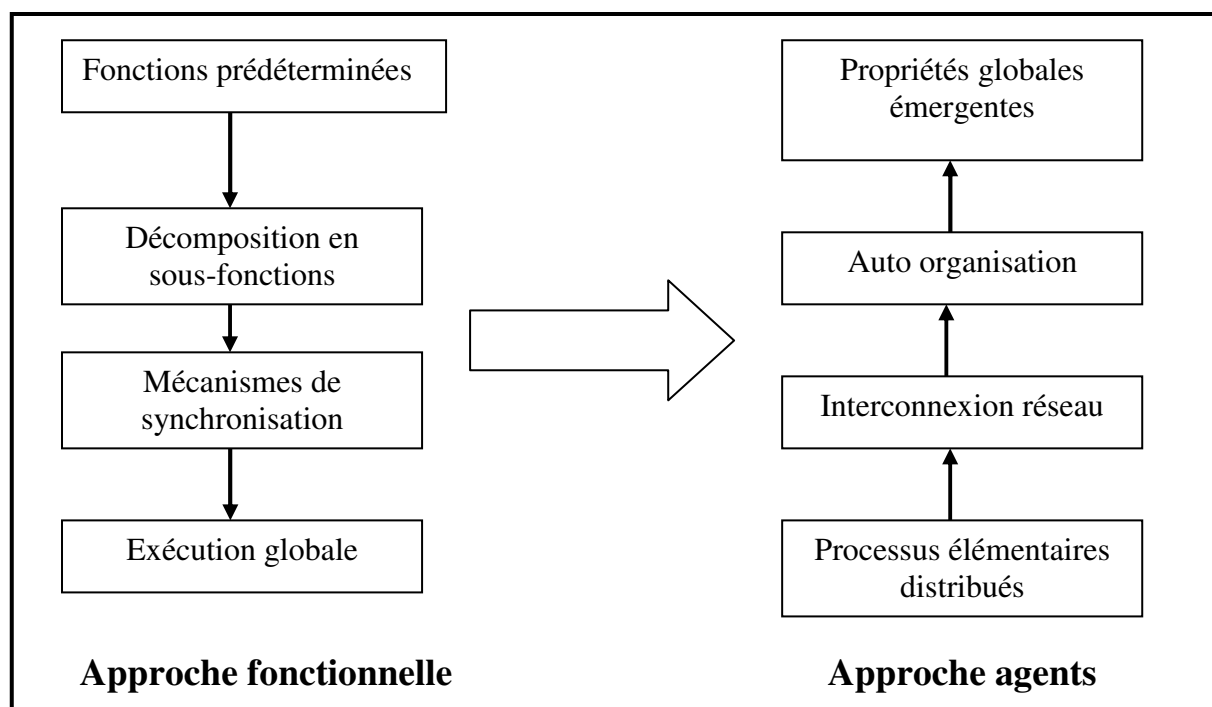


Figure 5: Schéma expliquant le passage d'une approche fonctionnelle vers une approche multi agent [37]

PABADIS aborde le problème de la flexibilité et la reconfiguration de systèmes de fabrication. L'approche suivie est la répartition de certaines fonctions du système d'exécution Manufacturier dans un environnement multi agent pour augmenter la flexibilité de la planification de la production. Pour atteindre ces objectifs, le projet PABADIS utilise :

- des agents mobiles pour décrire les différentes pièces, où l'agent oriente le produit associé à travers le processus de fabrication.
- des agents stationnaires représentant les installations de fabrication de l'entreprise nous parlons des machines, systèmes de transport, et bases de données.

Donc les agents du PABADIS ne sont implémenter qu'au niveau du MES qui sert de support à l'ERP dans le domaine de gestion de ressources et qui est biensure un outil propre au système PABADIS, la figure qui suit illustre ce que nous avons remarqué:

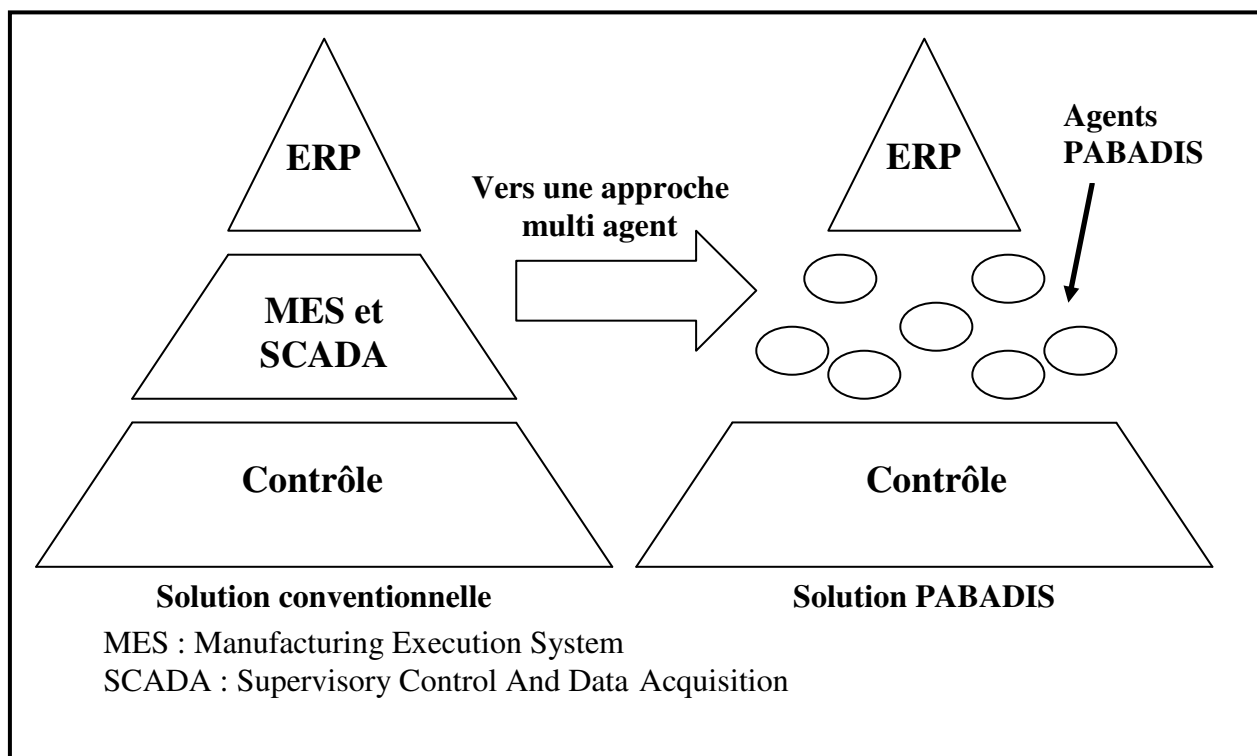


Figure 6: Schéma représentatif de l'approche d'intégration d'agents chez le PABADIS

Pendant que le projet ExPlanTech a renforcé le système ERP avec des agents qui répondent à des besoins supplémentaires spécifiques, car il intègre le système Proplant avec ses agents, qui sont destinés à la planification, dans un système ERP déjà existant, cela dans le but de profiter des bénéfices en terme de performance et de la faciliter de mise à niveau qualitative de la méthode de planification de la production. Alors les agents du système ExPlanTech ne sont pas intégrés à l'intérieur de l'ERP mais par contre il lui sert de support ce qui est expliqué dans le schéma qui suit : [40]

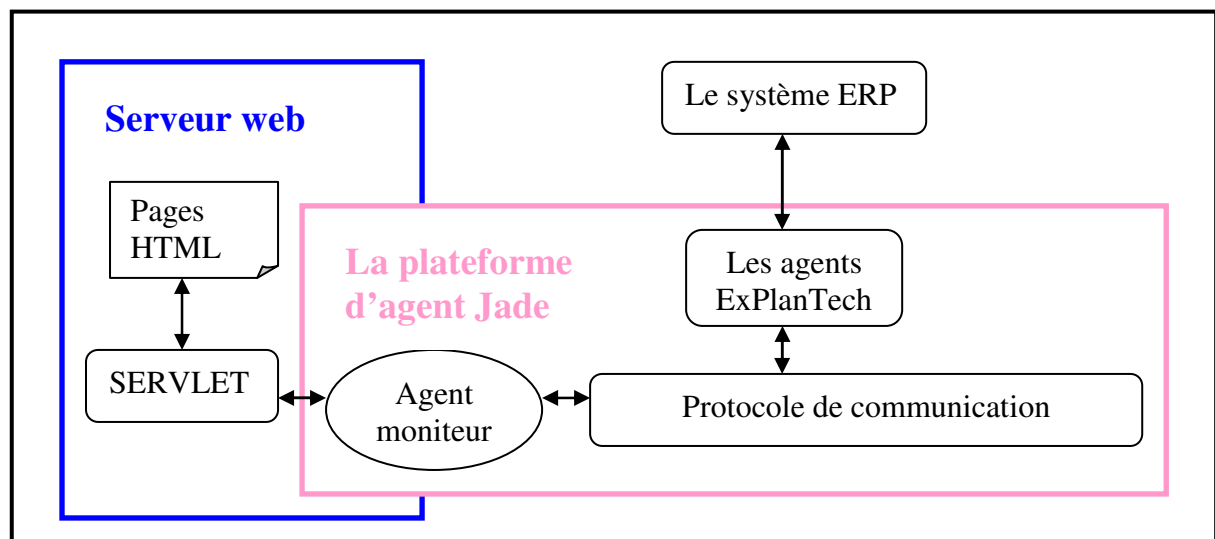


Figure 7: schéma général de l'implantation du ExPlanTech

NB : Le SERVLET, est un composants Java qui établit des communications.

De sont coté l'approche MAERP, conserve l'ancien système d'information en plus du nouveaux système ERP, tout en ajoutant quatre agents qui ont pour rôle le transfert des données entre les deux systèmes dans le but d'éviter la perte d'information sans avoir à changer la conception de l'architecture de données utilisée.

Donc les projet PABADIS, ExPlanTech, et l'MAERP ainsi que les projets que nous avons rencontrés lors de notre recherche et qui sont des projets destinés à l'intégration de l'approche agents dans les systèmes ERP repose sur l'ajout d'agents pour des taches précise afin d'améliorer le rendement du système, et non pas sur la distributions des services du système ERP lui-même et c'est l'objet de notre recherche.

Vu que les systèmes ERP sont des systèmes complexes composés de plusieurs services partageant une base de données centralisée il est nécessaire de leur garantir une certaine indépendance et donc une autonomie, mais une entreprise est vue comme un ensemble de sous systèmes inter opérables et complexes qui doivent coordonner leurs tâches pour satisfaire les besoins des utilisateurs en temps et en qualité de réponse, c'est pour cela que nous devons garantir la communication entre les services de notre entreprise, ce qui nous mène à être sûre que la solution multi agents est meilleur pour l'implantation des services de l'entreprise.

Rappelons qu'un système ERP est composé d'un ensemble de modules tel que le module de gestion commerciale, celui de gestion comptable et financière, de ressources humaines, ou de gestion client...etc. Dans le modèle que nous proposons nous allons remplacer chaque module du système ERP par un agent, aussi l'approche que nous proposons est une approche plutôt cognitive alors nos agents seront des agents cognitifs, pour plus de précision nos agents seront des systèmes experts, ils possèdent un savoir et peuvent raisonner. Aussi parce que un système expert regroupe les connaissances du domaine et raisonne afin d'aboutir à des résultats plus fiables, et puisque les systèmes experts sont des programmes conçus pour raisonner habilement à propos des tâches dont on pense qu'elles requièrent une expertise humaine considérable, et c'est le cas des modules de l'ERP chacun représente tout un domaine d'expertise, nous pensons que c'est un bon choix.

3. Conclusion

Quoique le concept de système multi agent soit un paradigme de génie logiciel qui est récemment apparu, il existe pas mal d'applications dans divers domaines qui lui font appel, dans notre projet nous nous intéressons à l'application de ce paradigme dans les systèmes ERP, c'est pour cela que nous avons consacré ce chapitre pour justifier l'utilité de l'intégration d'un système multi agent dans les systèmes ERP.

Ce chapitre contient deux parties, la première présente des généralités sur le paradigme de système multi agent comme branche de l'intelligence artificielle distribuée. Ce système qui est un ensemble hétérogène d'applications encapsulé (agents) qui participent au processus de prise de décision, communiquent, collaborent et négocient afin de répondre chacun à ses propres objectifs de conception mais aussi à un objectif qui est partagée au sein de la communauté.

La deuxième partie du chapitre contient l'apport des ERP aux systèmes multi agent, est une exposition de deux projets visant à intégrer l'approche multi agents dans les systèmes ERP. Enfin nous avons donné notre synthèse sur les approches utilisées et nous avons expliqué l'approche que nous avons proposée.

Le chapitre suivant va contenir les détails sur le modèle projeté.



Chapitre 3

Modélisation

1 Introduction

Le principal intérêt des SMA est qu'ils permettent de distribuer des entités communicantes et autonomes, dotées de compétences. Pour réaliser un SMA selon ces critères, il faut définir l'architecture de ses agents et la structuration des connaissances nécessaires pour leurs différentes activités.

Le modèle que nous proposons pour l'ERP est un système multi agent du type cognitif, les systèmes d'agents cognitifs sont composés d'un petit nombre d'agents coopérants afin d'aboutir à un objectif global par l'exécution d'opérations complexes. Ces agents là disposent d'une capacité de raisonnement sur une base de connaissances, ils traitent les informations relatives au domaine d'application, et à leur environnement. Chaque agent représente un système expert, qui est un logiciel ayant pour but la modélisation du comportement d'un expert humain. Le modèle de base d'un système expert est composé de deux éléments essentiels la base de connaissance et le moteur d'inférence.

D'autre part un agent cognitif fonctionne généralement comme suit ; il va acquérir des informations sur son environnement par la perception et la communication avec d'autres agents, et ainsi mettre à jour sa base de connaissances, d'après la quelle il va décider quel but à réaliser et donc proposer les plans à suivre, ensuite les tâches appropriés vont être exécutées et donc affecter le monde extérieur.

Sur ces principes nous nous sommes basés pour construire notre système multi agent qui est destiné aux applications ERP, et que nous allons détailler dans la suite de ce chapitre.

2. Présentation générale du modèle proposé

L'ERP comme nous l'avons déjà présentée dans le premier chapitre est un ensemble de modules fonctionnels spécialisé qui tourne autour d'une base de donnée unique, pour ce fait Et en se basant sur l'approche S.M.A qui fait distribuer l'expertise sur un ensemble d'agents qui modélisent les différents modules de l'ERP, nous proposons l'architecture de notre système qui est illustré dans la figure 16.

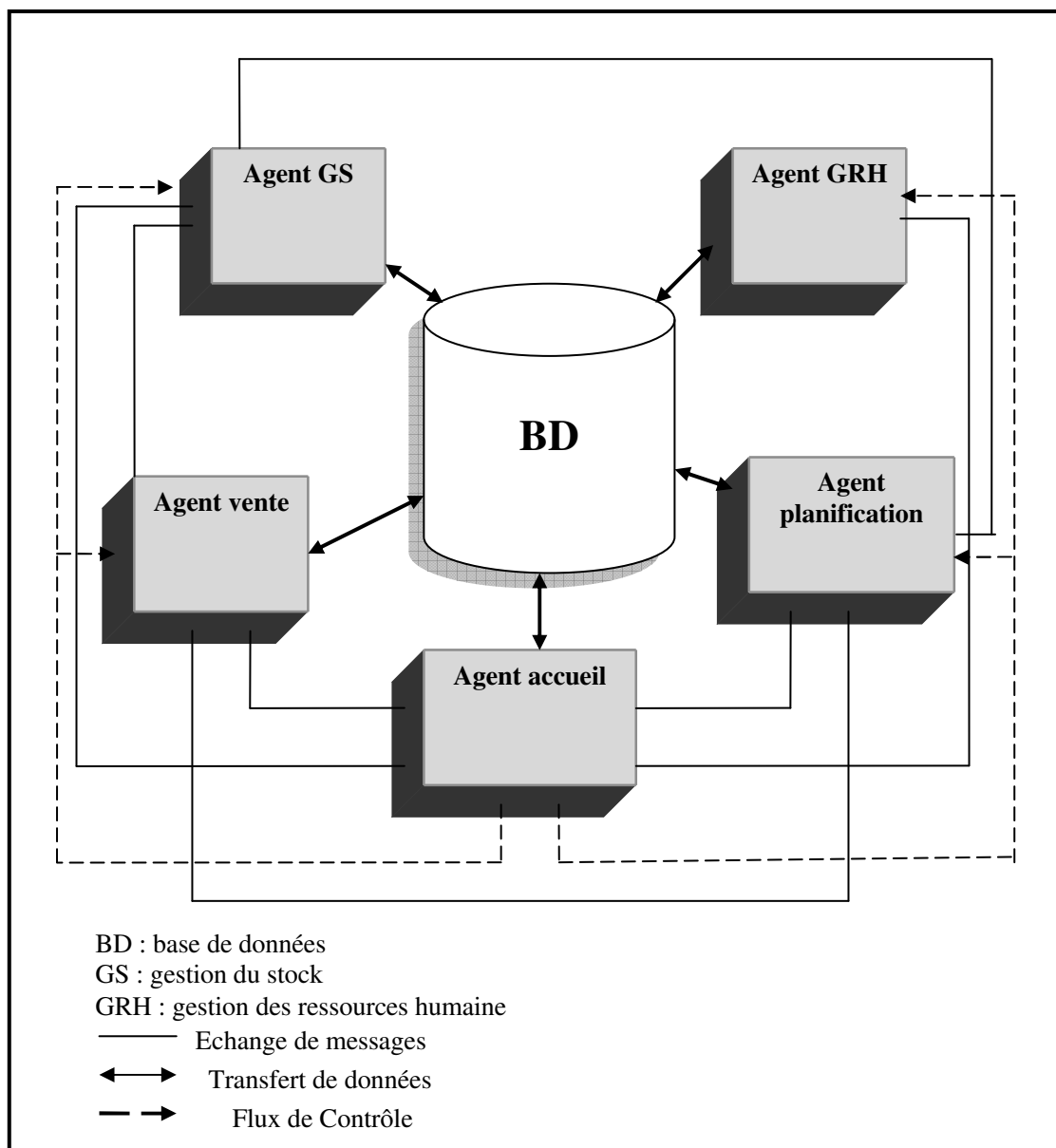


Figure 1 : schéma représentant la structure du modèle

Donc notre système comprend un ensemble d'agents cognitifs chacun expert dans un domaine utile au fonctionnement de l'entreprise, ces agents là appartiennent à cinq classe distinct qui sont :

- La classe d'agent GS : c'est l'agent responsable de la gestion du stock.
- La classe d'agent GRH : responsable de la gestion des ressources humaines.
- La classe d'agent planification : prend en charge la fonction de planification de l'entreprise.
- La classe d'agent vente : c'est l'agent responsable du processus de vente de produits.
- La classe d'agent accueil : qui contrôle l'accès aux différents agents.

