REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed Khider – BISKRA
Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie
Département d'informatique

N° d'ordre :……

Série :……..

Thèse

En vue de l'obtention du diplôme de Doctorat En Sciences

***Spécialité** : Informatique*

***Option** : Informatique*

Titre :

Une approche à base d'agents pour Data Mining à travers un ERP

Présentée et soutenue par :

Nadjib MESBAHI

le : 30/09/2018

Devant le Jury composé de :

Président :	Zaia ALIMAZIGHI	Professeur,	Université USTHB, Alger
Rapporteur :	Okba KAZAR	Professeur,	Université de Biskra
Examinateurs :	Mounia ABIK	Professeur,	Université Mohammed V de Rabat, ENSIAS, Maroc
	Kamel BOUKHALFA	Professeur,	Université USTHB, Alger
	Khaled REZEG	MCA,	Université de Biskra
	Abdelhamid DJEFFAL	MCA,	Université de Biskra
Invité :	Samir BOUREKKACHE	MCB,	Université de Biskra

Dédicaces

En premier lieu, je veux rendre grâce à Dieu, le Clément et le Très Miséricordieux pour son amour éternel. Je dédie cette thèse à :

ma mère pour sa tendresse et à mon père pour sa patience et son encouragement

ma femme et à mes enfants qui ont su comprendre mon occupation.

mes très chers frères et ma chère sœur pour leurs conseils,

mes cousins et cousines,

tous ceux que j'aime,

tous mes amis.

Remerciements

Tout d’abord, je remercie Dieu Tout Puissant de m’avoir donné la force et la patience nécessaire pour achever ce travail de thèse.

Mes vifs remerciements vont à Monsieur Okba KAZAR, professeur à l'Université de Biskra qui a assuré l'encadrement de cette thèse ainsi que pour ses précieux conseils et la confiance qu’il m’a accordée qui ont fortement contribué à mener à bien ce travail.

Je veux également remercier M. Zaia Alimazighi, Professeur à l’Université USTHB, pour m’avoir fait l’honneur de présider le jury avec autant d'intérêts.

Mes respects et ma gratitude vont aussi aux membres du jury : Mme Mounia ABIK, Professeur à l’Université Mohammed V de Rabat (Maroc), M. Kamel BOUKHALFA, Professeur à l’Université USTHB, M. Khaled REZEG, Maître de Conférences A à l'Université de Biskra, M. Abdelhamid DJEFFAL, Maître de Conférences A à l'Université de Biskra et M Samir BOUREKKACHE, Maître de Conférences B à l'Université de Biskra qui m’ont fait grand honneur de juger ce travail et qui par leurs observations constructives m’ont permis de l’enrichir.

Je tiens à remercier vivement M. Saber BENHARZALLAH, Maître de conférences A à l'Université de Batna 2 d’avoir suivi mon travail et aussi pour ses précieux conseils.

Enfin, je tiens à remercier tous ceux qui m’ont aidé et soutenu de près ou de loin pour l’accomplissement de cette thèse, surtout, mes chers amis MM. M. Zoubeidi, A. Merizig, A. Alloui, D. Rezki, O. Mesbahi, R. Mesbahi, N. Rabie, F. Zaiz, Z. Sayah, M. Zemmouri, A. Belhadi et C. Bakir.

Nadjib Mesbahi

ملخص

أصبحت اليوم البرامج المدمجة لإدارة الشركات (ERP) تمثل المركز العصبي لنظم المعلومات في الشركات. توفر هذه الأدوات إدارة متسقة ومتجانسة للمعلومات بين مختلف أصحاب المصالح التجاريين للشركة وذلك باستخدام قاعدة بيانات كبيرة وممركزه لبرنامج ERP.

نظرا لكمية البيانات الهائلة المخزنة في قواعد البيانات (ERP) الناتجة عن الاستخدام اليومي لنظام (ERP)، فمن المهم دمج أدوات لصنع القرار، و ذلك لتحليل وتفسير الكم الهائل لبيانات الأعمال . و هذا من أجل استخلاص المعارف المتعلقة بصنع القرار سوآءا على المستوي الداخلي أو الخارجي للشركة، مما يسمح لمديري الشركة باتخاذ القرارات بكفاءة و جودة عاليتين في مجال (التمويل، الموارد البشرية، الأسواق، المنافسة، ...). ولهذا الغرض، فإن التنقيب على البيانات " Data Mining " يعتبر تقنية قوية مخصصة لاستخراج المعارف لاستغلالها في تقييم وتحليل بيانات الأعمال من أجل اتخاذ القرارات بكفاءة أكثر في الشركة.

ومع ذلك، تحتاج خوارزميات التنقيب على البينات إلى قدرات معالجه كبيرة، والتي يمكن توفيرها من خلال استخدام المعالجة المتوازية والموزعة على عدة حواسيب، ولكن الصعوبة تمكن في أن تعقيد هذه الخوارزميات يزداد طردا مع تطور قاعدة بيانات ERP.

من هذا المنطلق، فإن الهدف من أطروحتنا هو اقتراح حلول لإعادة النظر في مهام التنقيب على البيانات الغير الخاضعة للرقابة على البرامج المدمجة لإدارة الشركات، وذلك باستخدام الأنظمة المتعدد الأعوان (Système Multi-Agents). مناهجنا المقترحة في هذه الأطروحة تسمح بمعالجة بيانات أعمال البرامج المدمجة لإدارة الشركات بطريقة متوازية وموزعة، وهذا راجع للجانب اللامركزي للنظام المتعدد الأعوان، الذي يهدف إلى توزيع عملية اكتشاف المعارف عبر عدة كيانات مستقلة ومتعاونة تسمي أعوان. النهج المقترح الأول هو عبارة عن "نهج قائم على الأعوان"Agents" لتجمع البيانات من خلال البرامج المدمجة لإدارة الشركات مع الأخذ في عين الاعتبار خصوصيات التنفيذ المتوازي الموزع. النهج الثاني المقترح يمثل "نهجًا موزعًا قائمًا على الأعوان لاستخراج قواعد ارتباط الأعمال"، والذي يعتمد على التقسيم الذكي لبيانات البرامج المدمجة لإدارة الشركات. للتحقق من صحة النهج المقترح، فإننا طبقناه في دراسة حالة حقيقية على الشركة الوطنية لخدمات الآبار(ENSP) . الهدف الرئيسي لعملنا هذا هو تحسين وتسريع عملية استخراج المعرفة من قاعدة البيانات الكبيرة لنظام (ERP). ونتيجة لذلك، ستصبح عملية اتخاذ القرار في البرامج المدمجة لإدارة الشركات أكثر تحسنا وأكثر فعالية.

الكلمات المفتاحية: البرامج المدمجة لإدارة الشركات (ERP)، تنقيب البيانات (Data Mining)، التجميع، قواعد ارتباط الأعمال، خوارزمية K-Means، خوارزمية Apriori، التقسيم الأفقي للبينات، الأنظمة المتعدد الأعوان (Système Multi-Agents).

Abstract

Nowadays, Enterprise Resource Planning (ERP) has become the nerve center of enterprise information systems. These tools provide consistent and homogeneous information management between the different business processes of the company using a large central ERP database.

Since a huge amount of data stored in this ERP database produced by the daily use of the ERP system, it is important to integrate decision-making tools to analyze and interpret these ERP business data. The aim of this is to extract decision-making knowledges both internal or external business environment, allowing the management of the company to be more efficient in her function of decision-making, in several fields such as (finance, HR, markets, competition ...). For this purpose, Data Mining is a powerful technology dedicated to knowledge extraction to exploit and analyze the ERP business data stored in a large centralized database.

However, Data Mining algorithms require generally large processing capabilities, that can be provided by resorting to parallel and distributed treatments, but the limits of these algorithms increases exponentially with the evolution of the (ERP) database.

The main goal of our thesis is to propose solutions to revisit unsupervised Data Mining tasks on the (ERP) software, by using the multi-agent system paradigm. Our proposed approaches allow a parallel and flexible manipulation of (ERP) business data due to the decentralized aspect of multi-agent systems, aiming at the distribution of the knowledge discovery process between several autonomous and cooperative entities. The first approach is to propose « An agent-based approach for data clustering through an ERP », by taking into account the specificities of distributed parallel execution. In the second approach, we propose « An agent-based distributed approach for extracting business association rules », which relying on the intelligent partitioning of ERP data. To validate this approach, we applied it during the realization of a real case study at the level of the National company of Well Services (ENSP). The main objective of our work is to improve and accelerate the process of extracting knowledge through the large database of (ERP) system. As a result, the decision process of (ERP) systems becomes more improved.

Keywords: ERP, Data Mining, Clustering, Business Association Rules, K-Means, Apriori, Horizontal Partitioning, Multi-Agent System (MAS).

Résumé

Aujourd'hui, le progiciel de gestion intégré (ERP) est devenu le centre nerveux des systèmes d'information des entreprises. Cet outil fournit une gestion homogène et cohérente des informations entre les différents acteurs métiers de l'entreprise à l'aide d'une grande base de données ERP centrale.

Vu qu'une énorme quantité de données stockées dans cette base de données ERP produite par l'utilisation quotidienne du système ERP, il est important d'intégrer des outils décisionnels pour analyser et interpréter ces données ERP métiers, afin d'extraire des connaissances décisionnelles tant au niveau interne qu'externe permettant au management de l'entreprise d'être plus efficace dans la prise de décision (finance, RH, marchés, concurrence,…). A cet effet, le Data Mining est une technologie puissante dédiée à l'extraction des connaissances à exploiter pour valoriser et analyser les données métiers ERP stockées dans une base de données volumineuse centralisée.

Cependant, les algorithmes propres au Data Mining nécessitent généralement de grandes capacités de traitement, qui peuvent être fournies par le recours à des traitements parallèles et distribués, mais la difficulté de ces algorithmes augmente exponentiellement avec l'évolution de la base de données ERP.

L'objectif de notre thèse est de proposer des solutions pour revisiter des tâches de Data Mining non supervisé sur le progiciel ERP, tout en utilisant le paradigme du système multi-agents. Nos approches proposées permettent une manipulation parallèle et flexible des données métiers ERP dues à l'aspect décentralisé des systèmes multi-agents, ayant pour objectif la distribution du processus de découverte de connaissances entre plusieurs entités autonomes et coopératives. Il s'agit en premier lieu de proposer « une approche basée agents pour le clustering de données via un système ERP », tout en prenant en compte les spécificités d'exécution parallèle distribuée. Et en second lieu, nous proposons « une approche distribuée à base d'agents pour l'extraction des règles d'association métiers », tout en se basant sur le partitionnement intelligent de données ERP. Pour valider cette démarche, nous avons appliqué ladite approche lors de la réalisation d'une étude de cas réel au niveau de l'Entreprise Nationale de Services aux Puits (ENSP). Le but principal de nos travaux dans le cadre de cette thèse est d'améliorer le processus d'extraction des connaissances à travers de la grande base de données centrale du système ERP, et ce en termes de temps d'exécution et qualité de présentation des connaissances. Par conséquent, le processus de décision de systèmes ERP devient plus amélioré.

Mots-clés : ERP, Data Mining, Clustering, Règles d'association métiers, K-Means, Apriori, partitionnement horizontal, Système Multi-Agents (SMA).

Liste des publications

La liste suivante représente les articles publiés dans le cadre de cette thèse.

A.1 Revues Internationales

- **Nadjib Mesbahi,** Okba Kazar, Saber Benharzallah, Merouane Zoubeidi, Djamil Rezki, and Abdelhak Merizig. A new approach agent-based for distributed of association rules by business to improve the decision process in ERP systems (AADARB-ERP). International Journal of Data Mining, Modelling and Management, Volume. xx, Issue. xx, 2018 Pages xxx-xxx – Inderscience **(Accepted).**

- **Nadjib Mesbahi**, Okba Kazar, Saber Benharzallah, Merouane Zoubeidi. A Cooperative Multi-Agent Approach-Based Clustering in Enterprise Resource Planning. International Journal of Knowledge and Systems Science, Volume 6 Issue 1, January 2015, Pages 34-45, IGI Publishing Hershey, PA, USA. **(Disponible en ligne).**

- **Nadjib Mesbahi,** Okba Kazar, Saber Benharzallah, Merouane Zoubeidi, Samir Bourekkache. Multi-Agents approach for data mining based k-Means for improving the decision process in the ERP systems. International Journal of Decision Support System Technology. Volume 7 Issue 2, April 2015 Pages 1-14 - IGI Publishing Hershey, PA, USA. **(Disponible en ligne).**

A.2 Conférences Internationales

- **Nadjib MESBAHI,** Okba KAZAR, Merouane ZOUBEIDI and Saber BENHARZALLAH. An agent-based modeling for an enterprise resource planning (ERP), ACIIDS 2014, Bangkok, Thailand LNCS Volume 551, 2014, pp 225-234, Springer.

- **Nadjib MESBAHI,** Okba KAZAR, Saber BENHARZALLAH and Merouane ZOUBEIDI. A cooperative multi-agent approach for knowledge discovery from the enterprise resource planning. Proceeding of the 3rd IEEE International Workshop on Advanced Information Systems for Enterprises (IWAISE 2014). Tunis, Tunisia. Http://www.lifl.fr/iwaise14/program.html.

- **Nadjib MESBAHI,** Okba KAZAR, Saber BENHARZALLAH et Merouane ZOUBEIDI. Approche Multi-Niveaux à base d'agents pour la modélisation d'un progiciel de gestion intégré : Etude du cas de l'Entreprise National de Services aux Puits. Proceeding of the International Conference on Artificial Intelligence and Information Technology (ICA2IT 2014). Ouargla, Algérie.

- **Nadjib Mesbahi,** Okba Kazar, Une approche à base d'agents pour la modélisation d'un progiciel de gestion intégré (ERP). Conférence internationale SIIE'2009. Hammamet (TUNISIE) – 12-14 Février 2009. http://siie2009.loria.fr/.

- Merouane ZOUBEIDI, Okba KAZAR, Saber BENHARZALLAH and **Nadjib MESBAHI.** ISPGI : Une architecture sémantique à base d'agents pour l'interopérabilité d'un PGI. Proceeding of the International Conference on Artificial Intelligence and Information Technology (ICA2IT 2014). Ouargla, Algerie.

- Merouane ZOUBEIDI, Okba KAZAR, **Nadjib MESBAHI** ET Saber BENHARZALLAH. Vers une architecture d'intégration de la sémantique via les Services Web : Etude de cas du Progiciel de Gestion Intégré. La conférence francophone sur les Systèmes Collaboratifs (SysCo'2014). Hammamet, Tunisie, pp. 101-114.

- Merouane ZOUBEIDI, Okba KAZAR, Saber BENHARZALLAH and **Nadjib MESBAHI.** Toward an architecture for the Semantic Integration in an Enterprise Resource Planning. Proceeding of the 3rd IEEE International Workshop on Advanced Information Systems for Enterprises (IWAISE'14). Tunis, Tunisia. Http://www.lifl.fr/iwaise14/program.html.