Cette architecture garantie l'indépendances des module de l'entreprise et en même temps les garde en contacte puisque les fonction d'une entreprise ne doivent pas être isolées mais plutôt complémentaires, ce contacte est assuré via la communication des agents entre eux dans le cas de notre système il s'agit d'une communication par envoi de messages.

Le flux de contrôle indiqué dans le schéma, représente le contrôle d'accès qu'effectue l'agent administrateur sur les autre agents et qui garantie la sécurité du système.

De plus la base de données unique et centralisé qui garantie l'unicité et l'intégrité de l'information est géré bien sure par un système de gestion de base de données qui permet le transfert de ces dernière depuis et vers les agent du système.

3. Présentation détaillée des agents de notre modèle

3.1. Agent accueil (AC)

Il joue le rôle de l'interface qui relie l'utilisateur aux différents agents de l'ERP, il contrôle l'accès à ces derniers et assure la sécurité du système, en protégeant les comptes utilisateurs.

Dans le cas de notre système il existe deux types d'utilisateurs, le type employé et celui de l'administrateur :

- Le type Administrateur qui entre en mode réglage des paramètres.
- Le type Employé qui accède directement aux différentes fonctions des agents, ce type est réparti en quatre comptes utilisateur chacun propre à un agent précis, ces comptes sont protégés chacun par un mot de passe pour des raisons de sécurité.

3.1.1. Architecture de l'agent accueil

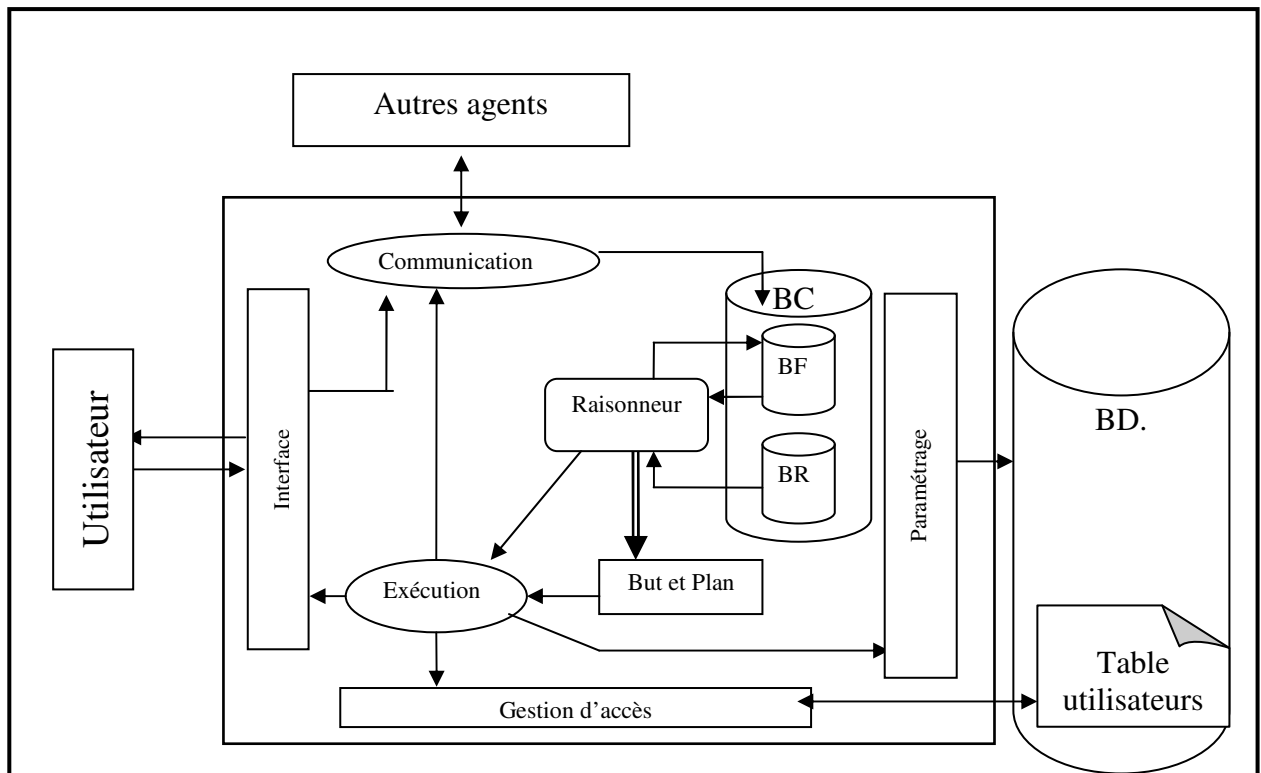


Figure 2: Architecture de l'agent accueil

L'architecture des agents de notre système est inspirée des systèmes experts, pour ce fait ils sont composés d'une base de connaissance notée «BC » qui à sont tour contient une base de règle «BR » et une base de fait «BF », de plus le deuxième composant de notre système expert est un moteur d'inférence qui est nommé « raisonneur », le raisonneur utilise la base de connaissances pour son raisonnement dans le but de déterminer le plan et contrôler l'exécution de ses actions, qui sont réparti sur trois modules dans le cas de l'agent accueil :

- **Un module interface :** responsable de la liaison de l'utilisateur avec l'agent, il collecte les données entrée par l'utilisateur, et saisie la demande d'activation de l'un des agents du système selon le choix de l'utilisateur bien sure.
- **Un module gestion d'accès :** qui gère les droit d'accès, cet agent effectue l'identification des utilisateurs, vu que l'accès à chaque agent est protégé par un nom d'utilisateur et un mot de passe pour cela le module de gestion d'accès fait appel à la base de donnée unique et centralisé du système et plus précisément la table utilisateurs.

- **Un module de paramétrage** : ce module est accessible seulement par l'utilisateur du type administrateur, il permet d'initialiser la base de données par les informations de base nécessaires au fonctionnement, tel que les salaires de base pour chaque grade d'employé, et création de comptes utilisateur du type employé.

3.1.2. Fonctionnement de l'agent accueil

Afin de réaliser son but l'agent accueil suit les étapes suivantes dans son fonctionnement :

a) Le cas d'un utilisateur du type employé

- Saisie du choix de l'utilisateur
- Selon le choix il présente le test d'identification associé.
- Une fois l'utilisateur identifié, il envoie une requête d'activation de l'agent choisi

b) Le cas d'un utilisateur du type administrateur

- Test d'identification.
- Une fois l'administrateur identifié présentation des fonctions du module de paramétrage.
- Saisie du choix de l'administrateur.
- Selon ce choix appel à la fonction adéquate.
- Saisie des informations d'initialisation.
- Mettre à jour la base de données du système.

Les fonctions principales de cet agent sont les suivantes :

a) L'identification

- ✓ Saisie du nom d'utilisateur et du mot de passe.
- ✓ Interrogation de la table utilisateurs qui contient les informations des comptes utilisateur.
- ✓ Comparaison des données saisies avec ceux de la base de données.

b) Le paramétrage il comporte trois fonctions qui sont

b.1) établissement des comptes utilisateur

- ✓ Saisie des mots de passe pour chaque agent.
- ✓ Les insérer dans la table des utilisateurs.

b.2) établissement des salaires de base

- ✓ Saisie des salaires de base entrée par l'utilisateur pour chaque grade.

- ✓ Les insérer dans la table des employés.

b.3) Consultation

- ✓ Saisie du choix de l'utilisateur.
- ✓ Interrogation de la base de données et l'affichage de la table choisi par l'utilisateur.

3.1.3. Le savoir de l'agent accueil

Comme tout système expert les agents de notre système ont besoin de savoir afin d'accomplir leurs raisonnements, le savoir de l'agent accueil est le suivant :

- *Si* l'utilisateur est identifié *Alors* lui permettre l'accès aux fonctions du système.
- *Si* l'utilisateur n'est pas identifié *Alors* afficher un message d'erreur
- *Si* l'utilisateur est l'administrateur *Alors* lui permettre l'accès aux fonctions de paramétrage.
- *Si* l'administrateur demande de consulter une table de la base de données *Alors* afficher la table demandée.
- *Si* l'utilisateur choisi un agent x et il est identifié au niveau de cette agent *Alors* lancer la requête d'activation de l'agent x.

3.2. L'agent vente (VEN)

Cet agent se charge du côté commercial de l'entreprise il prend en charge la gestion des clients car ils garde leurs coordonnées dans la table des clients qui réside dans la base de données du système aussi les commandes des clients sont enregistrées dans une table spécifique, encore l'agent vente établit ces commandes et effectue le suivi du processus de vente.

3.2.1 Architecture de l'agent vente

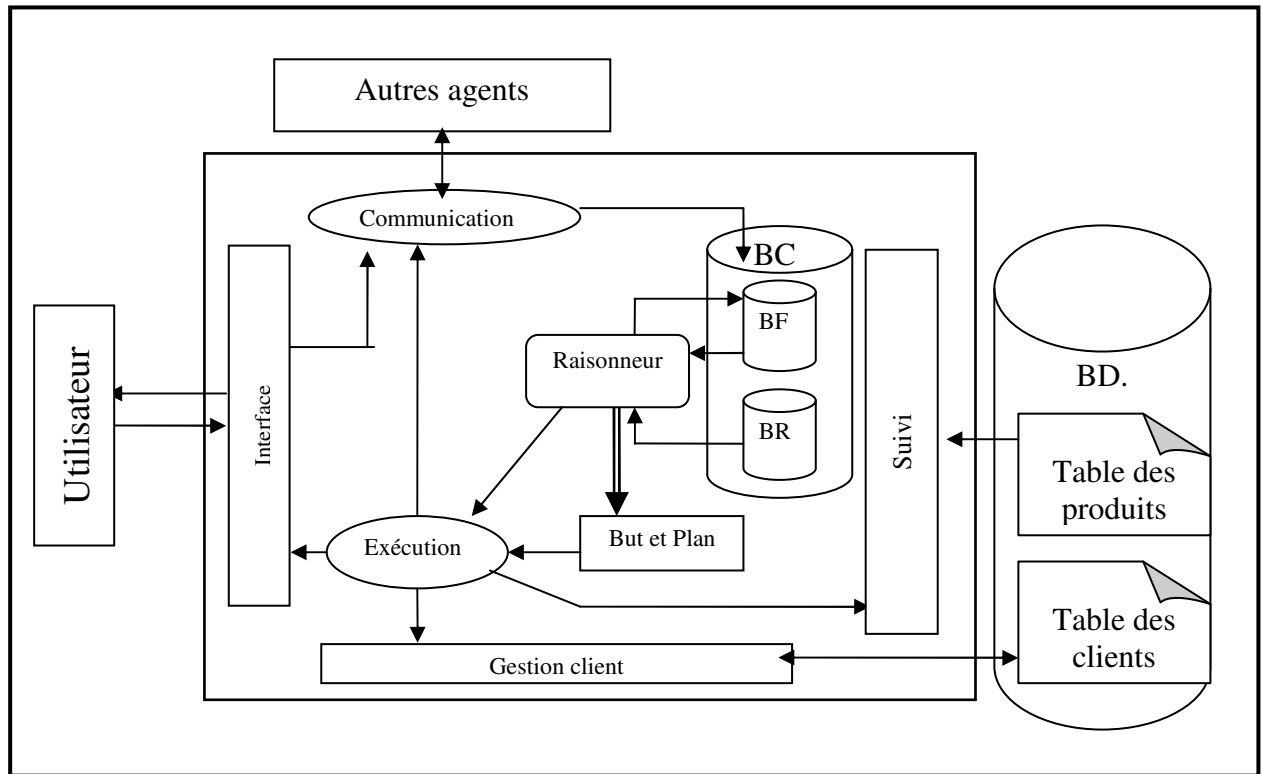


Figure 3: Architecture de l'agent vente

L'agent vente comporte de plus de ses composants de bases trois modules qui lui font distingué, et qui assurent son fonctionnement. Ces modules sont :

- **Un module interface** : fournit une interface entre l'utilisateur et l'agent, il saisie les données entrée par l'utilisateur, ces données concerne soit les client ou bien leurs demandes. Aussi il peut saisir la demande de consultation de l'état commercial de l'entreprise ; par exemple ce qui concerne le budget et le taux de vente, et suite à cette demande il affiche la table associée.
- **Un module gestion des clients** : ce module à pour rôle l'inscription des clients ce qui veut dire la collecte d'information sur les clients, mais aussi l'enregistrement de leurs demandes.
- **Un module de suivi** : c'est au niveau de ce module que le processus de vente est effectué, selon la commande cet agent vérifie la disponibilité des produit requis par le client en accédant à la table de produit existant en stock, ensuite il gère la vente en établissant la facture et la mise à jour du montant de la caisse. Dans le cas ou le produit demander n'existe pas cet agent envoi une requête à l'agent de planification.

3.2.2. Fonctionnement de l'agent vente

Les étapes que suit l'agent vente dans son fonctionnement sont les suivantes :

- La saisie du choix de l'utilisateur
- Selon le choix il fait appel à l'une des suites d'instructions suivantes :

a) Le cas de consultation

- Saisie du choix des informations à consulter
- Accès à la base de données puis affichages de l'information demandée.

b) Le cas d'une commande

- Saisie des informations du client
- Gestion du client
- Saisie de sa commande
- Enregistrement de la commande
- Interrogation de la base de donnée et vérification de la disponibilité du produit.
- Dans le cas d'une réponse affirmative, réalisation de la facture et son affichage
- Suite à une confirmation du client et paiement l'utilisateur ie l'employé mis à jour le contenu de la caisse
- Ensuite envoi d'une requête à l'agent de gestion du stock pour le retrait des produits vendus.
- Dans le cas où le produit commandé est inexistant, lancement d'une requête à l'agent de planification pour connaître le délai où le produit sera prêt.

3.2.3. Le savoir de l'agent vente

Le savoir qu'utilise l'agent vente pour son raisonnement est le suivant :

- *Si* le client est identifié *Alors* insérer sa commande dans la table des commandes associée à ce client.
- *Si* l'utilisateur n'est pas identifié *Alors* l'inscrire.
- *Si* les produits en stock répondent en genre et en nombre à la commande du client *Alors* établir le processus de vente.

- *Si* les produits en stock ne répondent pas soit en genre ou en nombre à la commande du client *Alors* lancer une requête de planification des délais à l'agent planification.

3.3. Agent gestion de stock (GS)

L'agent de gestion du stock a pour charge le suivi du magasin, c'est-à-dire les produits qui entrent et qui sortent des magasins aussi ce qui concerne les matières premières nécessaires à la fabrication des produits.