A.3 Chapitre du livre

- **Nadjib Mesbahi,** Okba Kazar, Saber Benharzallah, Merouane Zoubeidi, Djamil Rezki. A clustering approach based on cooperative agents to improve decision support in ERP. Technological Innovations in Knowledge Management and Decision Support. 2018, IGI Publishing Hershey, PA, USA. (**Disponible en ligne).**

Liste des figures

Liste des tableaux

Table des matières

Introduction générale

Contexte et problématique de recherche

De nos jours, les entreprises évoluent dans un environnement de plus en plus complexe et plein de changements. Elles sont confrontées à plusieurs défis : des marchés saturés, une compétitivité accrue, des clients plus exigeants et moins fidèles, etc. Dans ce contexte, les entreprises doivent faire face à une concurrence féroce dans un environnement en évolution perpétuelle. D'autant plus que les entreprises font face à un défi majeur, celui de satisfaire en permanence les exigences des différents clients. C'est pour cela que les entreprises se retrouvent dans l'obligation d'avoir recours à des outils optimisés, adaptés et sophistiqués qui facilitent les tâches et offrent des fonctionnalités riches et utiles. Parmi ces outils, les systèmes de gestion intégrés tels que l'ERP (Entreprise Ressources Planning). Ce dernier est un progiciel intégré de gestion et d'analyse par excellence destiné à la gestion de plusieurs domaines fonctionnels et/ou opérationnels au sein des entreprises. Il permet d'optimiser la diffusion interne des informations, d'améliorer les processus de gestion et d'automatiser les tâches, ce qui augmente énormément la réactivité des entreprises. L'ERP s'impose à présent comme un véritable standard pour la cohérence des informations entre les différents acteurs métiers à l'aide d'une grande base de données ERP centrale. Il est considéré comme un des facteurs de succès des entreprises actuelles, puisqu'il représente le centre nerveux du système d'information qui offre la possibilité de gérer l'ensemble des ressources de l'entreprise (moyens matériels et financiers, et ressources humaines).

Au fil du temps et à cause des usages quotidiens, la base de données ERP va systématiquement contenir une énorme quantité de données. Celles-ci dépassent notre habilité humaine d'analyse pour conclure les meilleures décisions. A cet effet, le progiciel ERP nécessite désormais d'être complété par un outil décisionnel. Ce dernier a pour mission principale l'analyse et l'interprétation des données ERP afin d'extraire des connaissances utiles pour la prise de décisions dans les entreprises. Dans ce sillage, l'extraction de connaissances à partir des bases de données (ECBD ou « Knowledge Discovery in Databases (KDD) ») est un domaine de recherche très actif qui permet la découverte de connaissances intéressantes ou de motifs (patterns), à partir des grandes bases de données. Ce domaine connaît une croissance spectaculaire mais il pose aussi certains défis à la communauté de la recherche. Le KDD a été mis en place au début de 1991 comme le processus non trivial d'extraction d'informations valides, nouvelles, potentiellement utiles, et compréhensibles à partir de données [1]. Le Data Mining est le cœur du processus KDD. Il permet à cet effet d'effectuer divers traitements plus ou moins complexes pour l'extraction des connaissances variées telles que la classification, le clustering et les règles d'association.

A cet effet et pour soutenir la prise de décision dans le système ERP, plusieurs approches ont été proposées pour utiliser la technologie dite « Data Mining » qui s'intègre avec la plateforme de l'ERP pour atteindre les objectifs décisionnels. Les techniques du « Data Mining » sont fondées sur plusieurs algorithmes gourmands en ressources et qui exigent des capacités de traitement indispensables pouvant être fournis grâce aux traitements parallèles distribués (décentralisés). Par ailleurs, plus la base de données ERP est

volumineuse, plus la difficulté de ces algorithmes augmente exponentiellement. Il est donc nécessaire de trouver de nouvelles méthodes qui tiennent compte des spécificités du progiciel ERP et de la technologie Data Mining également.

Contributions et objectifs du travail

Dans le cadre de cette thèse, nos travaux visent à proposer des solutions pour adapter certaines tâches de Data Mining suite aux caractéristiques du progiciel de gestion intégré (ERP) par l'utilisation de la technologie du système multi-agents, tout en respectant les spécificités de technologie Data Mining et ERP. Dans ce travail de recherche, nous abordons deux problèmes de Data Mining non supervisé : le clustering et les règles d'association que nous adaptons particulièrement au contexte de l'ERP qui a été spécialement conçu sur une architecture centralisée. L'utilisation du système Multi agents dans ce cadre est primordiale, puisqu'elle permettra de créer des modèles de connaissances claires et robustes. Cela est assuré par l'intégration des agents coopératifs au sein du processus de l'extraction de connaissances grâce à la distribution, la collaboration, la flexibilité, l'évolutivité et l'efficacité du SMA. Il rendra l'exécution des algorithmes du Data Mining d'une manière parallèle et distribuée pour une extraction efficace des connaissances dont la base de données ERP est volumineuse et centralisée. Les contributions principales de travaux de cette thèse s'articulent autour des deux volets suivants :

- **Proposition d'une approche multi-agents coopératifs pour le clustering de données via un ERP «CMAAC-ERP ».** Elle repose sur le parallélisme et la distribution des tâches à l'aide du système multi agents. L'approche proposée tourne autour du rôle du clustering de données ERP, basée sur la distribution de complexité de l'algorithme k-means sur plusieurs agents autonomes et coopératifs. Notre objectif tracé dans ce travail est de regrouper efficacement les enregistrements de la base de données ERP en classes d'objets similaires.
- **Proposition d'une approche spécifique pour l'extraction des règles d'association métiers à partir d'une grande base de données ERP centralisée «AADARB-ERP »,** tout en reposant sur le parallélisme et la distribution des données. Elle est particulièrement basée sur les agents coopératifs pour fournir une distribution et un partitionnement intelligent de données ERP métiers. En outre, elle est également capable d'exécuter de multiples processus d'extraction des règles d'association métiers, de façon parallèle et distribuée, à partir de grande masse de données ERP. Notre approche permettra la réduction du temps global de traitement des règles d'association métiers avec une meilleure présentation des règles par métier et, par conséquent, elle améliora le processus décisionnel dans les systèmes ERP.

Les objectifs de l'approche «AADARB-ERP » s'articule autour des points suivants :

1. Amélioration de la qualité de présentation de connaissances vis-à-vis de l'utilisateur Data Miner, par l'usage des règles plus claires offertes par la technique de règles d'association.
2. Limitation de l'espace de recherche des règles d'association via le partitionnement de données métiers ERP (qui concerne l'opération de Data Mining) sur des parties de

données moins volumineuses afin de paralléliser l'extraction des règles d'association par métier.

3. Distribution de ces parties de données sur plusieurs machines distantes pour rendre le traitement de données ERP en parallèle distribué, ce qui réduit le temps global de l'exécution des règles d'association.
4. Partitionnement horizontal de données ERP par métier, permet de produire des règles d'associations par métier plus significatives et plus cohérentes.
5. Réduction du temps global consommé par la tâche de l'extraction des règles d'association métiers à partir du système ERP.

Un projet d'application a été développé validant notre travail de recherche. En fait, la deuxième approche est plus fiable et efficace que l'approche classique basée sur l'algorithme « Apriori » vu les résultats obtenus d'après les expérimentations que nous avons réalisées. Le système développé est plus performant en termes de vitesse d'exécution et de flexibilité. Notre objectif déjà maintenu est celui d'exploiter toute une masse de données ERP pour relever un ensemble de connaissances décisionnelles en temps optimisé. Nos efforts consentis dans ce cadre, ont abouti à l'amélioration tangible des processus décisionnels sur une plateforme ERP.

Organisation de la thèse

Cette thèse est organisée en cinq chapitres. Après, la présentation de l'introduction générale, nous avons choisi d'introduire notre thèse par un chapitre qui présente les concepts liés aux progiciels de gestion intégrés (ERP). Nous avons présenté en premier lieu, les architectures des ERP, leurs avantages et leurs inconvénients. Ensuite, nous avons montré les différentes topologies existantes des ERPs et l'évolution du marché ERP. Nous avons présenté par la suite les processus d'implantation d'un ERP et les facteurs clés pour faire réussir cette implantation. Enfin, nous avons abordé l'impact de l'implantation de l'ERP sur la performance des entreprises.

Le deuxième chapitre se rapporte à la présentation du processus de l'extraction de connaissances à partir des données et de la technologie de Data Mining. Nous nous intéresserons plus précisément de techniques du clustering et d'extraction des règles d'association. Nous discuterons ainsi la technique de Data Mining distribuée où nous mettons l'accent sur les algorithmes de règles d'association parallèles et distribués.

Le troisième chapitre est un passage sur les travaux de Data Mining à base d'agents. Ce chapitre a l'objectif de situer nos travaux par rapport aux travaux existants dans la littérature. En premier lieu, nous montrons la motivation et les avantages de la contribution du système multi-agents dans le Data Mining. Nous présentons par la suite les travaux de Data Mining à base d'agents pour les tâches du clustering et de l'extraction des règles d'association avec des analyses comparatives.

Le quatrième chapitre vise à présenter notre première contribution pour une approche de clustering de données ERP à base d'agents coopératifs, et ce pour améliorer l'aide à la décision dans le système ERP. Il commence par la présentation des théoriques fondamentales où nous mettons l'accent sur le clustering à base K-means et les approches existantes pour intégrer le Data Mining avec la plateforme ERP, et ce afin de positionner notre approche proposée. Nous présentons par la suite l'architecture générale de l'approche multicouche

proposée pour clustering de données ERP. Ensuite, nous décrivons l'architecture fonctionnelle des agents du système proposé ainsi que leurs interactions.

Le dernier chapitre est consacré à la description de notre deuxième contribution pour une approche de règles d'association distribuée par métier basée agents à partir de données du système ERP. En premier lieu, nous montrons la motivation et les objectifs de notre approche proposée. Ensuite, nous présentons l'architecture générale du système développée et l'architecture du système multi agents proposée pour l'extraction des règles d'association métiers. Nous ferons par la suite une description détaillée de notre système, tout en expliquant les structures et les fonctionnalités des agents utilisés ainsi que leurs interactions. Après, nous faisons une brève présentation de l'environnement du développement et des interfaces de l'application développée pour valider cette approche. Enfin, nous clôturons notre travail par une expérimentation de l'approche proposée et une conclusion générale.

Nous terminons notre thèse par une conclusion générale qui vise à présenter le code source des agents développés, d'autre part, nous n'oublierons pas de signaler les perspectives possibles à notre travail.

Chapitre I : Progiciels de gestion intégrés (ERP)

I.1. Introduction

De nos jours, les entreprises doivent développer leurs activités dans un environnement de plus en plus complexe et difficile. Ils rentrent dans une concurrence réelle pour assurer une amélioration croissante de leurs services, d'accroitre leurs savoirs faires, de réduire les coûts, d'augmenter la production et de faire face aux défis du marché. Afin de prendre en considération ces facteurs, les entreprises s'orientent vers de nouveaux outils basés sur les technologies de l'information et de la communication. Ces outils optimisés facilitant les tâches et offrant des fonctionnalités plus riches et plus utiles, en l'occurrence les Progiciels de Gestion Intégrés(PGI) ou (Enterprise Ressource Planning) ERP.

L'ERP est un progiciel de gestion par excellence pour le pilotage des systèmes opérationnels des entreprises. Il permet une gestion homogène du système d'information de l'entreprise et une communication cohérente entre les différents acteurs métiers [2]. On peut dire que l'ERP est le centre nerveux du système d'information de l'entreprise, qui offre la possibilité de gérer l'ensemble des moyens de l'entreprise (ressources humaines, matérielles et financières) [1], [3].

Le premier chapitre s'attache tout d'abord à définir ce que recouvre le terme ERP aux entreprises. Pour le comprendre, nous allons d'abord nous intéresser à citer sa définition et ses principales caractéristiques. Ensuite, nous nous attacherons à décrire les architectures des ERP ainsi que leurs avantages et leurs inconvénients. Puis, nous montrons les topologies des ERPs existantes et l'évolution du marché ERP. Après, nous présentons les processus d'implantation d'un ERP et les facteurs clés de succès. Enfin, nous aborderons l'impact de l'implantation de l'ERP sur la performance des entreprises.

I.2. Historique de l'ERP

Au cours des années 1960-1970, les premiers systèmes de production, et plus précisément, la GPAO (Gestion de Production Assistée par Ordinateur) sont apparus dans les entreprises industrielles. Elles sont fondées essentiellement sur la méthode de gestion MRP (Material Requirements Planning) ou en français planification de ressource de production qui est utilisée pour assurer le calcul des besoins et d'établir une planification des délais. D'ailleurs, le principal but de la GPAO est de transformer les données commerciales relatives aux ventes en données de production. [4]

Au début des années 1980, le concept des MRP II remplace les MRP I, c'est l'époque où les progiciels commencent à s'imposer face aux développements spécifiques. Ils sont développés pour les moyennes et grandes entreprises, tout en intégrant des fonctions importantes ; telles que le calcul des coûts, le suivi de production, et la planification à long terme, moyen terme, court terme. Parallèlement, et dans la même période, le concept de CIM (Computer Integrated Manufacturing) a été utilisé pour l'intégration des activités de production [11].

Au début des années 1990, et avec la généralisation du concept d'intégration de systèmes informatiques et des processus métier dans tous les domaines de l'entreprise, et bien sûr, avec la maturité de méthode de gestion MRP, les ERP (Enterprise Resource Planning) ont vu le jour [10]. Les ERP sont des progiciels de gestion et d'analyse dotés d'une forte cohérence pour la diffusion des informations en interne, d'améliorer les processus de gestion et d'automatiser les tâches, ce qui augmente énormément la réactivité des entreprises.

Au début des années 2000, et avec la diffusion de l'internet l'ERP a incorporé d'autres extensions d'affaires tels que : CRM (Customer Relationship Management) pour la gestion des relations clients, SCM (Supply Chain Management) pour la gestion de la chaîne logistique, PLM (Product Lifecycle Management) pour la gestion du cycle de vie du produit, E-Business ou encore le Business intelligence [6].

Actuellement, les fournisseurs d'ERPs ont intégrés d'autres fonctions plus récentes et plus riches tels que : l'exécution des opérations et l'analytique en temps réel, l'utilisation des outils mobiles ainsi que le déploiement de la solution ERP dans une Platform Cloud [8]. Compte tenu de ce qui précède, le système ERP fournit à l'ensemble des acteurs de l'entreprise, une image unifiée, intégrée, cohérente et homogène de l'ensemble des informations dont ils ont besoin [210].

Aujourd'hui, les grands éditeurs qui ont regroupé une offre complète sont : SAP, ORACLE Business Suite et Microsoft avec sa gamme DYNAMICS. D'autres grands éditeurs, tels que SAGE offrent également des suites de gestion intégrées qui peuvent être, à juste titre, considérées comme un ERP. [5]

I.3. Définitions de l'ERP

L'acronyme ERP (Enterprise Ressource Planning) d'origine américaine, pour désigne les Progiciels de Gestion Intégrés (PGI). ERP est le terme le plus couramment utilisé.

Dans ce contexte, il existe plusieurs définitions de l'ERP, on les présente comme suit :

Premièrement, l'auteur de l'ouvrage « Manager avec les ERP » de Jean-Louis Lequeux [190], définit l'ERP comme étant : « un sous ensemble du système d'information capable de prendre en charge la gestion intégrale de l'entreprise, incluant la gestion comptable et financière, la gestion de la production et de la logistique, la gestion des ressources humaines, la gestion administrative ainsi que la gestion des ventes et des achats ».

Ce type de systèmes offre une communication parfaite entre les différents processus principaux de l'entreprise. A cet effet, l'objectif des ERP est de fournir un outil de gestion à l'entreprise permettant l'optimisation de la durée de mise sur le marché de ses produits et services.

L'auteur REIX en 2002 [211] a défini l'ERP comme « une application informatique paramétrable, modulaire et intégrée, qui vise à fédérer et à optimiser les processus de gestion de l'entreprise en proposant un référentiel unique et en s'appuyant sur des règles de gestion standard ». Cette définition met l'accent sur l'aspect de standardisation dans la gestion que fournit le progiciel ERP.