3.3.1. Architecture de l'agent GS

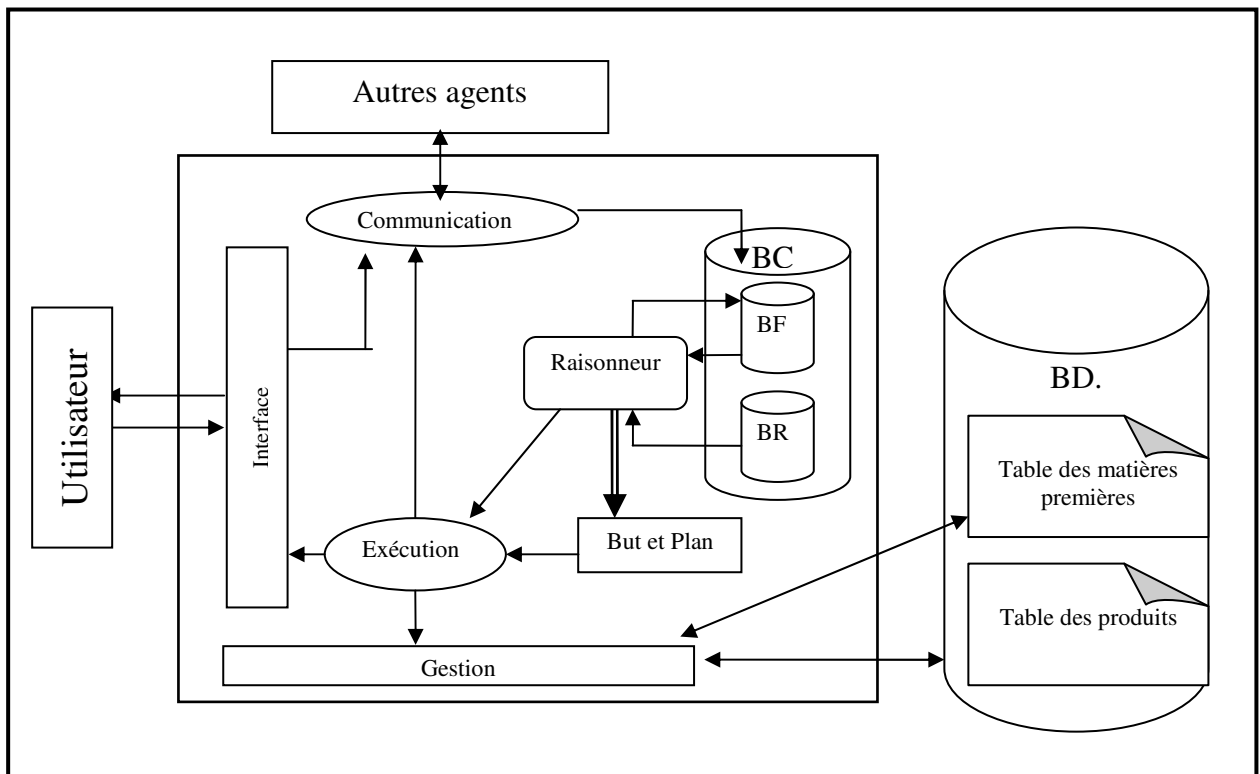


Figure 4: Architecture de l'agent gestion de stock

L'agent GS répartit son fonctionnement sur deux modules qui sont :

- **Un module interface** : cet interface permet, la saisie des informations sur les produits qui sont entrées par l'utilisateur, il permet aussi de consulter les tables contenant ces informations aux choix de l'utilisateur.
- **Un module gestion** : possède trois tâches à accomplir, l'enregistrement des nouveaux produits venant en stock ainsi que les informations qui les accompagnent, la deuxième est la mise à jour des bases de données dans le cas d'arrivée de nouveaux matériels au magasin, et la dernière c'est le retrait de produits.

3.3.2. Fonctionnement de l'agent GS

Afin d'accomplir ses tâches l'agent de gestion du stock suit les étapes suivantes, sachant que cet agent fonctionne selon deux modes :

a) Le premier mode

Dans ce cas l'agent GS est activé par l'utilisateur, et il fonctionne comme suit :

- Saisie du choix de l'utilisateur
- Selon ce choix l'agent fait appel à l'une des trois fonctions suivantes :
 - 1) **ajout nouveau matériel**
 - ✓ Saisie des informations sur le nouveau matériel
 - ✓ L'insérer dans la table adéquate selon qu'il soit un produit ou une matière première
 - 2) **Entrer en stock**
 - ✓ Recherche du matériel dans les tables du stock.
 - ✓ Mettre à jour le champ du nombre du produit en y ajoutant le nombre entrant.
 - 3) **Retrait**
 - ✓ Recherche du matériel dans les tables du stock
 - ✓ Mettre à jour le champ du nombre du produit en effectuant la soustraction du nombre retiré à partir du nombre existant.

b) le deuxième mode

Dans ce cas l'agent GS est activé suite à une demande de l'agent vente dans le but d'accomplir le processus de vente, et il fonctionne comme suit :

- réception de la requête envoyée par l'agent vente via le protocole de communication.
- Identification de cette requête et extraction des informations nécessaires
- Lancement de la fonction de retrait en utilisant les informations extraites.

3.3.3. Le savoir de l'agent GS

Le savoir de l'agent GS est composé des points suivants :

- *Si* l'agent GS reçoit un message de la part de l'agent vente *Alors* exécuter le retrait.
- *Si* l'agent GS reçoit un message de la part de l'agent accueil *Alors* saisie du choix de la fonction à exécuter.
- *Si* l'utilisateur choisi de consulter les informations du stock *Alors* afficher les tables associées adéquates à son choix.

3.4. L'agent planification (PLA)

Cet agent est destiné à établir les délais de production, et indiquer le manque en matière première.

3.4.1. Architecture de l'agent planification

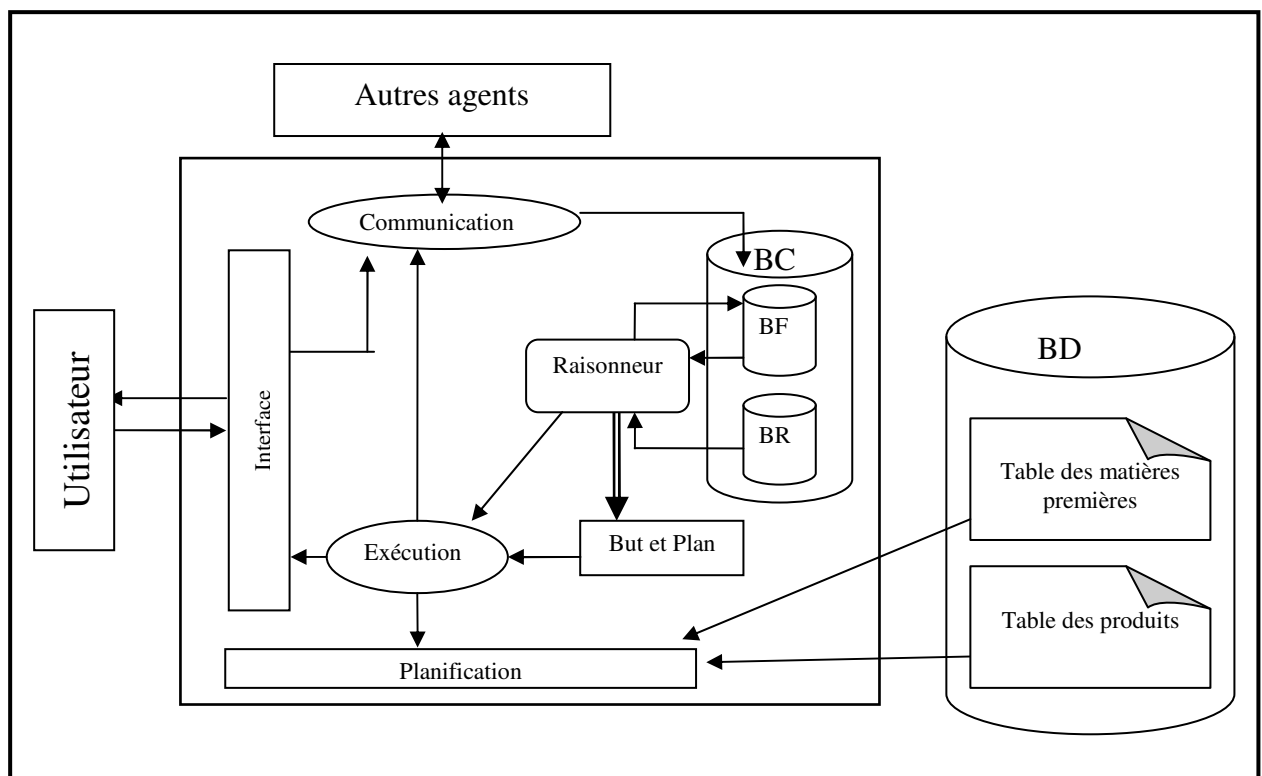


Figure 5: Architecture de l'agent planification

Le fonctionnement de l'agent de planification s'effectue au niveau des deux modules suivants :

- **Un module interface** : qui saisie les informations nécessaires à la planification
- **Un module planification** : servant à calculer les délais de production d'un produit en utilisant les informations contenues dans la table des produits et celle des matières premières.

3.4.2. Fonctionnement de l'agent planification

L'agent de planification est activé de deux manières, pour cette raison il possède deux cas de fonctionnement

a) cas d'activation par l'agent de vente

- Réception de la requête envoyée par l'agent de vente
- Extraction des informations contenues dans cette requête
- Recherche du produit en question dans la table des produits et consulter le champ du délai de sa fabrication
- Calcul du délai de fabrication en fonction du nombre de produit commandé tel que la requête l'indique
- Affichage du rapport du résultat.

b) cas d'activation par le choix de l'utilisateur

- Saisie du produit et son nombre entrés par l'utilisateur
- Recherche du produit en question dans la table des produits et consulter le champ du délai de sa fabrication
- Calcul du délai de fabrication en fonction du nombre de produit commandé tel que la requête l'indique
- Affichage du rapport du résultat.

3.4.3. Le savoir de l'agent planification

Le savoir de cet agent peut être comme suit :

- **Si** l'agent planification reçoit un message de la part de l'agent vente **Alors** exécuter le plan ' a '.
- **Si** l'agent planification reçoit un message de la part de l'agent accueil **Alors** exécuter le plan ' b '.

- Si il y a un manque dans les matières premières *Alors* ajouter le délai de procuration de cette matière première fois le nombre nécessaire.

3.5. L'agent gestion des ressources humaine (GRH)

L'agent de gestion des ressources humaine est responsable de ce qui concerne le personnel de l'entreprise, ils se charge de la gestion du paiement ainsi que les augmentation de grade

3.5.1. Architecture de l'agent GRH

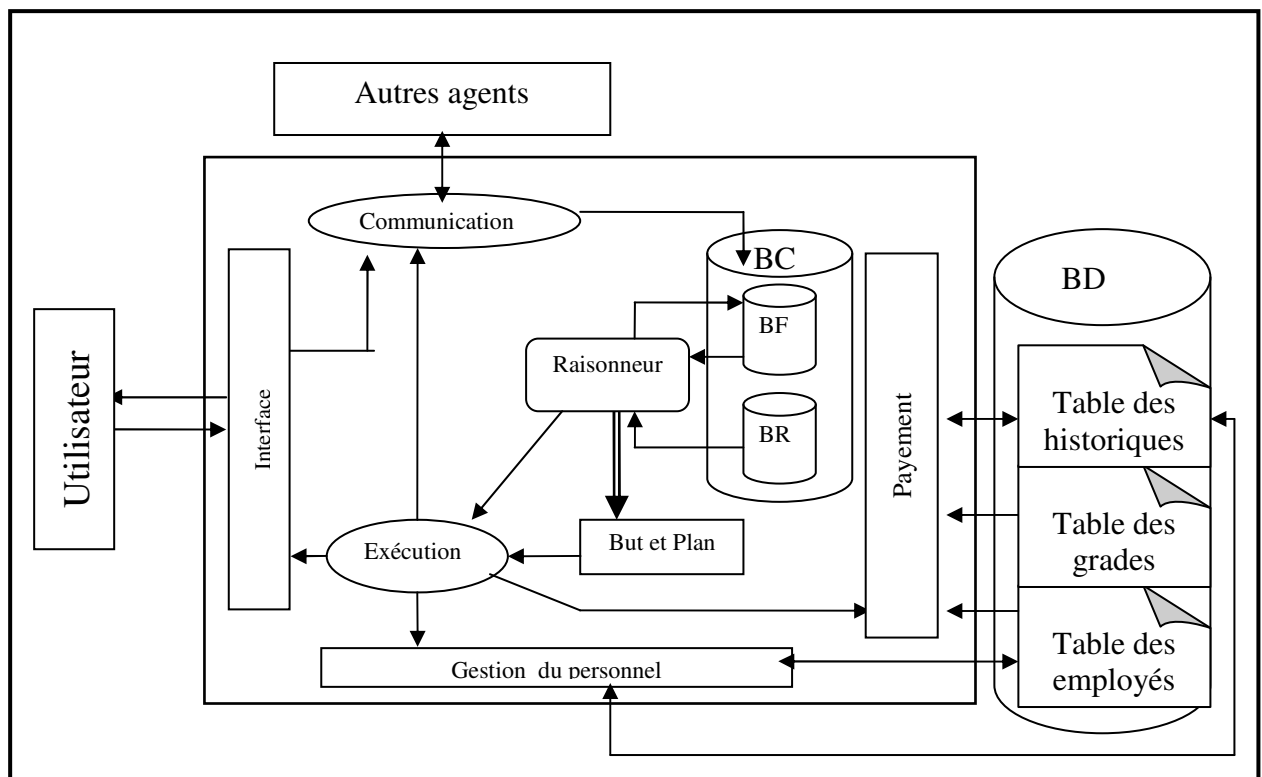


Figure 6: Architecture de l'agent GRH

Les modules fonctionnels propre à l'agent GRH sont les suivants :

- *Un module interface* : à pour rôle l'acquisition des informations saisie par l'utilisateur, et qui sont les informations concernant les employés de l'entreprise il permet aussi de consulter ces informations à la demande de l'utilisateur par l'affichage de la table des historiques, ou celle des employés.

- **Un module gestion du personnel:** ce module a pour charge l'ajout d'enregistrement lors du recrutement d'un nouveau employé, cet enregistrement contient les informations nécessaires à l'identification de l'employé et son état dans l'entreprise, nous parlons de son grade et le département au quel il appartient.

Aussi il fait appel à une fonction qui gère les absences et les sauvegarde dans la table des historiques.

En plus il sert dans le cas d'augmentation du grade d'un employé

- **Un module payement :** responsable du calcul de la paye en fonction de grade de l'employé et s'il a reçu une prime, et s'il a absenté durant le mois. De plus il garde un historique des primes virer et la paye dans une table de la base de donnée.

3.5.2. Fonctionnement de l'agent GRH

L'agent GRH fonctionne comme suit :

- Saisie du choix de l'utilisateur

a) Dans le cas du recrutement d'un nouvel employé

- Saisie des informations concernant l'employé via un formulaire
- Ajout des informations saisies à la base de données, précisément la table des employés

b) En ce qui concerne le payement

- Identification de l'employé
- Consultation de son grade dans la table des employés
- Recherche du salaire de base associé au grade dans la table des grades
- Lui soustraire le pourcentage des absences
- Versement du montant du salaire dans le compte de l'employé
- Sauvegarde de l'information dans la table des historiques
- Soustraction du montant du salaire de la caisse de l'entreprise

c) En ce qui concerne les prime même processus que celui du payement

d) En ce qui concerne les absences

- Saisie du nombre de jours absenté
- Mettre à jour la table des employés, en ajoutant le nombre de jours absenté au contenu du champ correspondant.

3.5.3. Le savoir de l'agent GRH

- *Si* l'utilisateur choisi de consulter les informations concernant le virement des payes *Alors* afficher la table des historiques.
- *Si* l'utilisateur choisi d'ajouter un nouvel employé à la base *Alors* lui fournir le formulaire d'information à remplir ensuite insérer les informations du formulaire dans la table des employés
- *Si* l'utilisateur vire la paye dans le compte de l'employé *Alors* soustraire la somme de la caisse de l'entreprise.

4. Le mécanisme de communication et de contrôle utilisés

Le type de communication entre agents que nous avons utilisé est une communication par envoi de message qui répond aux normes du langage de communication FIPAACL, dont la syntaxe est comme suit :

```
(request
  :sender Agent_A
  :receiver Agent_B
  :content
  (...)
  :in-reply-to action
  :replay-with reponse
  :language FIPA-SLO
)
```

Figure 7: Syntaxe d'un message FIPA-ACL

Nous avons choisi cette syntaxe, parce que notre système est destiné à être implémenter sous la plateforme jade qui garantie le contrôle du comportement des agent et supporte les message FIPAACL.

5. Fonctionnement général du système

Pour mieux expliquer la structure ainsi que le fonctionnement de notre système nous avons fait appel au langage AUML, pour la modélisation de notre système d'agent :

En premier lieu, nous allons décrire les classes d'agents existants et les relations qui les lient par le diagramme de classe AUML suivant :

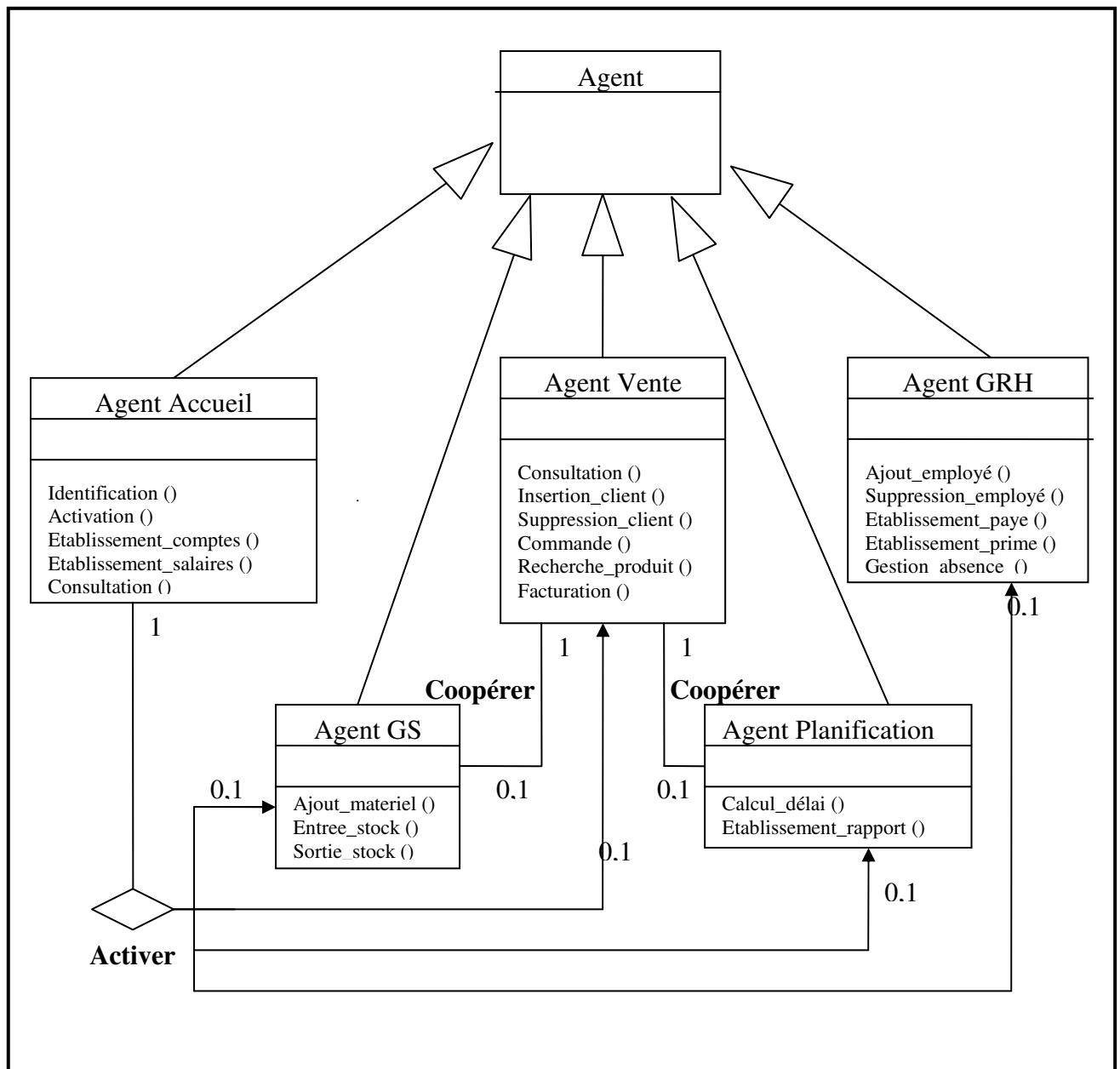


Figure 8 : diagramme de classe AUML du modèle proposé

Par la suite nous allons décrire le processus de fonctionnement général du système en faisant appel au diagramme de séquence AUML.

Le premier contact de l'utilisateur avec le système sera via l'agent accueil qui va contrôler l'accès aux autres agents pour qu'ils accomplissent les tâches qu'ils ont en charge, ce processus est décrit dans la figure suivante :

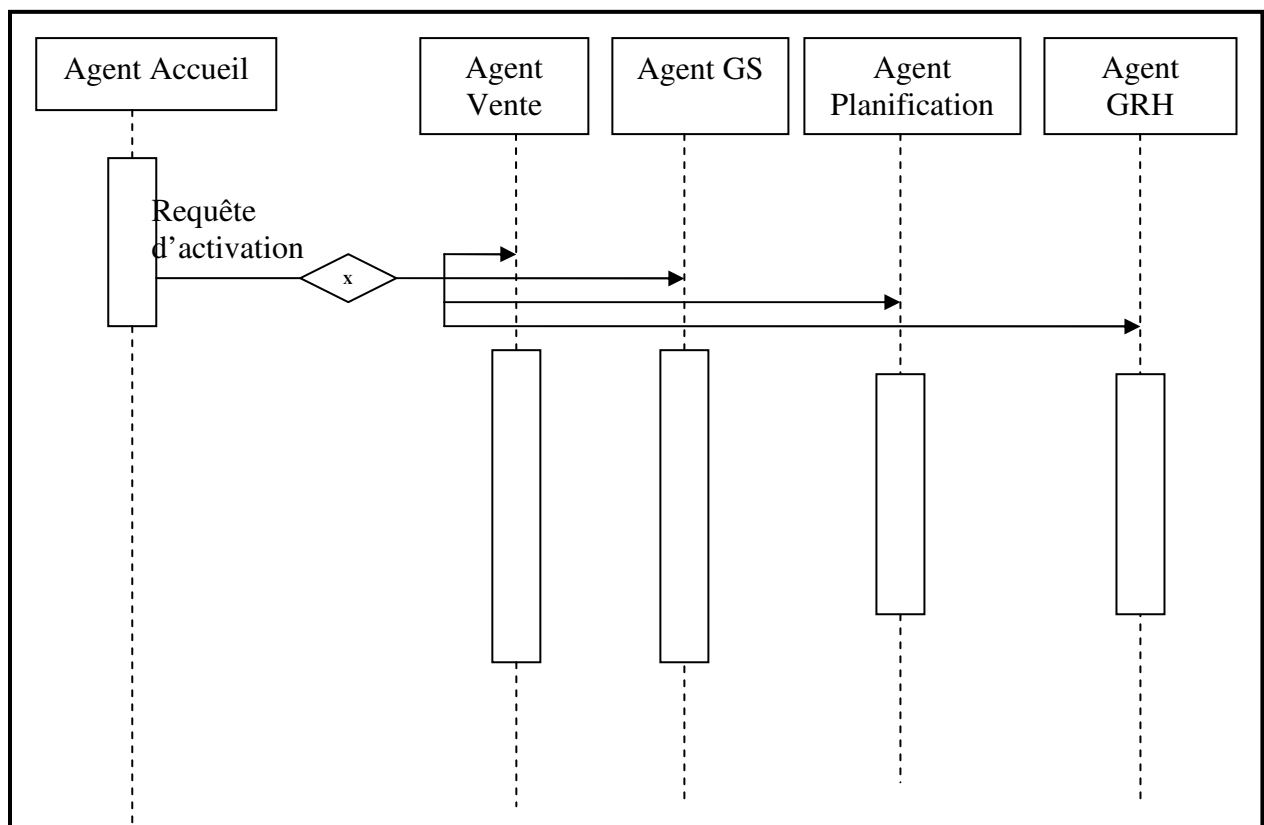


Figure 9 : diagramme de séquence AUML centré sur l'agent accueil

A l'arrivée d'une commande l'utilisateur active l'agent vente à travers l'agent accueil bien sûr, et suit le processus illustré dans le diagramme suivant :

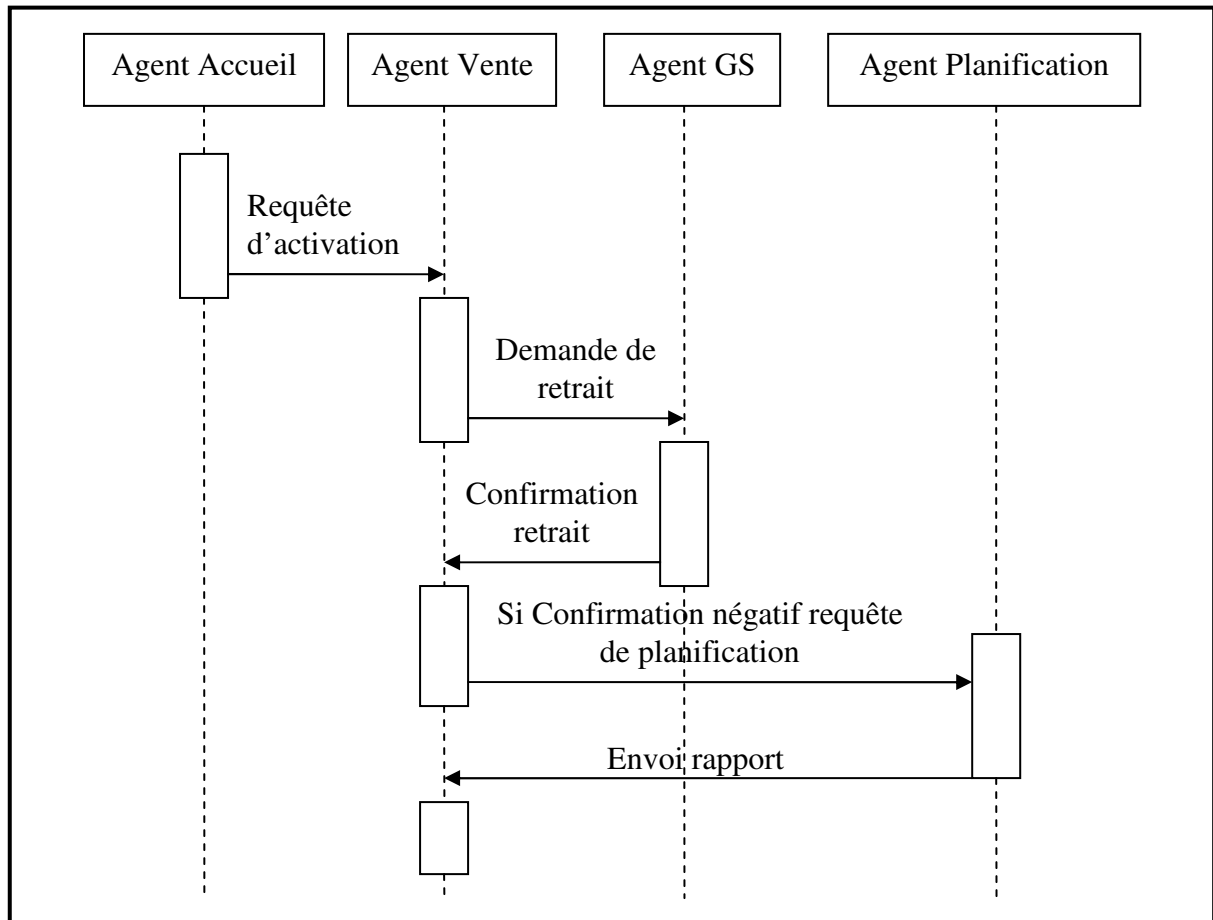


Figure 10 : diagramme de séquence AUML dans le cas d'une commande

Dans cette partie nous avons renforcé la présentation du processus principal du fonctionnement de notre modèle par l'utilisation des diagrammes AUML, qui ont montré que les agents de ce modèle sont dotés d'un bon degré d'indépendance, le nombre d'interactions est minimum due à la centralisation de la base de données et son unicité, et donc la complexité du système sera minimiser.

6. Conclusion

Dans ce chapitre nous avons expliqué avec plus de détail l'approche que nous proposons, ensuite nous avons présenté l'architecture globale de notre système, suivi par une illustration détaillée de l'architecture de chaque agent en expliquant son rôle et la manière dont il fonctionne pour accomplir son but.

Nous avons utilisée une architecture à base d'agent coopérants pour bénéficier de l'autonomie, la modularité, la distribution et l'intelligence des agents, dans le domaine des systèmes ERP, puisque cette approche apporte de la souplesse du fonctionnement pour ce type de système et répond parfaitement à leurs nature complexe et en pleine évolution.



Chapitre 4
Etude de cas et
validation

1. Introduction

Après avoir présenter notre modélisation de l'approche que nous avons proposée suite à une recherche approfondie, nous passons à la réalisation d'un exemple illustratif en utilisant des outils que nous jugeons performant pour notre solution, l'ensemble de ses outils forme notre environnement de développement dont les caractéristique seront citées.

2. Environnement de développement

Nous utilisons pour le développement de notre système un environnement JAVA qui est le Borland Jbuilder la neuvième édition, accompagné du système de gestion de base de données MySQL.

Nous avons utilisé le langage JAVA du fait qu'il fournit plusieurs avantages le plus important est la portabilité de plus :

- ✓ **Distribué** : les fonctionnalités réseau de Java sont à la fois fiables et facile à utilisées.
- ✓ **Fiabilité** : car il intègre un modèle de pointeur qui minimise les risques d'écrasement de la mémoire et d'endommagement de données.
- ✓ **Multi thread** : les avantages du multi thread sont une meilleure inter réactivité et un meilleur comportement en temps réel [44].

Aussi notre système multi agents est réalisé et géré par la plateforme JADE, nous utilisons la dernière version qui est la « 6.3 ».

2.1. Présentation de la plateforme JADE

La plateforme JADE (Java Agent Development Environment) est une plateforme destinée à la création d'agents logiciels et leurs gestion tout en respectant la norme FIPA, cette plateforme est implémentée en langage java ce qui garantie sa portabilité ; elle fournit un ensemble d'outils facilitant la mise en œuvre des systèmes multi agents, elle comprend :

- * Un environnement d'exécution où les agents JADE peuvent vivre.
- * Une suite d'outils graphiques permettant l'administration et le suivi des activités et de l'exécution des agents.
- * Une bibliothèque de classes nécessaire à la programmation des agents [45].

Le fait que cette plateforme est conforme à la norme FIPA qui a pour objectif de garantir l'interopérabilité entre les systèmes autonomes elle est dotée des outils suivants :

- Gestion des agents
- Transport de messages
- Langage de communication d'agents
- Langage a contenu sémantique
- Protocoles d'interaction
- Ontologie
- Et sécurité

Et par la suite les services fournis par une plateforme conforme à la norme FIPA et qui sont implémentés comme des agents :

- ***Système de Gestion d'Agents (AMS)*** : C'est un agent central qui supervise les autres agents et les accès à la plate-forme c'est-à-dire qu'il fournit un service similaire à celui des Pages Blanches.
- ***Canal de Communication entre Agents (ACC)*** : un agent fournissant la route pour les interactions entre agents dans et en dehors de la plateforme.
- ***Facilitateur d'Annuaire (DF)*** : agent qui fournit permet le recensement des services disponibles ce qui ressemble au service des Pages Jaunes.

D'un autre côté la plateforme jade possède son propre nombre d'agents garantissant chacun un service différent, mais dont l'ensemble forme l'outil graphique de la plateforme, et qui sont présentés par la suite :

- ***Le Remote Management Agent (RMA)*** : cet agent agit en tant que console graphique pour la gestion et le contrôle de la plateforme.
- ***L'agent Dummy*** : c'est un agent composé d'une interface graphique et d'un sous-agent, il a pour rôle le débogage et le suivi des agents, aussi il envoie et reçoit des messages.
- ***L'agent Sniffer*** : permet la visualisation des interactions des agents, et de garder un historique sur ces interactions, en d'autres termes il surveille les échanges de messages dans une plateforme.

- *L'agent Introspector* : celui là prend en charge la surveillance de l'état des agents durant leurs cycles de vie, mais aussi les message émis et reçus.
- *L'agent DF GUI* : cet agent à pour objectif d'inspecter le service des pages jaunes [46].

2.1.1. L'environnement d'exécution JADE

Un des avantages de la plateforme JADE est de garantir la distributivité des agents et ce se matérialise par la notion de conteneur.

Un environnement d'exécution est appelé un conteneur ; chaque conteneur peut contenir plusieurs agents. L'ensemble des conteneurs actifs constitue une plateforme. Cette plateforme peut s'exécuter sur plusieurs machines, elle contient toujours un conteneur particulier qui est dit principal (*Main Container*) et qui regroupe les agents fournissant les services de base de la plateforme.

La plateforme d'agents peut être distribuée sur plusieurs machines même dans le cas où ces derniers possèdent des systèmes d'exploitations différents.

L'architecture d'une plateforme JADE est basée sur la coexistence de plusieurs Machines Virtuelles (VM) Java et la communication entre elles se fait par la méthode RMI (Remote Method Invocation) de Java ; qui est une API Java permettant de manipuler des objets distants instanciés sur une autre machine virtuelle, se situant sur une autre machine du réseau comme si l'objet était sur la machine virtuelle de la machine locale.

2.1.2. La communication entre les agents JADE

La plateforme JADE offre à ses agents la possibilité de communiquer via le langage de communication FIPA-ACL.

Chaque agent possède une file d'attente où les messages qu'il reçoit sont stockés, la plateforme garantie le dépôt d'un message dans la file d'attente ; dès lors l'agent récepteur sera prévenu de son arrivée et par la suite il peut prendre ce message au moment fixé par le programmeur.

Il existe plusieurs types de communications, selon la position des l'agents dans le système, nous citons :

- **Même conteneur** : dans ce cas on a pas besoin d'invocation distante.
- **Plusieurs conteneur dans la même plateforme avec cache** : dans ce cas l'RMI est invoqué une seule fois afin de sérialisé et désérialisé le message ACL.
- **Plusieurs conteneur dans la même plateforme sans cache** : Deux appels RMI, le premier pour mettre à jour le cache depuis la table descriptive global des agents, et le second pour envoyer le message; l'objet qui décrit l'agent et qui est retourné du premier appel est sérialisé et désérialisé, puis le message ACL suit le même chemin.
- **Plusieurs plateformes JADE** : Un appel direct à distance est effectué vers le ACC, ce qui provoque une invocation à distance par la méthode CORBA de l'OMG (Object Management Group) qui est une méthode de communication permettant de manipuler des objets à distance avec n'importe quel langage contrairement à la méthode RMI qui est une solution tout Java, CORBA est beaucoup plus compliqué à mettre en oeuvre, c'est la raison pour laquelle de nombreux développeurs se tournent généralement vers la RMI.

Suite à l'invocation CORBA déjà citée, nous passons à une transformation d'un objet java vers une chaîne de caractères Java, ensuite vers un flux de byte IIOP sur le côté expéditeur et une transformation inverse du coté récepteur.

- **Plusieurs plateformes non JADE**: la même chose que pour le cas précédent, mais ce qui se passe à l'autre bout de la liaison dépend de la nature et l'implémentation de l'autre plate-forme ; celle qui reçoit le message [47].

Donc l'utilisation de cette plateforme, nous facilite la programmation efficace d'un système multi agents, de plus elle garantie la distributivité et ainsi la communication sur divers postes de travail sur le réseau, sans oublier le fait qu'elle fournit toutes les caractéristique dont nous avons besoin pour, modéliser le concept d'agents.

3. Etude de cas

Dans le but de bien expliquer l'approche que nous avons adoptée et bien illustrer notre conception, nous avons essayé d'établir un exemple en faisant une petite étude sur une entreprise à Biskra et nous avons aussi essayé d'appliquer notre approche sur elle. Ce que nous allons expliquer dans la partie suivante.

3.1. Présentation générale de l'entreprise

L'entreprise que nous étudions est une entreprise de production de matériaux de construction qui porte le nom de "*Technique sud construction*" à son propriétaire Mr *Chaabi M^{ed} Taher*, et qui se situe au Quartier Boukhari à Biskra. Cette entreprise est une PME qui a prouvé sa compétence dans le marché national et surtout dans le sud de l'Algérie ; elle produit différents matériaux nécessaires à la construction, comporte un nombre réduit d'employés, et possède ses propres machines qui lui sont utiles pour la production ainsi que pour transporter les matières premières.