En effet, la définition proposée par Willis et al en 2003 [215] semble la plus complète, « 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 ainsi les informations nécessaires à la gestion et au contrôle des principales activités de l'entreprise en commençant par l'approvisionnement à la production/ exploitation… ; jusqu'à la commercialisation et la livraison des produits. Il évite de saisir les informations plus d'une fois et les met à la disposition des différents acteurs métiers de l'entreprise. [25]

En outre, l'auteur JONE en 2006 [214] propose une autre définition dans laquelle le terme ERP renvoie à l'infrastructure logicielle qui assure non seulement la cohérence interne de l'ensemble de l'entreprise mais aussi apporte un support à ses processus commerciaux externes.

I.4. Caractéristiques d'un ERP

Selon la référence [5], l'ERP est considéré comme étant le centre nerveux du SI de l'entreprise, qui possède les caractéristiques globales suivantes :

1. gestion effective de plusieurs domaines de l'entreprise par des modules intégrés ou des progiciels susceptibles d'assurer une collaboration des processus ;
2. existence d'un référentiel unique des données ainsi que les indications nécessaires pour retrouver les données elles-mêmes sur la base de données ;
3. adaptation des règles de fonctionnement (professionnelles, légales et règles dictées par le marché) ;
4. unicité d'administration du sous-système applicatif (les applications) ;
5. uniformisation des Interfaces Homme-Machine (IHM) : même ergonomie des écrans, mêmes boutons, même famille de barres du menu, mêmes touches de fonctions et de raccourcis ;
6. existence d'outils de développement ou de personnalisation de compléments applicatifs.

Il faut noter ici que la solution intégrée qui ne répond pas au moins aux trois premiers critères précédemment définis n'est pas un ERP [5].

I.5. Architecture d'un ERP

Dans notre recherche bibliographique, nous avons constaté qu'il existe deux types d'architectures pour un progiciel de gestion intégré à savoir, l'architecture technique et l'architecture modulaire.

I.5.1. Architecture technique d'un ERP

Actuellement, la plupart des systèmes ERP tels que SAP et Oracle utilisent une architecture de type Trois-Tiers Client/serveur avec une base de données centralisée de type relationnelle. [12]

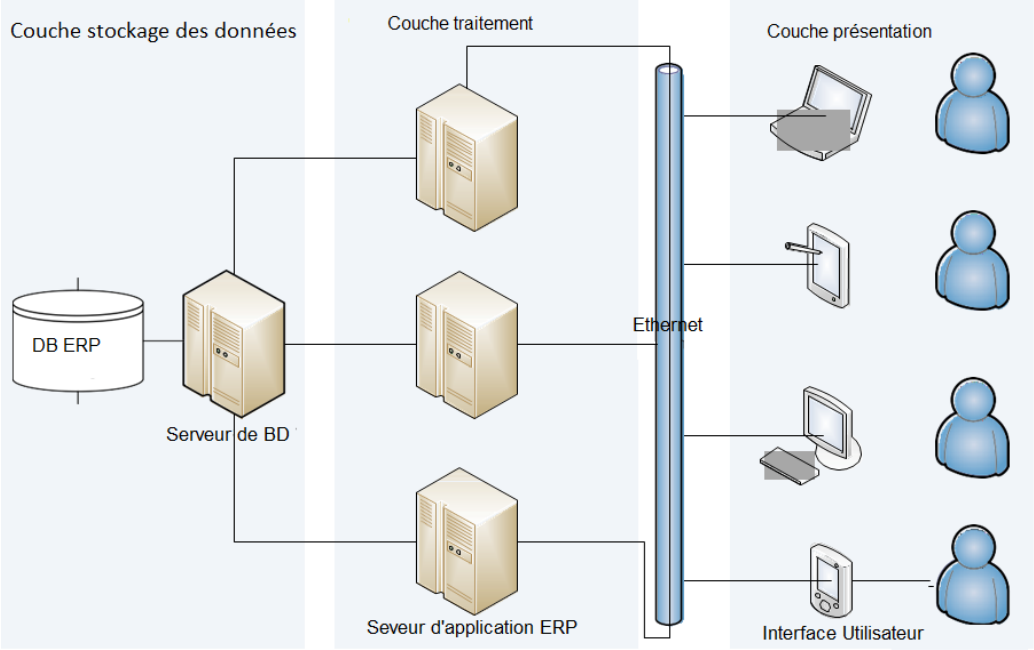


Figure I-1 : Architecture technique de l'ERP [12]

L'architecture technique illustrée dans la Figure I-1 comporte trois couches. La première est la couche de « Serveur de base de données » qui est employée pour gérer et stocker un volume important de données à travers d'une base de données relationnelle. La deuxième couche c'est le « Serveur d'application » où on trouve l'application serveur de l'ERP qui permet d'implémenter le logique métier, les processus d'affaires, les règles métier, l'authentification et la gestion des utilisateurs. La dernière couche est celui de « Présentation » qui est représentée sous la forme d'une interface utilisateur. Elle est employée sur plusieurs périphériques allant de postes de travail jusqu'aux appareils mobiles. Cette architecture technique offre une grande flexibilité et évolutivité au progiciel ERP, par exemple l'extensibilité de l'architecture est obtenue par l'introduction d'une nouvelle couche (Tiers) pour un autre serveur (serveur application, serveur web ... etc.)

Plus récemment, une autre architecture technique est apparu dans le monde des ERP, pour donner une autre alternative de mettre en place du système ERP qui est « l'architecture en mode hébergé », possible soit auprès d'une solution Cloud de l'éditeur lui-même, par exemple comme la solution « SAP HANA Enterprise Cloud (HEC) », soit auprès des partenaires commerciaux de l'éditeur de progiciel dans leurs environnements respectifs. [8]

I.5.2. Architecture modulaire d'un ERP

Selon [5] 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 la littérature des ERP, il existe plusieurs classifications de l'architecture modulaire d'un ERP, mais elles sont toutes d'accord que les modules essentiels pour un ERP sont ceux montrés dans la figure suivante :

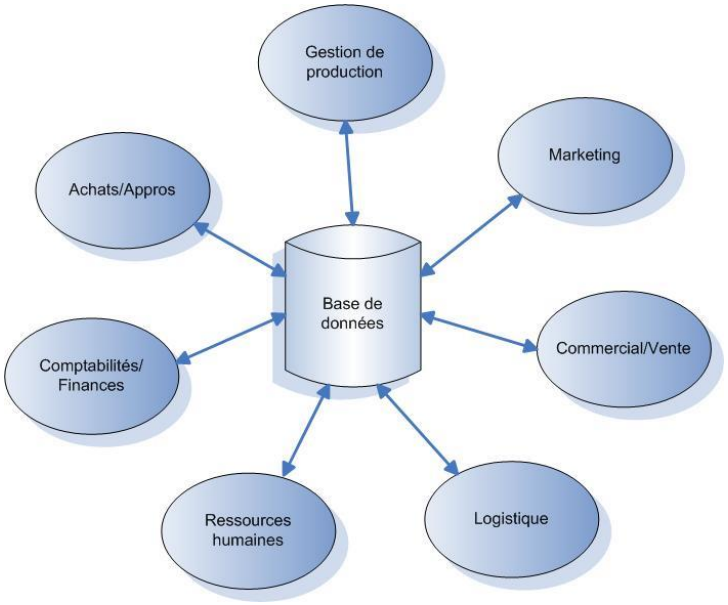


Figure I-2 : Architecture générale d'un ERP [7]

Dans cette architecture, l'utilisation d'une même base de données assure la cohérence de l'information dans l'organisation qui va permettre une gestion homogène et cohérente du système d'information de l'entreprise. Autour de la base de données ERP, il se trouve plusieurs modules fonctionnels et/ou opérationnels en citant entre autres : le module ressources humaines, le module comptable et financier, le module production, logistique ainsi que le module des ventes et des achats.

Selon [7], un système ERP est modulaire dans le sens où il est possible de n'avoir qu'une ou plusieurs applications en même temps, ou module par module. Les systèmes modulaires comme celui de l'ERP, permettent une compatibilité sure entre les modules, ils s'imbriquent comme des blocs de Lego et fonctionnant ensemble (pas de recours à la vérification de compatibilité entre modules). La figure I.3 représente l'architecture modulaire d'un ERP.