3.2. Fonctionnement de l'entreprise et application de l'approche

Le fonctionnement de l'entreprise repose sur trois points essentiels celui de production, de vente, et la gestion du personnel. Dans cette partie nous allons définir les différents agents de la conception que nous avons proposée de telle façon qu'ils répondent aux besoins fonctionnels de l'entreprise que nous étudions, commençons par l'agent AC.

3.2.1. Diagramme de classe UML de la base de données

Le schéma suivant présente le diagramme de classe UML de la base de données centralisé de l'ERP, ce diagramme explicite l'organisation des tables de cette base ainsi que les relations qui les lient.

Le schéma précédent représente une récapitulation de la partie qui va suivre ; où nous allons bien détailler chacune des tables ainsi que son utilité dans le système.

3.2.2. Les agents de l'entreprise

3.2.2.1. L'agent AC

L'agent AC est exactement comme nous l'avons décrit dans le chapitre précédent, la table utilisateurs qu'il utilise a la structure suivante :

Num	utilisateur	Mot de passe
1	Administrateur	
2	Vente	
3	Planification	
4	GS	
5	GRH	

3.2.2.2. L'agent vente

Cette entreprise a une gamme variée de produit qu'elle met en disposition de ses clients et qui est divisé en trois groupes, nous avons mets ces produits dans des tables accompagnés par les prix ; donc les tables de produits sont comme décrites si dessous.

Une table de produit comporte six champs :

- **N° d'ordre** : contient un numéro d'ordre attribué au produit dans sa catégorie.
- **Désignation**: contient la désignation du produit qui est en une sorte une caractéristique qui le différencie des autres.
- **Prix H.T**: c'est le prix hors taxe.
- **T.V.A**: c'est une taxe sur le prix qui vaut 17% du prix hors taxe.
- **Prix T.T.C**: le prix tout taxe ; c'est le prix total du produit y compris la taxe.
- **NBR**: le nombre d'instances du produit existant en stock.

a) les buses en ciment HTS

N° d'ordre	Désignation	Prix H.T	T.V.A	Prix T.T.C	NBR	Délai
01	<i>Buses D 150x1mm</i>	<i>284,00</i>	<i>48,28</i>	<i>332,28</i>		
02	<i>Buses D 200x1mm</i>	<i>320,00</i>	<i>54,40</i>	<i>374,40</i>		
03	<i>Buses D 250x1mm</i>	<i>390,00</i>	<i>66,30</i>	<i>456,30</i>		
04	<i>Buses D 300x1mm</i>	<i>520,00</i>	<i>88,40</i>	<i>608,40</i>		
05	<i>Buses D 400x1mm</i>	<i>700,00</i>	<i>119,00</i>	<i>819,00</i>		
06	<i>Buses D 500x1mm</i>	<i>1140,00</i>	<i>193,80</i>	<i>1333,80</i>		
07	<i>Buses D 600x1mm</i>	<i>1400,00</i>	<i>238,00</i>	<i>1638,00</i>		
08	<i>Buses D 500x1mm A</i>	<i>1800,00</i>	<i>306,00</i>	<i>2106,00</i>		
09	<i>Buses D 600x1mm A</i>	<i>2100,00</i>	<i>357,00</i>	<i>2457,00</i>		
10	<i>Buses D 700x1mm A</i>	<i>2450,00</i>	<i>416,50</i>	<i>2866,50</i>		
11	<i>Buses D 800x1mm A</i>	<i>2950,00</i>	<i>501,50</i>	<i>3451,50</i>		
12	<i>Buses D 1000x1mm A</i>	<i>3900,00</i>	<i>663,00</i>	<i>4563,00</i>		
13	<i>Buses D 1200x1mm A</i>	<i>6200,00</i>	<i>1054,00</i>	<i>7254,00</i>		
14	<i>Buses D 1500x1mm A</i>	<i>8760,00</i>	<i>1489,20</i>	<i>10249,20</i>		

CHAPITRE IV : ETUDE DE CAS ET VALIDATION

b) les buses en ciment CPA

N° d'ordre	Désignation	Prix H.T	T.V.A	Prix T.T.C	NBR	Délai
<i>01</i>	<i>Buses D 150x1mm</i>	<i>248,00</i>	<i>42,16</i>	<i>290,16</i>		
<i>02</i>	<i>Buses D 200x1mm</i>	<i>289,00</i>	<i>49,13</i>	<i>338,13</i>		
<i>03</i>	<i>Buses D 250x1mm</i>	<i>330,00</i>	<i>56,10</i>	<i>386,10</i>		
<i>04</i>	<i>Buses D 300x1mm</i>	<i>400,00</i>	<i>68,00</i>	<i>468,00</i>		
<i>05</i>	<i>Buses D 400x1mm</i>	<i>650,00</i>	<i>110,50</i>	<i>760,50</i>		
<i>06</i>	<i>Buses D 500x1mm</i>	<i>960,00</i>	<i>163,20</i>	<i>1123,20</i>		
<i>07</i>	<i>Buses D 600x1mm</i>	<i>1100,00</i>	<i>187,00</i>	<i>1287,00</i>		
<i>08</i>	<i>Buses D 500x1mm A</i>	<i>1600,00</i>	<i>272,00</i>	<i>1872,00</i>		
<i>09</i>	<i>Buses D 600x1mm A</i>	<i>1800,00</i>	<i>306,00</i>	<i>2106,00</i>		
<i>10</i>	<i>Buses D 700x1mm A</i>	<i>2250,00</i>	<i>382,50</i>	<i>2632,50</i>		
<i>11</i>	<i>Buses D 800x1mm A</i>	<i>2750,00</i>	<i>467,50</i>	<i>3217,50</i>		
<i>12</i>	<i>Buses D 1000x1mm A</i>	<i>3500,00</i>	<i>595,00</i>	<i>4095,00</i>		
<i>13</i>	<i>Buses D 1200x1mm A</i>	<i>4500,00</i>	<i>765,00</i>	<i>5265,00</i>		
<i>14</i>	<i>Buses D 1500x1mm A</i>	<i>8000,00</i>	<i>1360,00</i>	<i>9360,00</i>		

c) les bordures en ciment CPA

N° d'ordre	Désignation	Prix H.T	T.V.A	Prix T.T.C	NBR	Délai
<i>01</i>	<i>Bordures de trottoirs 0,20/100</i>	<i>160,00</i>	<i>27,20</i>	<i>187,20</i>		
<i>02</i>	<i>Bordures de trottoirs 0,25/100</i>	<i>190,00</i>	<i>32,30</i>	<i>222,30</i>		
<i>03</i>	<i>Bordures de trottoirs 0,28/100</i>	<i>210,00</i>	<i>35,70</i>	<i>245,70</i>		

d) Agglomères en ciment CPA

N° d'ordre	Désignation	Prix H.T	T.V.A	Prix T.T.C	NBR	Délai
01	Parpaings 10/20/40 creux	18,00	3,06	21,06		
02	Parpaings 15/20/40 creux	29,00	4,93	33,93		
03	Parpaings 20/20/40 creux	31,00	5,27	36,27		
04	Parpaings 10/20/40 CPO	34,00	5,78	39,78		
05	Parpaings 15/20/30 Plein	31,00	5,27	36,27		
06	Parpaings 15/20/55	29,00	4,93	33,93		
07	Parpaings 20/20/55	34,00	5,78	39,78		

L'agent vente saisie **Prix H.T** entrée par l'utilisateur bien sure, et calcule les deux autre champs **T.V.A** et **Prix T.T.C**.

Le dernier champ est mis à jour après chaque vente; c'est la fonction de retrait qui s'en charge et qui est invoqué par l'agent GS suite à une requête envoyée par notre agent vente.

3.2.2.3. L'agent GRH

Cet agent comme nous l'avons expliqué dans le chapitre de modélisation comporte trois modules celui d'interface, plus celui de gestion du personnel et celui du paiement.

Le module de gestion du personnel comme nous l'avons montré auparavant utilise une table où il stock les informations sur les employés tel que le nom, le prénom et la situation familiale. D'autre part la table contient les champs suivants :

- **SITUATION:** c'est la situation familiale de l'employé.
- **Poste :** présente le poste occupé par l'employé, CPT dans le cas d'un comptable par exemple.
- **Pay.un:** c'est le paiement unitaire ce qui veut dire par jour
- **NB- JRS:** ce champ contient le nombre de jours travaillés par l'employé par mois.
- **NB-HRS:** c'est le nombre d'heurs travaillés par l'employé par mois.

CHAPITRE IV : ETUDE DE CAS ET VALIDATION

N°	Nom	Prénom	SITUATION	Poste	Pay.un	NB- JRS	NB-HRS
1	Terghini	Rabiaa	célibataire	CPT	710,00	22,00	173,33
10	Mansour	Salim	marié	M.O	450,64	17,00	133,94
11	Mihoubi	Rahmoune	marié	M.O	450,64	17,00	133,94
12	Kahloul	Ahmed	marié	M.O	450,64	17,00	133,94
13	Brahimi	Mohamed	marié	M.O	600,00	22,00	173,33
14	Mansouri	Mosbah	marié	M.O	450,64	19,00	149,69
18	Dhiafi	Tahar	célibataire	SOUD.	550,00	22,00	173,33
19	Mansour	Rachid	marié	C.PARC	600,00	22,00	173,33
43	Brahimi	Khaled	marié	M.O	450,64	15,00	118,18
51	Touti	Amor	marié	GARD	450,81	22,00	173,33
55	Djeddou	Rachid	marié	Chauf.	605,00	22,00	173,33
63	Hassani	Raouf	célibataire	C.AT	600,00	22,00	173,33
66	Soufa	Adel	célibataire	M.O	450,80	2,00	15,76
68	Ghlam	Madani	marié	M.O	450,80	22,00	173,33
73	Athamnia	Mohamed	célibataire	M.O	450,80	11,00	86,67
74	Kahoul	lotfi	célibataire	M.O	450,80	12,00	94,54
79	Sebaa	Yaakoub	célibataire	CLRC	450,80	17,00	133,94
85	Dhaouadi	Lazhar	célibataire	M.O	450,80	18,00	141,82
87	Saadi	Mourad	marié	GARD	450,81	22,00	173,33
89	Morghad	Achour	marié	SOUD.	550,00	22,00	173,33
90	Ouannas	Brahim	célibataire	MCANIC	550,00	22,00	173,33
92	Azri	Belkacem	marié	M.O	450,80	22,00	173,33
94	Ellak	Antar	marié	M.O	450,80	8,00	63,03
97	Hassani	Leila	marié	Ad.Bur.	500,00	22,00	173,33

Le module du payement est responsable évidemment du calcul de la paye incluant les primes. Dans le calcul de la paye de l'entreprise "*Technique sud construction*" entrent plusieurs facteurs, par conséquent l'agent GRH fait appel à deux tables contenant plusieurs champs qui interviennent dans le calcul de la paye et qui sont :

- **SAL.BASE** : c'est le salaire de base, se calcul par la multiplication du payement unitaire par le nombre de jours travaillés par l'employé par mois.
- **PRI** : une prime individuelle fixé par l'utilisateur donc l'agent n'a pas a la calculée il va seulement la saisir.
- **IEP** : indemnité d'expérience professionnelle qui est une prime d'expérience.
- **PZ** : prime de zone qui fait 21% du salaire de base.
- **F/M** : frais de mission.
- **IFSP/AST** : indemnité forfaitaire service permanent
- **SAL.POST** : salaire de poste qui vaut la somme des champs déjà cités.
- **P.PANIER** : prime de panier donné dans le cas où l'employé ne prend pas de pose déjeuné pendant la journée.
- **B.IRG** : base IRG qui est une taxe sur la paye et qui vaut la somme du salaire de poste et la prime de panier.
- **SAL.BRUT** : salaire brut égal à la somme de la **B.IRG** et les frais de mission.
- **ABAT** : l'abattement dans le cas d'un célibataire est égal à 450 DA sinon 300 DA.
- **RET_IRG** : le retrait IRG est une somme à retirer dans le cas où le salaire brut dépasse 1500 DA.
- **NET_PAY** : le net à payer ce calcul comme suit :

$$\text{NET_PAY} = \text{SAL.BRUT} - \text{CSS} - \text{RET_IRG}$$

Et par la suite nous présentons la structure des deux tables sur les quelle ces champs sont répartis et qui sont la table de primes et celle de payement, accompagnée d'un échantillon d'informations de paye pour des employés de l'entreprise "*Technique sud construction*".

▪ **Table des primes**

N	PRI	IEP	PZ	F/M	IFSP/AST	P. PANIER
1	1562,00	1405,80	3280,20	0,00	427,95	0,00
10	0,00	766,09	1608,78	0,00	0,00	1159,40
11	0,00	919,31	1608,78	0,00	0,00	1159,40
12	0,00	995,91	1608,78	0,00	0,00	1159,40
13	0,00	924,00	2772,00	1124,24	0,00	1500,40
14	0,00	770,59	1798,05	0,00	0,00	1295,80
18	1210,00	847,00	2541,00	1304,42	0,00	1500,40
19	1320,00	1518,00	2772,00	6502,62	0,00	1500,40
43	0,00	47,32	1419,52	0,00	0,00	1023,00
51	0,00	198,36	2082,74	589,00	0,00	1500,40
55	0,00	133,10	2795,10	1722,84	0,00	1500,40
63	0,00	0,00	2772,00	3965,08	0,00	1500,40
66	0,00	9,02	189,34	0,00	0,00	136,40
68	0,00	495,88	2082,70	0,00	0,00	1500,40
73	0,00	49,59	1041,35	0,00	0,00	750,20
74	0,00	54,10	1136,02	0,00	0,00	818,40
79	0,00	0,00	1609,36	0,00	0,00	1159,40
85	0,00	0,00	1704,02	0,00	0,00	1227,60
87	0,00	0,00	2082,74	589,00	0,00	1500,40
89	1210,00	0,00	2541,00	4075,19	0,00	1500,40
90	1210,00	0,00	2541,00	2075,19	0,00	1500,40
92	0,00	0,00	2082,70	0,00	0,00	1500,40
94	360,64	0,00	757,34	0,00	0,00	545,60
97	0,00	0,00	2310,00	2887,90	0,00	0,00

CHAPITRE IV : ETUDE DE CAS ET VALIDATION

▪ **Table de paye**

N	SAL. POST	B.IRG	SAL.BRUT	ABAT	RET_IRG	NET_PAY
1	22295,95	22295,95	22295,95	450,00	156,00	20133,31
10	10035,75	11195,15	11195,15	300,00	0,00	10291,94
11	10188,97	11348,37	11348,37	300,00	0,00	10431,36
12	10265,58	11424,98	11424,98	300,00	0,00	10501,08
13	16896,00	18396,40	19520,64	300,00	0,00	18000,00
14	11130,81	12426,61	12426,61	300,00	0,00	11424,84
18	16698,00	18198,40	19502,82	450,00	0,00	18000,00
19	18810,00	20310,40	26813,02	300,00	0,00	25120,12
43	8226,43	9249,43	9249,43	300,00	0,00	8509,05
51	12198,92	13699,32	14288,32	300,00	0,00	13190,42
55	16238,20	17738,60	19461,44	300,00	0,00	18000,00
63	15972,00	17472,40	21437,48	450,00	0,00	20000,00
66	1099,95	1236,35	1236,35	450,00	0,00	1137,36
68	12496,18	13996,58	13996,58	300,00	0,00	12871,92
73	6049,74	6799,94	6799,94	450,00	0,00	6255,46
74	6599,71	7418,11	7418,11	450,00	0,00	6824,14
79	9272,96	10432,36	10432,36	450,00	0,00	9597,79
85	9818,42	11046,02	11046,02	450,00	0,00	10162,37
87	12000,56	14089,96	14089,96	300,00	0,00	13009,91
89	15851,00	21426,59	21426,59	300,00	0,00	20000,00
90	15851,00	19426,59	19426,59	450,00	0,00	18000,00
92	12000,30	13500,70	13500,70	300,00	0,00	12420,67
94	4724,38	5269,98	5269,98	300,00	0,00	4844,79
97	13310,00	16197,90	16197,90	300,00	0,00	1500,00

3.2.2.4. L'agent GS :

Comme tout autres entreprise ; celle que nous étudions possède un magasin où elle stocke ses produits; donc elle nécessite un mécanisme de gestion pour contrôler la circulation de la marchandise ; l'agent GS rempli cette tâche en mettant à jour la base de données dans le cas de retrait ou celui d'entrée de produit au magasin, pour ce fait il utilise deux tables l'une pour noter les produits consommés au cours du jour et l'autre table pour noter les produits fabriqués par l'entreprise et qui doivent être stockés dans le magasin:

➤ **La table de consommation:** contient les champs suivants

- **Désignation** : nom du produit accompagné par les caractéristiques qui lui distingue.
- **Qté Ant:** quantité antérieure c'est la quantité du produit déjà consommée.
- **Qté J:** quantité du produit consommée dans la journée courante.
- **Mont:** la somme d'argent équivalente aux produits consommés dans la journée courante.
- **Cum, Qté:** la quantité totale de produit consommés qui vaut la somme de la quantité antérieure et celle de la journée.
- **Mont Cum:** somme d'argent équivalente la quantité totale de produits consommés.

Désignation	Qté Ant	Qté J	Mont	Cum, Qté	Mont Cum
Ciment CPA (Sac)	1177.4	33.00	11055.00	1210.40	405484.00
Ciment HTS (Sac)	60.5	0.00	0.00	60.50	26620.00
Sable rivière (m ³)	92.29	5.57	1531.75	97.86	26911.50
Sable Concasser (m ³)	41.33	2.1	1312.50	43.43	27143.75
Gravier 3/7(m ³)	97.3	6.44	8855.00	103.74	66134.25
Gravier 7/15(m ³)	24.08	2.11	29012.50	26.19	16696.13
Acier (kg)	3285.68	114	6384.00	3399.68	190382.08
Total			58150.75		759371.71

➤ **La table de production :** contient les champs suivants

- **Désignation :** nom du produit accompagné par les caractéristiques qui lui distingue.
- **Qté Ant:** quantité antérieure c'est la quantité du produit existante dans le magasin.
- **Qté J:** quantité du produit produite dans la journée courante.
- **Mont:** la somme d'argent équivalente à la production de la journée courante.
- **Cum, Qté:** la quantité totale de produit existant qui vaut la somme de la quantité antérieure et celle de la journée.
- **Mont Cum:** somme d'argent équivalente la quantité totale de produit existant en stock.

Désignation	Qté Ant	Qté J	Mont	Cum, Qté	Mont Cum
Bordures 025	3442	169	32110.00	3611	649980.00
Bordures 028	903	134	28140.00	1037	186660.00
Buses dim 800 A CPA	50	20	51000.00	70	178500.00
Buses dim 800 A HTS	25	0.00	0.00	25	70000.00
Buses dim 600 A CPA	51	0.00	0.00	51	91800.00
Buses dim 1000 A CPA	269	20	71000.00	289	10259500.00
Buses dim 500 A CPA	108	0.00	0.00	108	172800.00
Buses dim 1000 A HTS	13	0.00	0.00	13	50700.00
HOIRDIS 15/55	748	0.00	0.00	748	20944.00
Total			182250.00		11680884.00

Notant bien que :

Montant = Montant unitaire x Quantité par jour

Quantité cumulée = Quantité antérieure + Quantité par jour

Montant cumulé = Montant unitaire x Quantité cumulée

Aussi la fonction de retrait ainsi que celle d'entrée en stock met à jour le champ nombre appartenant aux tables de produit déjà présentées dans la partie de l'agent vente. D'autre part la fonction ajout nouveau matériel accède à ces tables afin d'insérer des enregistrements supplémentaires propres aux nouveaux matériels entrant en stock.

3.2.2.5. L'agent planification

L'agent planification rassemble quelques fonctionnalités nécessaires pour le contrôle d'activité de l'entreprise pour "*Technique sud construction*" ces fonctionnalités sont les suivantes:

- Le calcul et la conservation des frais de production sont l'une des fonctionnalités indispensables pour cette entreprise, dans le but est de garder le contrôle sur les revenus et ainsi maintenir l'équilibre économique de l'entreprise, cette fonctionnalité utilise la table décrite ici :

	ANTECEDENT	JOURNEE	CUMUL
Main d'œuvre	261660.86	24675.68	388300.00
Personnel indirect	128790.00	4910.00	133700.00
Personnel à la tâche	124890.00	9090.00	133980.00
Charges patronales	138211.61	12475.05	150686.66
Usage matériel	618808.32	23692.50	642500.82
Taxe foncière et assurances	3340.00	167.00	3507.00
Eau et électricité	26600.00	1330.00	27930.00
Administration	18000.00	900.00	18900.00
Carburant	6000.00	300.00	6300.00
perte	19210.00	3690.00	22900.00
Total	1345510.79	81230.23	1426741.02

Donc cette table contient les différentes catégories de dépenses de l'entreprise, et leurs montant pendant le jour et sont précédent aussi le cumul, aussi l'agent planification va calculer la somme des dépenses par jour.

- De plus l'entreprise a besoin de calculer le montant des dépenses sur la production, et de savoir ses bénéfices et le montant de gain ajouté à sa caisse ce qui se récapitule et se sauvegarde dans la table suivante:

	ANTECEDENT	JOURNEE	CUMUL
PRODUCTION	2235694.00	182250.00	2417944.00
COUT DE PRODUCTION	2091993.90	139380.98	2231374.88
BENEFICE	143700.10	42869.02	186569.12
VENTE	2381194.00	497948.00	2879142.00
RESULTAT	289200.10	358567.02	647767.12

Et voici la description des champs de la table:

- **PRODUCTION:** c'est la somme d'argent que doit apporter la vente de l'ensemble des produits.
- **COUT DE PRODUCTION:** c'est la somme d'argent dépenser afin d'accomplir le processus de production.
- **BENEFICE :** vaut la soustraction du **COUT DE PRODUCTION** depuis la **PRODUCTION**.
- **VENTE:** c'est la somme d'argent récolté par les produits vendus.
- **RESULTAT :** représente le gain et se calcule avec la soustraction du **COUT DE PRODUCTION** du contenu du champ **VENTE**.

Dans cette partie nous avons expliqué le fonctionnement propre à l'entreprise "*Technique sud construction*" pour chaque agent, accompagné de la structure de la base de données qui est centralisé. Ensuite nous allons décrire l'environnement dans lequel nous avons implémenté le logiciel ainsi que la manière dont il est conçu.

4. Résultat

Voici le résultat de la validation de l'exemple étudié, nous décrivons les interfaces principales de chaque agent et présentons quelques fonctions importantes du code implémenté.

4.1. L'agent AC

La fenêtre ci-dessous est la fenêtre principale de l'agent AC et donc c'est la première qui apparaît lors du lancement du système, depuis cette fenêtre le reste des agents peuvent être activés à travers des boutons.

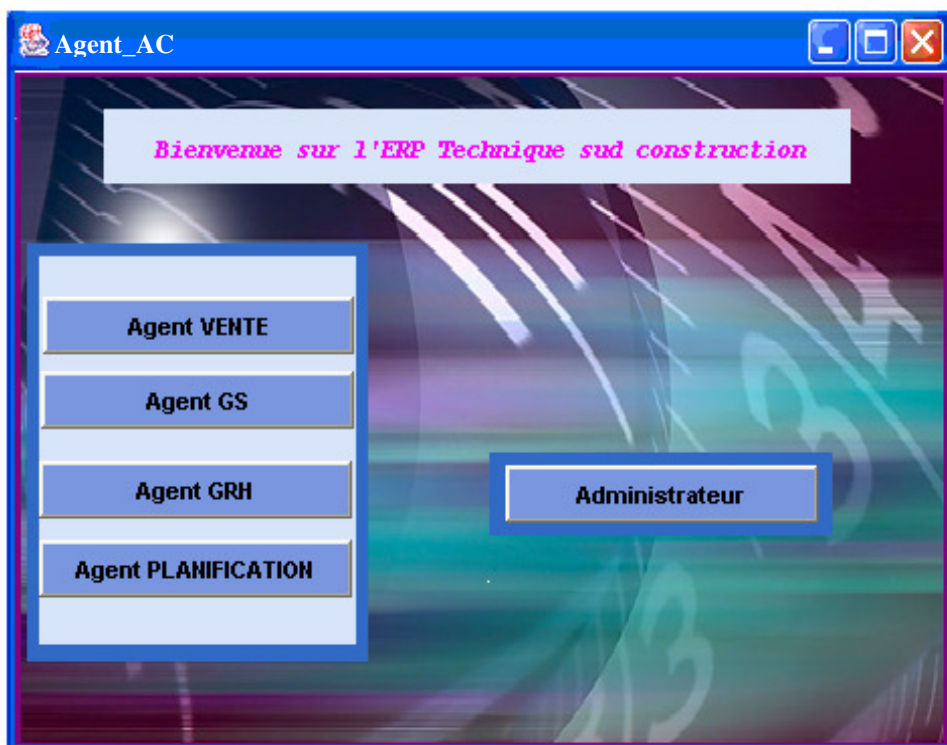


Figure 1 : fenêtre principale de l'agent AC

Comme nous l'avons déjà présenté dans le chapitre de modélisation, l'accès aux comptes est protégé, et donc nécessite une identification ce qui est illustré dans la fenêtre suivante.

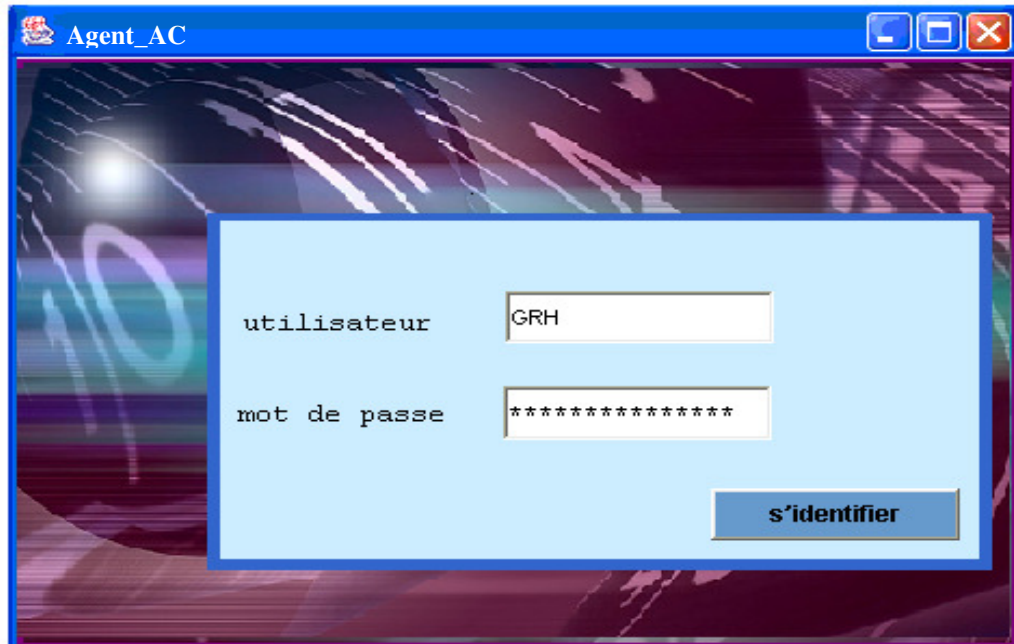


Figure 2 : fenêtre d'identification

Suite à cette identification l'accès sera permis soit à l'un des agents du système ou bien au compte administrateur géré par l'agent AC, et qui fournit les fonctions présentes dans cette fenêtre :



Figure 3 : fenêtre du compte administrateur

4.2. L'agent vente

Le module interface de l'agent vente fournit un ensemble de fenêtres chacune pour une tâche spécifique ; celle présentée dans cette partie est la fenêtre principale qui permet l'accès aux autres.

Comme nous pouvons le constater, cette fenêtre contient deux parties ; la première permettant la saisie de la commande et son lancement, et la deuxième est dédiée à la consultation de la base de données.

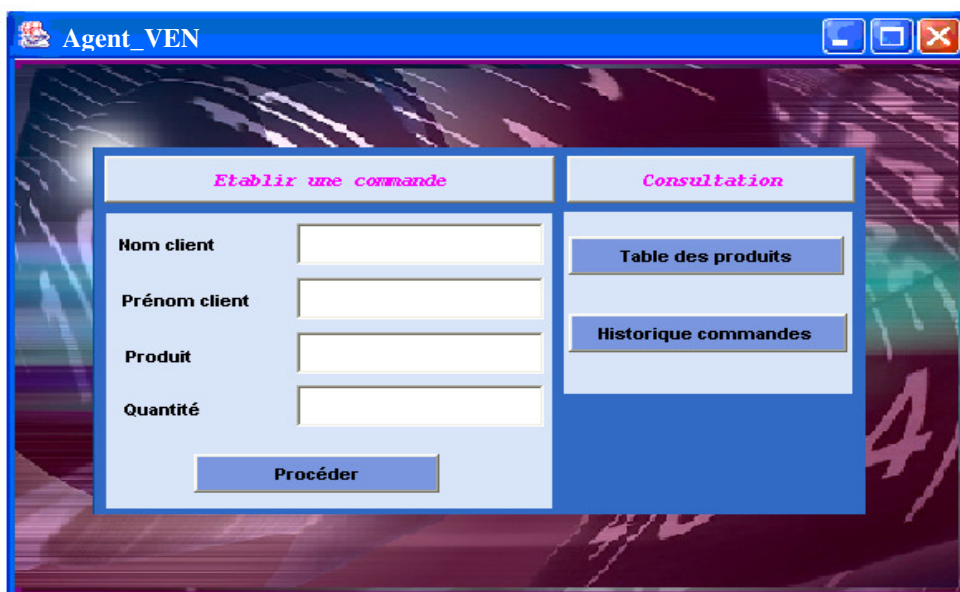


Figure 4 : fenêtre principale de l'agent VEN

4.3. L'agent GRH

La fenêtre principale de l'agent GRH contient deux boutons essentiels, l'un pour appeler la fonction de gestion du personnel et l'autre pour effectuer le processus de paiement.

D'autre part elle contient un ensemble de boutons pour la consultation de la base de données.

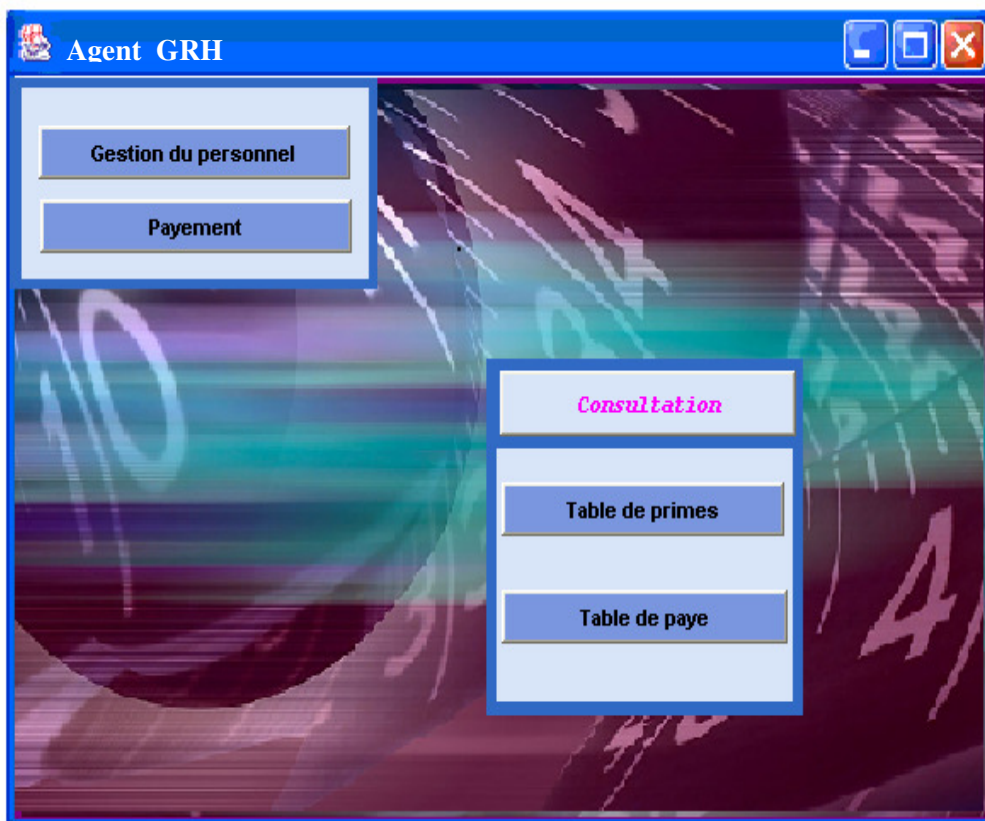


Figure 5 : fenêtre principale de l'agent GRH

4.4. L'agent GS

L'interface de l'agent GS contient une partie pour la saisie des informations du produit qui va être soit retiré ou entré au magasin, de plus aux boutons permettant l'accès aux fonctions manipulant la base de données et ainsi ajouter ou supprimer un enregistrement d'un produit.

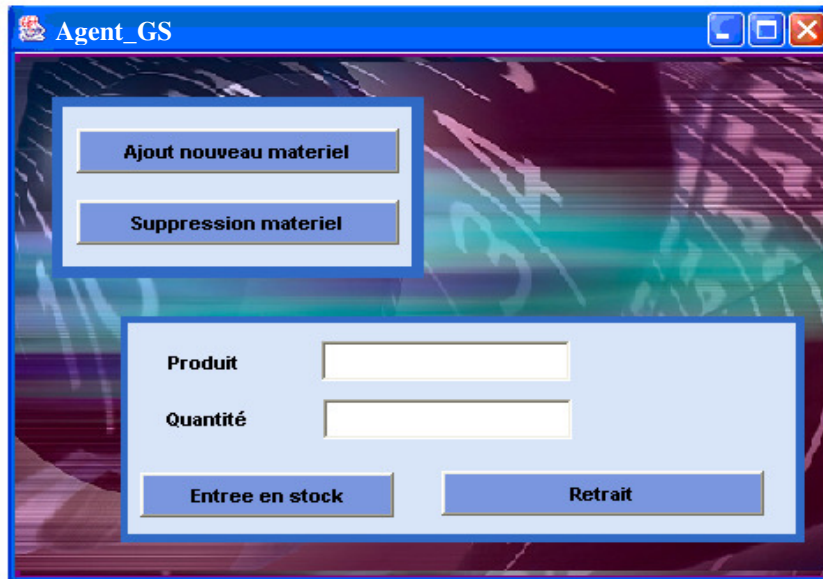


Figure 6 : fenêtre principale de l'agent GS

Dans ce qui suit la fenêtre qui apparaît suite à l'appuie sur le bouton «ajout nouveau matériel», elle contient les champs d'un enregistrement du produit à insérer dans la table de produits.