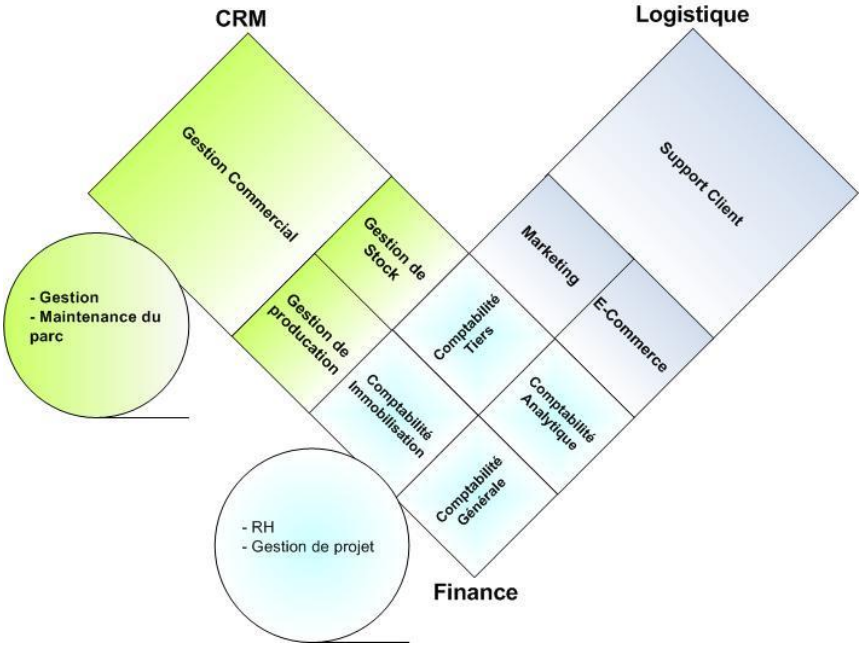


Figure I-3 : Architecture modulaire ERP [7]

Cette architecture modulaire intègre plusieurs modules retouchant aux grandes activités de l'entreprise qui rendent le système d'information intégré et unifié.

Une autre classification d'après la référence [3] basée sur les domaines fonctionnels de l'entreprise classe les modules de l'ERP en trois domaines distincts :

Le domaine métier : rassemble un ensemble des modules qui prennent en charge la réalisation des processus situés sur la chaîne de valeur de l'entreprise, tel que les modules de gestion des achats, de gestion de production, ou de gestion des stocks ...etc.

Le domaine du pilotage : les modules de ce domaine prennent en charge les fonctions qui facilitent l'analyse et la synthèse du fonctionnement du système de l'entreprise, tels que les prévisions, et les évaluations 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.

I.6. Avantages d'un ERP

L'ERP présente plusieurs avantages par rapport à un nouveau développement spécifique, nous les citons dans les points suivants [7] & [9] & [13] :

1. L'intégrité et l'unicité du SI : c'est à dire qu'un ERP permet une logique et une ergonomie unique à travers sa base de données, elle aussi unique au sens "logique". En bref, le progiciel ERP peut offrir un outil efficace pour éviter la redondance des données entre différents systèmes d'information des entreprises.

2. Le partage du même SI entre les différents acteurs métiers facilite la communication interne et externe
3. Un meilleur suivi des processus est assuré par une parfaite coordination des services (suivi de commande efficace ou bonne maîtrise des stocks par exemple)
4. Standardisation de la gestion des ressources humaines (pour les entreprises gérant de nombreuses entités parfois géographiquement dispersées)
5. L'utilisateur a la possibilité de récupérer des données de manière immédiate, ou encore de les enregistrer.
6. Les mises à jour des données sont réalisées obligatoirement en temps réel et dispatchées sur les modules concernés.
7. Un ERP est une solution multidevise et multilingue, par conséquent, il est adapté au marché international, particulièrement aux multinationales.
8. L'absence d'interfaçage entre les modules par le fait que les traitements sont synchronisés et les processus de gestion sont optimisés.
9. La maintenance corrective est garantie directement par l'éditeur de la solution et non plus par le service IT de l'entreprise. Cela permet une bonne maîtrise des coûts, des délais de réalisation et de déploiement.
10. Mise à disposition aux manageurs de l'entreprise des indicateurs de performances pour une prise de décision plus adéquate.

I.7. Intérêts de l'ERP dans l'entreprise

Plusieurs études, enquêtes et ouvrages ont été effectués aux USA et en France sur la mise en œuvre des ERP ont permis de mettre en avant les axes les plus fréquents d'amélioration cités par les entreprises pour justifier leur investissement dans l'ERP. Ainsi nous citons [14] :

1. Au niveau de l'ensemble des activités de l'entreprise, un pilotage par des indicateurs de performances et une meilleure connaissance des structures de coûts.
2. Au niveau des ventes, une augmentation du nombre d'offre qui doit permettre de générer des revenus complémentaires.
3. Au niveau financier, l'optimisation des fonds propres avec, par exemple, une amélioration des délais de paiement et une diminution des stocks.
4. Au niveau des achats, une optimisation des conditions d'achat par une meilleure capacité à négocier, avec une diminution des temps d'approvisionnement.
5. Au niveau de la production, la maîtrise des coûts de production et des niveaux de stock pour accroître la productivité et la réactivité.
6. Au niveau de la communication entre les différents services de l'entreprise, une meilleure circulation des informations par l'existence d'une base de données unique.

I.8. Inconvénients des ERP

Les ERP soulèvent quelques inconvénients parce qu'ils nécessitent la participation de nombreux acteurs à savoir : l'éditeur du progiciel ERP, l'intégrateur de la solution et l'équipe projet. A cet effet, nous citons leurs inconvénients [22] et [23] :

1. Lourdeur et rigidité d'implantation de la solution, si le projet ERP n'est pas bien piloté.
2. Coût élevé sauf dans le cas des progiciels ERP libres qui demandent seulement les formations aux utilisateurs de l'entreprise.
3. Résistance aux changements exprimés par les personnels du progiciel parce que les utilisateurs sont souvent mal préparés à cette tâche.
4. Sous-utilisation du logiciel parce que très souvent surdimensionné par rapport aux besoins des entreprises.
5. Certains modules du progiciel ERP peuvent être moins efficaces qu'un développement spécifique à l'entreprise ou une application spécialisée.
6. Nécessité de faire une maintenance continue qui est une forme de captivité vis-à-vis de l'éditeur.
7. Nécessité de bien connaître les processus de l'entreprise afin de bien paramétrer le fonctionnement de l'ERP propre aux besoins de l'entreprise

I.9. Topologies des ERP

De façon générale, on peut classer les ERP en trois catégories principales [17],[18],[19],[20] et [21]: ERP généralistes, ERP spécialisés (métiers) et ERP verticaux.

I.9.1. ERP généralistes

Comme leur nom l'indique les ERPs qui appartiennent à cette catégorie couvrent en globalité les processus d'une entreprise. Ils sont conçus pour répondre à des besoins de gestion standard, communs à la plupart des entreprises : gestion comptable et financière, gestion commerciale, gestion des ressources humaines, et, le cas échéant, gestion de production, gestion logistique. En outre, les ERPs généralistes peuvent être implémentés sur une large majorité des secteurs d'activité, qui sont peu complexes, mais ne permettent pas de couvrir toutes les spécificités des métiers.

I.9.2. Les ERP spécialisés (ERP métiers)

Aujourd'hui, les ERP généralistes n'ont pas toujours l'agilité nécessaire pour suivre le rythme de changements aussi rapides parce que les métiers évoluent en permanence et réclament des outils de plus en plus flexibles. Pour cela, les ERP spécialisés vont permettre de répondre parfaitement aux problématiques métiers complexes ou particulières. Ils se reposent sur des règles métiers relatives 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. Néanmoins, Ces ERPs métiers, sont moins coûteux que ceux des ERP généralistes, conviennent mieux aux PME dont les ressources sont généralement plus limitées que celles des grandes entreprises. Par ailleurs, les ERP métiers s'avèrent souvent plus adéquats à ce

type de besoin : un projet plus court, bien délimité, capable de générer des bénéfices mesurables. Parmi les secteurs d'activités recouvrés par cette catégorie de progiciel sont : la santé, la pharmacie, l'agroalimentaire, la chimie, la biologie, le BTP, le commerce ou la logistique.