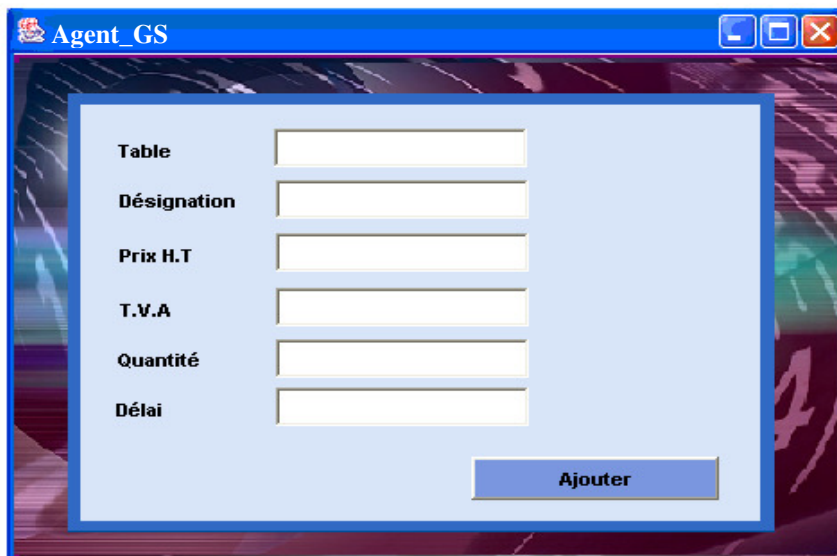


Figure 7 : fenêtre d'ajout d'un nouveau matériel

Et celle ci apparaît suite à l'appuie sur le bouton «suppression matériel», elle permet la saisie des informations nécessaire à l'opération de suppression.

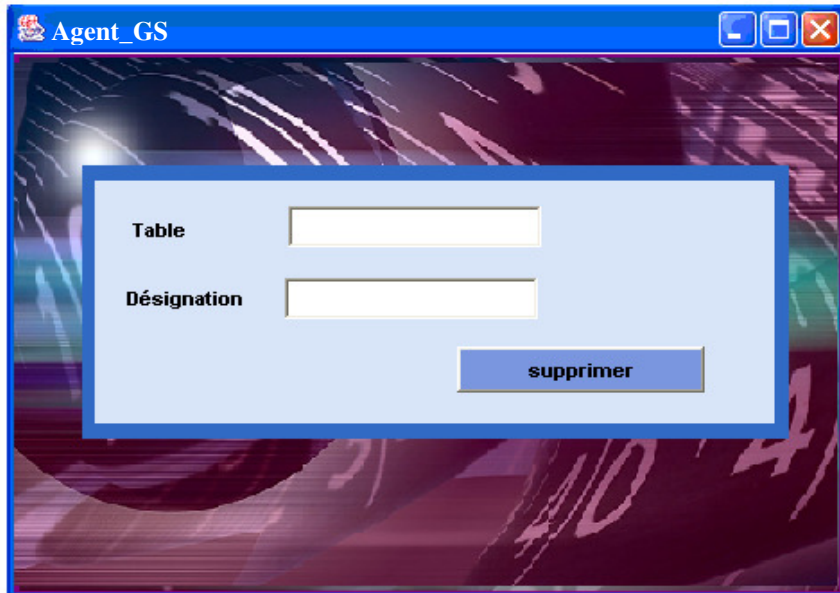


Figure 8 : fenêtre de suppression d'un matériel

4.5. L'agent planification

L'interface de l'agent planification commence par une fenêtre contenant trois boutons :

- Le premier pour lancer la fonction qui calcule la consommation par jour.
- Le deuxième pour lancer la fonction qui calcule la production par jour.
- Le troisième pour lancer la fonction qui calcule délais de production.

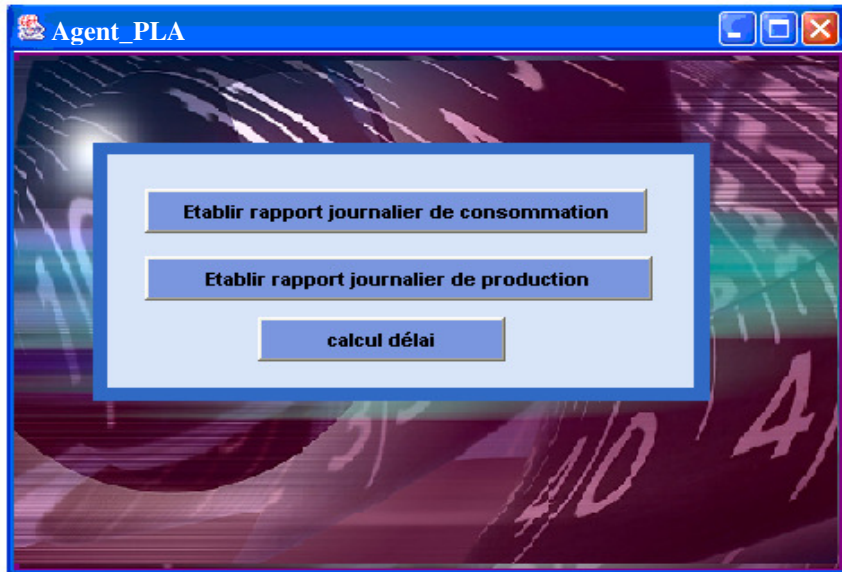


Figure 9 : fenêtre principale de l'agent PLA

Suite à l'appui sur le troisième bouton, la fenêtre ci après apparaît, elle a pour rôle la saisie d'informations nécessaires pour l'exécution de la fonction calculant le délai de fabrication.

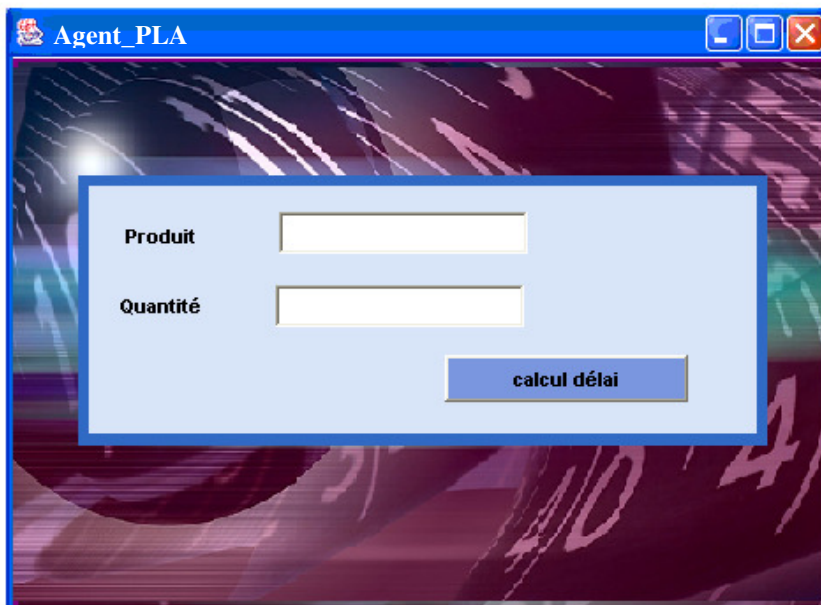


Figure 10: fenêtre du calcul du délai

Nous passons à la présentation de la structure générale du code associé à l'agent accueil, en mettant l'accent sur ces fonctionnalités.

```
package ERP;
import jade.core.Agent;
import jade.core.Runtime;
import jade.core.ProfileImpl;
import jade.core.behaviours.CyclicBehaviour;
import jade.domain.DFService;
import jade.domain.FIPAException;
import jade.domain.FIPAAgentManagement.DFAgentDescription;
import jade.lang.acl.ACLMessage;
import jade.lang.acl.MessageTemplate;
import jade.lang.acl.UnreadableException;
import javax.swing.JFrame;
import javax.swing.JPanel;
import java.awt.BorderLayout;
import java.awt.Toolkit;
import javax.swing.JButton;
import javax.swing.JLabel;

public class AcAgent extends Agent {
private JFrame jFrame = null;
.
.
.
// Initialisation de l'agent
protected void setup()
{
    getJFrame().setVisible(true);
    try {
// Création de description de l'agent
        DFAgentDescription dfd = new DFAgentDescription();
        dfd.setName(getAID());
    }
}
```

```
    DFService.register(this, dfd);
    }
    catch (FIPAException e) {
    e.printStackTrace();}

protected void takeDown() {
try {
// Suppression de l'agent du DF
DFService.deregister(this);
} catch (FIPAException e) {
e.printStackTrace();}
}
//ce bouton sert à créer l'agent Vente
jButton1.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent e) {
try {

// Récupération du Main Container en cours d'exécution
Runtime rt = Runtime.instance();

// Création du profil par défaut
ProfileImpl p = new ProfileImpl(false);
AgentContainer container = rt.createAgentContainer(p);
AgentController Agent=null;

// Création de l'agent Vente
Agent = container.createNewAgent("VEN", "ERP.VEN", null);
Agent.start();
} catch (Exception any) {
any.printStackTrace();}
}
```

```
}
```

//ce bouton sert à créer l'agent GS

```
JButton2.addActionListener(new java.awt.event.ActionListener() {  
public void actionPerformed(java.awt.event.ActionEvent e) {  
try {
```

// Récupération du Main Container en cours d'exécution

```
Runtime rt = Runtime.instance();
```

// Création du profil par défaut

```
ProfileImpl p = new ProfileImpl(false);  
AgentContainer container = rt.createAgentContainer(p);  
AgentController Agent=null;
```

// Création de l'agent GS

```
Agent = container.createNewAgent("GS", "ERP.GS", null);  
Agent.start();  
} catch (Exception any) {  
any.printStackTrace();}  
}}
```

// Même chose pour le reste des agents

```
.  
. .  
.
```

// Activation du compte Administrateur

```
JButton5.addActionListener(new java.awt.event.ActionListener() {  
public void actionPerformed(java.awt.event.ActionEvent e) {  
try {  
jFrame jFrame = new JFrame AD;  
jFrame.show();  
} catch (Exception any) {  
any.printStackTrace();}  
}}
```

-
-
-

// la classe principale

```
public static void main(String[] args) {  
try {  
•  
•  
•  
  
Runtime rt = Runtime.instance();  
ProfileImpl p = new ProfileImpl(false);  
AgentContainer container = rt.createAgentContainer(p);  
AgentController Agent=null;
```

// Création de l'agent AC

```
Agent = container.createNewAgent("AC", " ERP. AC ", null);  
Agent.start();  
} catch (Exception any) {  
any.printStackTrace();}  
•  
•  
•  
}}
```

Nous utilisons aussi une autre classe d'agents pour instancier les agents de notre système qui est la classe d'agents Gui représentant des agents avec interface. L'agent GS hérite de cette classe ; donc il a la forme suivante :

```
public class GSAgent extends GuiAgent {  
    private GSAgentGui myGui;  
    // Initialisation de l'agent  
    protected void setup()  
    {  
    // Création de la Gui  
    myGui = new GSAgentGui(this);  
    myGui.show();
```

// Création de description de l'agent

```
DFAgentDescription dfd = new DFAgentDescription();
dfd.setName(getAID());
DFService.register(this, dfd);
    }
catch (FIPAException e) {
e.printStackTrace();}
protected void takeDown() {
myGui.dispose();
```

// Suppression de l'agent du DF

```
try {
DFService.deregister(this);
} catch (FIPAException e) {
e.printStackTrace();}
}
.
.
.
}
```

De plus, l'activité principale dans tout système multi agent est la communication, et d'après ce que nous avons expliqué auparavant l'agent VENTE communique avec les deux agents GS et GRH, nous allons illustrer l'échange de message de notre système en prenant comme exemple la partie suivante du code de l'agent VENTE.

```
package ERP.GS;
import java.io.IOException;
import jade.core.Agent;
import jade.core.AID;
import jade.domain.FIPAAgentManagement.*;
import jade.domain.DFService;
import jade.domain.FIPAException;
import jade.core.behaviours.CyclicBehaviour;
import jade.lang.acl.ACLMessage;
```

```
import jade.lang.acl.MessageTemplate;
import jade.lang.acl.UnreadableException;

public class VENAgent extends GuiAgent {
    .
    .
    .
    // déclaration de l'objet à envoyer dans le message
    private Object[] obj=null;
    string designation;
    float quantité ;
    .
    .
    .

    protected void setup() {
    try {
    DFAgentDescription dfd = new DFAgentDescription();
    dfd.setName(getAID());
    DFService.register(this, dfd);
    } catch (FIPAException e) {
    e.printStackTrace();
    }

    // Agent Vente envoi une requête de retrait à l'agent GS
    // donc le message doit contenir le nom du produit vendu ainsi que sa quantité
    Object[] obj={"désignation","quantité"} ;
    // Préparation du message
    ACLMessage msg1 = new ACLMessage(ACLMessage.INFORM);
    try {
    msg1.setContentObject(obj);
    msg1.addReceiver(new AID("GS", AID.ISLOCALNAME));
    send(msg1);
    }
}
```

```
catch (IOException e) {
```

```
e.printStackTrace();
```

```
}
```

```
} catch (UnreadableException e) {
```

```
e.printStackTrace();}
```

```
}}}
```

// Reception de la réponse de l'agent GS

```
addBehaviour(new CyclicBehaviour(this) {
```

```
public void action() {
```

// Attente de la réponse de l'agent GS

```
ACLMessage msg =receive(MessageTemplate.MatchPerformative(ACLMessage.INFORM));
```

```
if (msg == null) {
```

// Blocage en attente du message

```
block();
```

```
}
```

```
protected void takeDown() {
```

```
try {
```

```
DFService.deregister(this);
```

```
} catch (FIPAException e) {
```

```
e.printStackTrace();
```

```
}}
```

```
.
```

```
.
```

```
.}
```

De son coté l'agent GS répond par une affirmation après qu'il effectue le retrait, pour ce fait il fait appel au code suivant :

// Création de la réponse

```
ACLMessage reply = msg.createReply();
```

```
reply.setContent("OK");
```

// Envoi la réponse

```
myAgent.send(reply);
```


5. Exemple illustratif

Cette partie est consacrée à l'illustration du déroulement d'une commande ; la première étape consiste à remplir le formulaire de la manière expliquée dans la figure suivante et qui est la fenêtre principale de l'agent vente.

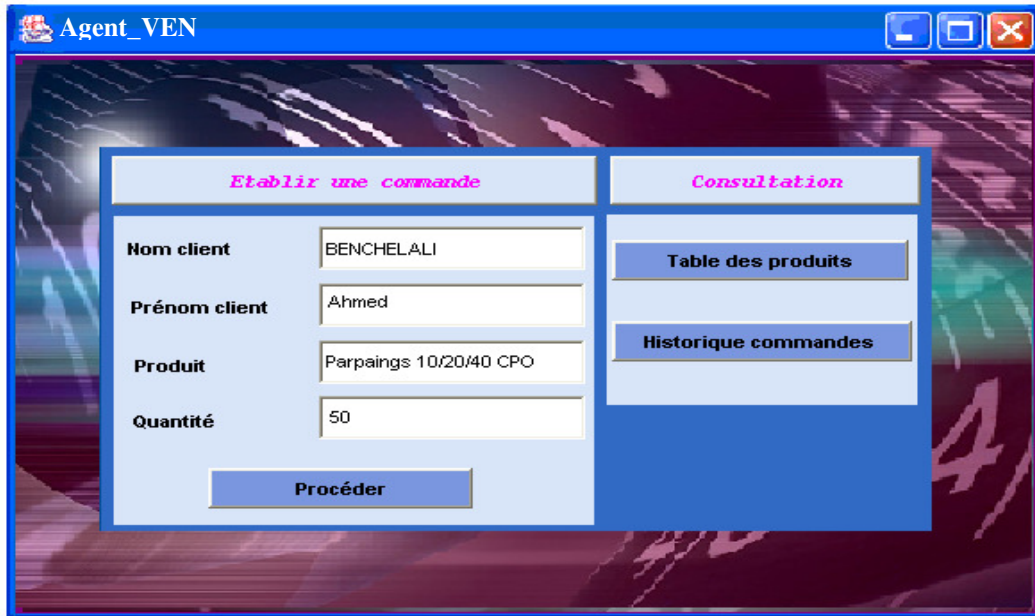


Figure 11: fenêtre principal de l'agent vente

Après le remplissage du formulaire et l'appuie sur le bouton « **Procéder** », l'agent vente envoie une requête de retrait à l'agent de gestion du stock ce qui fait apparaître une fenêtre sur laquelle l'échange de message se visualise.