I.9.3. Les ERPs verticaux

Les ERP verticaux appelés aussi « best of breed », qui sont apparus pour couvrir les insuffisances fonctionnelles des ERP généralistes. Ils fournissent des autres alternatives pour répondre parfaitement bien à un métier ou une spécificité, par l'ajout des couches logicielles plus ou moins bien intégrées, soit par des développements sur mesure, soit par des applications spécifiques dédiées à telle ou telle fonction métier. Cependant ils nécessitent des interfaces avec le reste des composants du système de l'entreprise, ce qui rend son intégration compliquée et coûteuse. Ce type de progiciel est considéré comme ERP du fait qu'il se comporte d'une manière à répondre à une logique semblable à celle d'un ERP. Pour Patrick Rahali, analyste senior au CXP, la verticalisation des ERP comprend plusieurs volets : [19]

- un pré-paramétrage ou une pré-configuration du progiciel ;
- l'ajout de modules ou de fonctions spécifiques ;
- l'implication de consultants connaissant le métier de l'entreprise ;
- une méthodologie de mise en œuvre adaptée.

I.10. Marché des ERP

Selon la mise à jour de Gartner le 9 juillet 2013, le marché de l'ERP est de près de 3.700 milliards de dollars au niveau mondial en 2013, une croissance moyenne de 4,5% par an. Ainsi, une étude récente en 2014 menée par des cabinets de conseil et d'études permet d'évaluer les parts de marché des grands éditeurs d'ERP propriétaires au niveau mondial. De ce fait, le marché mondial des ERP est dominé en 2014 par SAP (leader mondial), qui représente 25 % de la couverture en ERP. Oracle qui avait racheté Peoplesoft, représente 13 % du marché mondial. Sage est bien placé en particulier avec une clientèle de PME.

La figure I-5 ci-dessous illustre la répartition des parts de marché des principaux ERP en 2014 au niveau mondial.

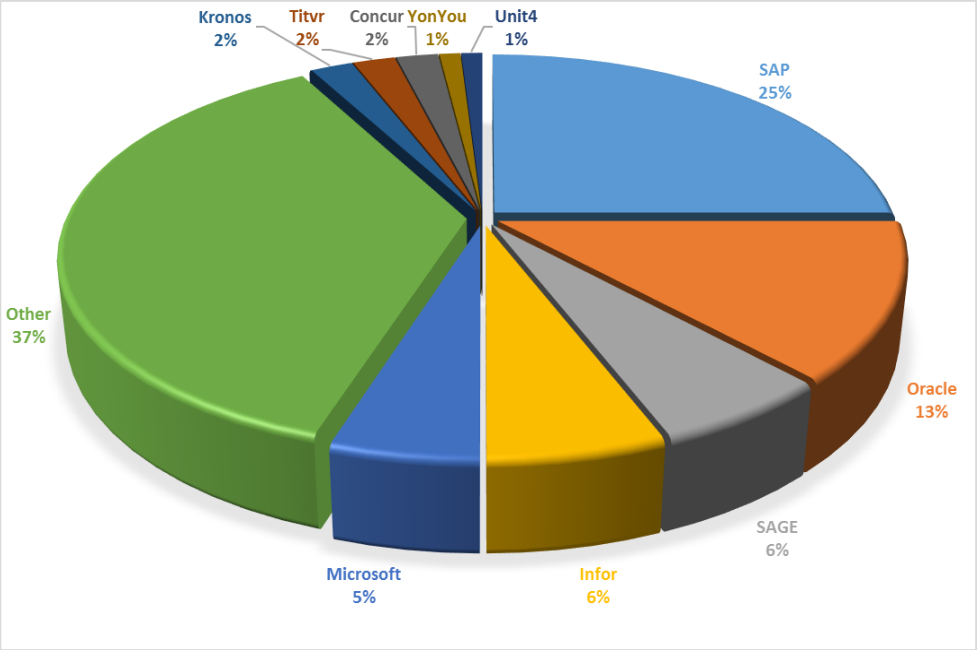


Figure I-4 : Marché mondial des ERP en 2014

Généralement, dans le marché, on distingue deux types des ERP ; les ERP propriétaires et les ERP open source :

I.10.1. Les ERP propriétaires

Les ERP propriétaire sont des progiciels créés par une société spécialisée dans la conception et la mise en place de logiciels et de systèmes informatiques. Ils nécessitent l'achat d'une licence. Parmi les principaux du marché mondial nous citons : SAP, People soft, Oracle, SAGE et Microsoft avec sa gamme DYNAMICS.

SAP est la société qui a donné naissance aux ERP. Elle est aujourd'hui encore leader sur le marché mondial des ERP. Le progiciel SAP a obtenu un succès important sur les grandes entreprises en proposant un progiciel multilingue et multidevises. L'éditeur SAP s'intéresse également aux marchés PME par sa suite Business One [16].

L'ERP SAP est une application client-serveur dont ses modules couvrent l'ensemble des domaines métiers de l'entreprise. Chaque module de SAP répond aux besoins complets de gestion. La solution SAP R/3 est complètement paramétrable, ce qui permet la possibilité aux entreprises d'implémenter certains modules spécifiques ou bien tous les modules fonctionnels. [16]. De plus, le progiciel SAP R/3 peut être adapté à des besoins spécifiques grâce à son environnement de développement.

I.10.2. Les ERP open source

Les ERP open source sont gratuits puisqu'il n'y a pas de coût de licence, mais il faut inclure dans le calcul du cout d'acquisition total : les frais de maintenance et d'assistance technique. Parmi les principaux progiciels Open Source: Aria, Open Bravo, Odoo, Compiere, ERP5, OFBiz, Tiny ERP. Dolibarr, SQL Ledger.

Les ERP open source sont très utilisés par les PME, car ils sont plus faciles à intégrer et à personnaliser, même si cela implique d'excellentes connaissances informatiques. En plus, le choix d'un ERP open source donne plusieurs avantages par rapport les ERP propriétaires :

- il n'implique pas l'acquisition d'une licence, ce qui permet de faire de sérieuses économies,
- il est 20 % à 50 % moins cher qu'un ERP propriétaire,
- l'absence de licence sur les ERP open source donne une forme d'indépendance aux entreprises qui ne prennent aucun engagement [17].

De nos jours, les ERP tirent parti du Web. Les utilisateurs peuvent accéder à ces systèmes au moyen d'un navigateur web. Ces progiciels sont de plus en plus destinés vers l'extérieur et sont capables de communiquer avec les clients, les fournisseurs et d'autres organisations.

I.11. Implantation de l'ERP

Comme nous l'avons déjà présenté dans la section précédente, il existe une évolution de produits ERP sur le marché. Selon le progiciel ERP choisi, l'entreprise doit penser à investir sur les ressources, humaines, matérielles et financières nécessaires à l'implémentation de l'ERP tels que : le recrutement des employés, la création d'un réseau, l'acquisition d'un serveur ERP et du nouveau matériel informatique … etc. L'implantation d'un ERP est un projet promoteur pour les entreprises qui supposent la refonte du SI et particulièrement la normalisation des procédures de gestion au sein de l'entreprise.

Dans cette section, nous présentons le processus de l'implantation de l'ERP ainsi que les facteurs clés pour que l'implantation de la solution ERP soit un succès.

I.11.1. Processus d'implantation de l'ERP

Vu les problèmes rencontrés lors de l'implantation d'un ERP, les auteurs de [216] ont introduit un modèle d'implantation comportant quatre étapes pour la mise en place de progiciel ERP :

1. Etape de conception : concerne l'étape de planification dont les décisions importantes sont prises et qui comprend les tâches suivantes : 1) la sélection du progiciel, 2) l'identification du chef de projet, 3) l'approbation du budget et 4) du plan de projet.