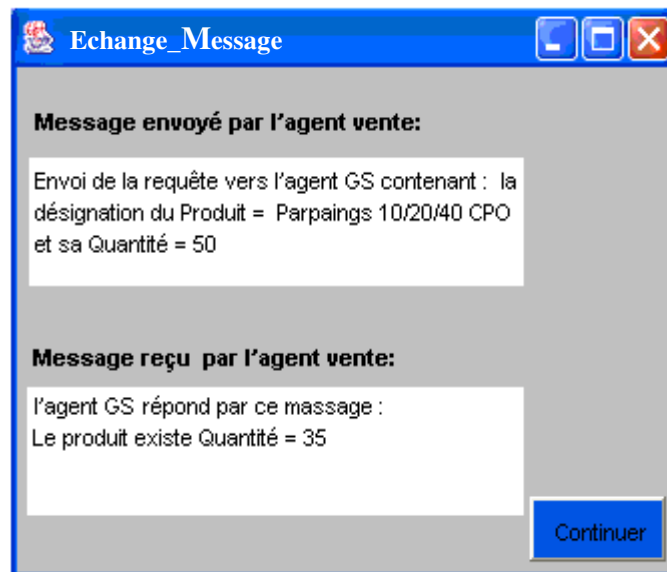


Figure 12: fenêtre d'échange de message entre l'agent vente et l'agent GS

Suite au clic sur le bouton « **Continuer** », l'agent vente poursuit son processus, dans ce cas le client a demandé une quantité de produit égale à 50 mais l'agent GS n'a trouvé en stock que 35, suite à cette réponse l'agent vente envoie une requête à l'agent planification afin de connaître le délai dans le quelle la quantité restante sera prête ; ce qui est illustré dans cette figure :

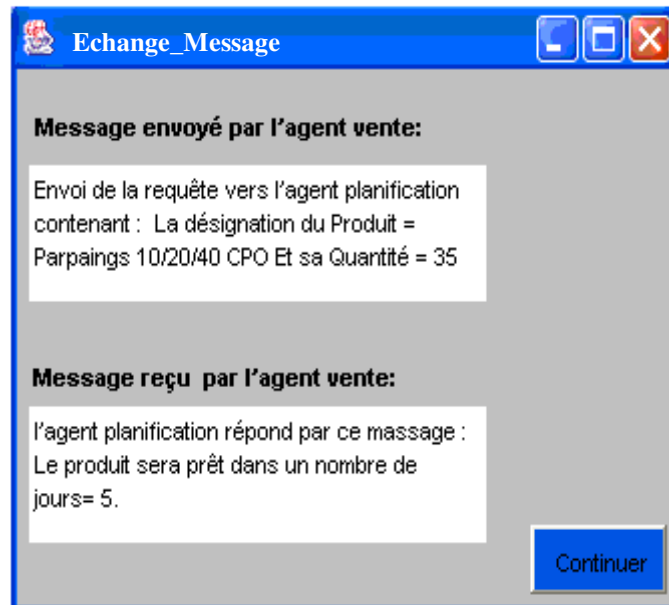


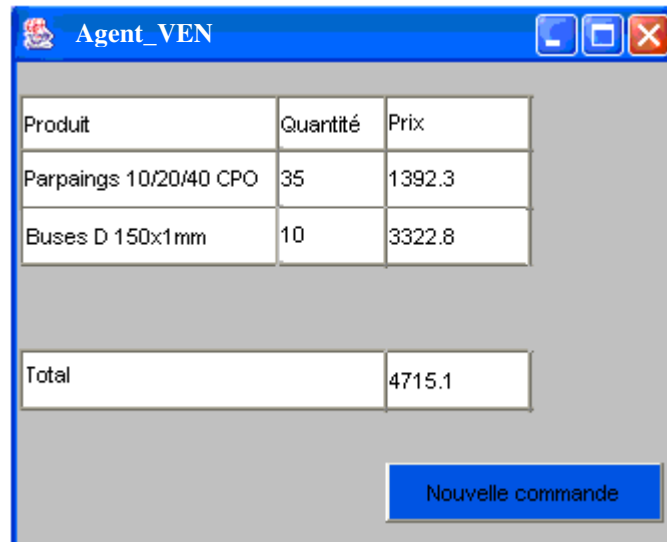
Figure 13: fenêtre d'échange de message entre l'agent vente et l'agent planification

Dès que l'utilisateur appui sur le bouton « **Continuer** », une fenêtre apparaît lui donnant le choix soit de faire une nouvelle commande ou bien d'établir la facture du client.



Figure 14: fenêtre intermédiaire pour le choix

Si l'utilisateur choisi le premier bouton il retourne à la fenêtre principale de l'agent vente, et s'il choisi le deuxième l'agent vente calcul la facture du client et l'affiche comme suit :



Produit	Quantité	Prix
Parpaings 10/20/40 CPO	35	1392.3
Buses D 150x1mm	10	3322.8
Total		4715.1

Nouvelle commande

Figure 15: fenêtre de la facture

De cette façon une commande est achevée, l'appui sur le bouton «**Nouvelle commande**» va engendrer le retour vers la fenêtre principale de l'agent vente et donc prendre une nouvelle commande.

6. L'agent sniffer

Nous faisons appel dans cette partie à l'agent sniffer qui est un outil propre à la plateforme JADE, et qui sert à visualiser les échanges de messages entre les agents du système. Dans le cas de notre système l'agent sniffer a donné le résultat suivant :

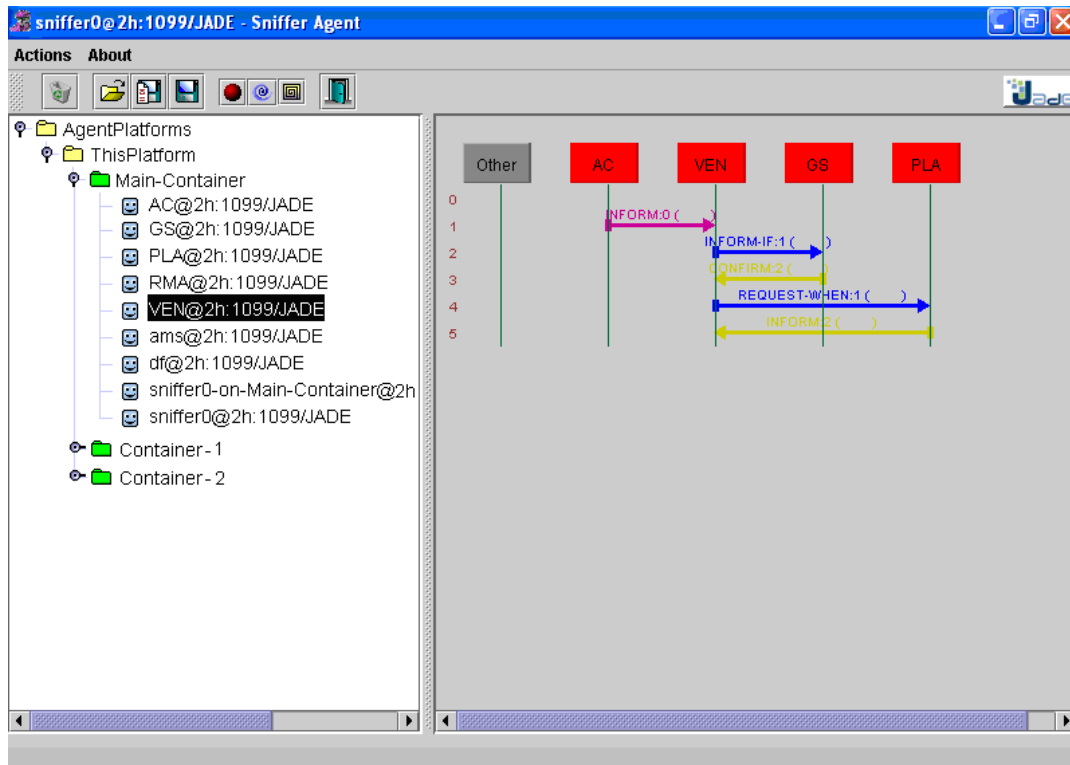


Figure 16: L'agent sniffer

Les flèches colorées représentent les messages envoyées ; leurs directions ainsi que leurs types. Dans ce cas l'agent sniffer éclairci les échanges de messages dans le cas d'une commande comme nous l'avons déjà expliqué auparavant.

7. Conclusion

D'après l'étude de cas que nous avons effectuée, nous avons pu constater le gigantesque taux de connaissance que doit manipuler une entreprise, et par conséquent l'importance d'un système ERP à la fois centralisé et basé connaissance. Donc l'approche projetée semble garantir la cohérence de l'information et sa fiabilité, et une manipulation de cette dernière qui est assez souple. Or nous ne prétendons pas que notre solution est complète mais plutôt robuste et apporte de bons résultats. Aussi nos recherches dans ce domaine nous ont permis d'avoir quelques idées de recherche supplémentaires dans le même cadre que nous allons exposer dans la conclusion générale ayant l'ambition de pouvoir les conquérir.



***CONCLUSION
GENERALE***

Au cour de notre recherche nous avons découvert le concept d'ERP qui n'est pas seulement un outil, mais un véritable concept global de management, qui intègre les méthodologies et les procédures propres a l'entreprise, et permet de lier des systèmes d'information disjoints comme la relation clientèle, et interfaces comptables.

De par son objectif, l'ERP est amené à être modifié périodiquement pour prendre en compte toutes les évolutions et les nouvelles orientations de l'entreprise. D'où l'importance d'un système flexible qui ne court pas de risque lors du changement, et nous avons éclairci que les systèmes multi agent sont une bonne solution pour relevé se défi .

Dans ce but nous avons présenté un modèle qui incarne une approche d'intégration d'agents dans les ERP, tout en argumentant notre opinion.

L'étude du cas de l'entreprise Technique sud construction nous a permis l'exploitation de notre approche ; et sa validation à travers un système ERP composé de cinq agents, qui remplissent des fonctions parmi les plus nécessaire pour une entreprise. Ce système est réalisé dans un environnement assez performant pour la modélisation de systèmes multi agent qui est la plateforme JADE.

Nos perspectives dans ce domaine de recherches sont les suivants :

- ✓ Rendre notre système plus distribué en introduisant des agents mobiles qui suivent les produits le long de leurs processus de fabrication.
- ✓ Vu que nous nous basant plus sur l'importance de la connaissance, nous avons songé à apporter plus d'intelligence à notre système en introduisant un agent pour la prise de décision
- ✓ Et pourquoi pas augmenter le niveau de raisonnement de nos agents en utilisant par exemples un raisonnement à base de cas

D'après les points que nous avons cités, nous pouvons juger le domaine des systèmes ERP comme un domaine apte à l'évolution et riche en sujets de recherche qui attend, celui qui vient pour relever ses défis.

Références

REFERENCES

- [1] Conception Et Au Pilotage D'une Entreprise Réseau", 3e Conférence Francophone de MOdélisation et SIMulation «Conception, Analyse et Gestion des Systèmes Industriels» MOSIM'01, Troyes (France), 2001.
- [2] S. OURARI, B. BOUZOUIA, "Approches et Outils d'Aide à la Décision pour le Pilotage des Systèmes de Production", Laboratoire de Robotique et d'Intelligence Artificielle, Centre de Développement des Technologies Avancées, France.
- [3] C. LASSIETTE, "Les progiciels de gestion intégrée", rapporté de http://www.creg.acversailles.fr/spip.php?article209&var_recherche=progiciel, 2007.
- [4] D. BENTLEY, "Independent vs. dependent demand, MRP inputs and processing, Closed-loop MRP, MRP II, ERP", San José State University, Washington, 2006.
- [5] P. W. BLEVINS, "Enterprise Resource Planning (ERP): An Executive Perspective -- An Update", GLOVIA International, United states.
- [6] F. DARRAS, "Proposition d'un cadre de référence pour la conception et l'exploitation d'un progiciel de gestion intégré", Thèse Présentée en vue de l'obtention du grade de docteur, Institut National Polytechnique De Toulouse, octobre 2004.
- [7] S. SPIEKERMANN, "Introduction to Enterprise Resource Planning (ERP): System Analysis and Design - Lecture 11", Institute of Information Systems, Allemagne, 2007
- [8] encyclopédie scientifique en ligne. Rapporté de <http://www.techno-science.net/>
- [9] A. CHAABOUNI, "IMPLANTATION D'UN ERP (ENTERPRISE RESOURCE PLANNING) : ANTECEDENTS ET CONSEQUENCES", XVème Conférence Internationale de Management Stratégique, Annecy / Genève, Juin 2006.
- [10] Alain Fernandez. "Principe de l'ERP". Rapporté de <http://www.piloter.org/>
- [11] K. JONES, "L'ABC de la gestion intégrée : Guide d'introduction pour les dirigeants", Aberdeen Group, Inc., 2004.
- [12] Etat de l'art des ERP. Rapporté de <http://www.gs1.fr/>
- [13] ACTORSSOLUTION. Rapporté de <http://www.actors-solutions.com/article52>
- [14] Michel La flamme, commerce électronique.
Rapporté de http://www.cvm.qc.ca/mlaflamme/comm_e/Comm_e.html
- [15] Wikipédia l'encyclopédie libre Rapporté de <http://fr.wikipedia.org/wiki/Accueil>
- [16] Groupe media SYNAXIOME. Rapporté de http://www.synaxiom.com/solutions_pour_les_entreprises/solutions_d_applications/erp.html

REFERENCES

- [17] Ministère de l'Economie et du Commerce extérieur du Grand-Duché de Luxembourg. Portal, 2007. Rapporté de <http://www.entreprises.public.lu/content/erp/index.php#top>
- [18] N. RIOU, "Les ERP", Master 2 SIR Alternance, 2008
- [19] "Compiere Solution ERP & CRM Open Source", 9 Juillet 2004
- [20] A. MILED, " Les ERP libres: Une réalité et des atouts considérables", rapporté de http://www.abdelmajid-miled.com/articles_fr1.php?id=18, 2005.
- [21] F. GARCIA, " Systèmes d'information integres: Système d'information ... Proiciel de gestion integer ... ERP (Entreprise Ressource Planning)", IUP Tarbes – M1, 2006
- [22] L. GHARSALLAH, "Impact de l'ERP sur la performance : cas d'IGL", Université de Sfax - Mastère Professionnel, Rapporté de http://www.memoireonline.com/05/07/463/impact-erp-performance-cas-igl.html#_Toc165367699
- [23] "Définition E-PROCUREMENT", Rapporté de <http://www.agrojob.com/dictionnaire/definition-E-procurement-2784.htm>
- [24] J. CARDOSO, R. P. BOSTROM, A. SHETH, "Workflow Management Systems and ERP Systems: Differences, Commonalities, and Applications", Apporté de <http://dme.uma.pt/jcardoso/Research/Papers/ABS-ITM02.pdf>
- [25] A. BAAZI, A. DAHANE, "Entreprises & Systèmes d'Information", Communication à Inforum'2005, Oran, Algerie, Mars 2005.
- [26] Ministère de l'Économie, de l'industrie et de l'emploi, "Enquête sur les TIC et le commerce électronique en 2002: Définitions", Apporté de http://www.industrie.gouv.fr/sessi/enquetes/tic/tic2002_definition.htm
- [27] J.P. BRIOT et Y. DEMAZEAU, " Introduction aux agents" , Laboratoire d'Informatique de Paris 6, Université Paris 6, Rapporté de <http://www-poleia.lip6.fr/~briot/cours/intro-agents-iarfa02-03.pdf>.
- [28] Nikos Drakos, Computer Based Learning Unit, University of Leeds. Ross Moore, Mathematics Department, Macquarie University, Sydney. Rapporté de http://www.etis.ensea.fr/~revel/html/cours_IA/cours_IA.html.
- [29] J. FERBER et M. GALLAB, "Problématiques des univers multi-agents Intelligent", IN actes des journées nationales PRC.GRECO. Intelligence Artificielle. Toulouse, mars 1988.
- [30] M. Pěchouček, "Distributed Artificial Intelligence I", Gerstner Laboratory for Intelligent Decision Making and Control.

REFERENCES

- [31] Dr O. Kazar, " Conception et réalisation d'un modèle de réseau sémantique", mémoire en vue d'obtention du diplôme de Magister en informatique, Université de Constantine, 1996.
- [32] F. McCABE, S. McCANNELL, J. ODELL, K. RATICKAINE, K. STAUT et C. THAMPSON, "Agent Technology (Green Paper)". Agent Working Group OME Document ec/99 12.02 versions 0.9.24 Décembre 1999.
- [33] F. SANDAKLY, "Contribution à la mise en œuvre d'une architecture à base de connaissances pour l'interprétation de scènes 2D et 3D", thèse doctorat, université de Nice-Sophia Antipolis, 1995.
- [34] J. FERBER, " Les systèmes multi-agents, vers une intelligence collective", Interédition 1995.
- [35] Bih-Ru Lea, Wen-Bin Yu, "A MULTI-AGENT BASED ERP ARCHITECHTURE", Decision Sciences Institute, Annual Meeting Proceedings, 2002.
- [36] J. Kim, "Multi-Agent Based ERP", MAI Lab., 2003.
- [37] P. MASSOTTE, D. DIEP, R. BATAILLE, V. CHAPURLAT, A. MEIMOUNI, J. READY, "PABADIS : Plant Automation Based on Distributed Systems", Projet européen IST, 2003.
- [38] Site officiel de l'IST, Rapporté de <http://cordis.europa.eu/ist/about/about.htm>.
- [39] Site officiel de l'IST, Rapporté de <http://cordis.europa.eu/ist/projects/projects.htm>.
- [40] M. Pechoucek, A. Riha, J. Vokrinek, V. Marik, V. Prazma, "ExPlanTech: Applying Multi-agent Systems in Production Planning", Czech Republic.
- [41] M. PĚCHOUČEK, J. VOKŘÍNEK, J. HODÍK, P. BEČVÁŘ, J. POSPÍŠIL, "ExPlanTech: Multi-Agent Framework for Production Planning, Simulation and Supply Chain Management ", Czech Republic.
- [42] S. Labidi, W. Lejouad, "De l'Intelligence Artificielle Distribuée aux Systèmes Multi-Agents", INRIA INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE.
- [43] ABDELLI Belkacem, "Une approche basée agent pour le web sémantique", Mémoire en vue d'obtention du diplôme de Magister en informatique, Université de Biskra

REFERENCES

- [44] JAVA PLATE-FORME. Rapporté de
<ftp://ftp2.developpez.biz/developpo/java/coursjava/java.pdf>
- [45] Artem Katasonov, "Introduction to JADE", Université de Jyväskylä.
- [46] Tommy Yuan, Software Agents, "Introduction to JADE",2008.
- [47] Fabio Bellifemine, Agostino Poggi, Giovanni Rimassa, "JADE – A FIPA-compliant agent framework".