2. Etape d'implantation : permet d'analyser la situation actuelle de l'entreprise, la mise en œuvre du nouveau système ERP et son paramétrage avec les systèmes déjà existants ainsi que les tests de validation.

3. Etape de stabilisation : Dans cette étape le système ERP se stabilise tout en permettant la détection des différentes erreurs et éventuellement les corriger.

4. Etape d'amélioration continue : Elle contient les tâches suivantes : 1) la maintenance du système, 2) l'introduction de nouvelles fonctionnalités au système, 3) le support des utilisateurs, 4) les mises à jour éventuelles, etc.

4.

Un autre processus en cinq étapes est proposé par Ehie et Madsen en 2005 [24] pour la mise en œuvre d'un ERP telles que : préparation du projet, plan d'affaires, la réalisation, la préparation finale, et la mise en production et support (voir la Figure I-5 ci-dessous).

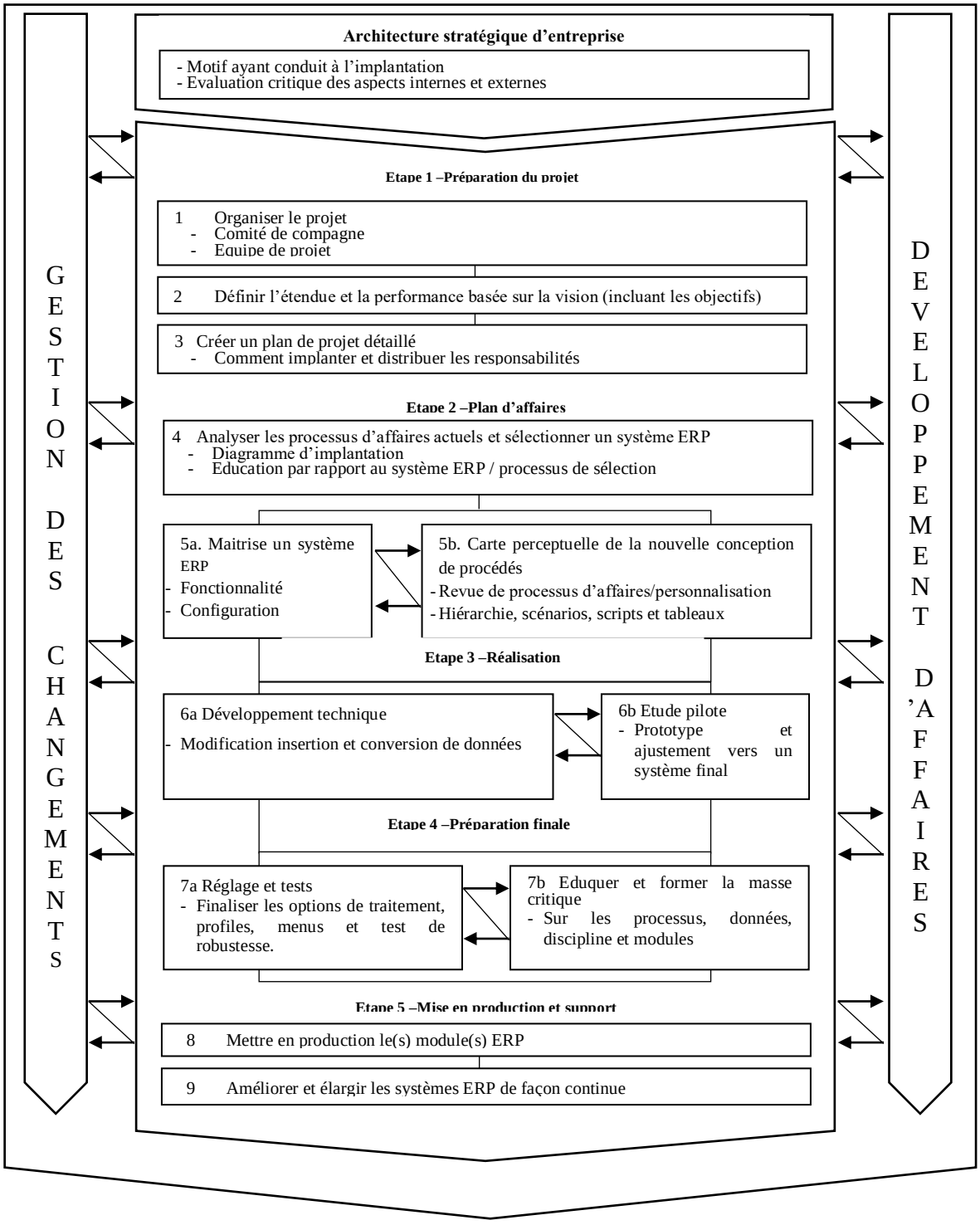


Figure I-5 : Processus d'implantation d'un ERP (Ehie et Madsen,2005) [24]

Ce processus passe à travers une première phase de préparation pour définir les objectifs et le plan du projet, établir l'équipe de projet avec les rôles de leadership et de fixer des

objectifs budgétaires. Une deuxième phase permet d'analyser les processus d'affaires actuels afin de sélectionner un système ERP approprié. Ensuite l'équipe de projet sera formée sur les fonctionnalités et les configurations du système ERP sélectionné afin d'être mieux adapter avec les processus d'affaires de l'entreprise. Dans la phase de réalisation, l'équipe de projet se concentre sur la mise en œuvre du nouveau système ERP, y compris la modification, le développement d'interfaces, et la conversion des données. En parallèle, une équipe teste le nouveau progiciel sur un site pilote, de manière à tester chaque module de l'ERP avec son processus métier correspondant, afin de détecter les problèmes éventuels. Quant à la préparation finale, l'ensemble du processus métiers est entièrement intégré et testé dans toute l'organisation avec des données complètes et avec des divers scénarios. Les utilisateurs finaux sont formés dans cette phase aussi bien pour l'utilisation du progiciel. Le but principal de la phase de mise en production et support est d'assurer que les modules du système ERP sont stables et mis en production. Les managers de l'entreprise peuvent avoir des extensions futures sur le nouveau progiciel pour des raisons concurrentielles.

I.11.2. Facteurs clés de succès de l'implantation de l'ERP

Rabaa'i et al (2009) [217] se base sur plusieurs revues de la littérature pour identifier les facteurs clés de succès (FCS) de la mise en œuvre de l'ERP. Il considère que les douze (12) premiers FCS sont les plus importants et les plus fréquemment cités :

1. Implication et soutien de la direction générale,
2. Gestion du changement, Gestion de projet ERP,
3. Business Process Reengineering et personnalisation du système ERP,
4. Formation des utilisateurs,
5. Composition de l'équipe d'ERP,
6. vision et planification, sélection des consultants et des relations,
7. plan de communication , sélection du système ERP,
8. l'intégration du systèmes ERP et les mesures d'évaluation post-mise en œuvre.

- **Implication et soutien de la direction générale**

L'implication et le soutien de la direction générale est considéré par plusieurs chercheurs comme étant un des FCS de la mise en œuvre de l'ERP. En effet, l'implantation d'un ERP doit obtenir l'approbation et le soutien de la direction générale. Le chef de l'entreprise doit attribuer une priorité importante au projet de l'implantation de l'ERP. Il doit s'engager avec sa bonne volonté et sa propre participation à fournir les ressources nécessaires pour la réussite de la mise en œuvre de l'ERP [25].

- **Gestion du changement**

Les auteurs Ehie et Madsen en 2005 [24] ont déclaré que la mise en œuvre d'un ERP implique plus que l'évolution des systèmes logiciel ou matériel. Idéalement, par la réingénierie des processus d'affaires, la mise en œuvre de l'ERP peut aider l'organisation à

bénéficier de niveaux plus élevés d'efficacité et d'amélioration de la performance. Cependant, la mise en œuvre de l'ERP peut causer des changements qui conduisent à la résistance chez les employés [26]. Par conséquent, l'équilibrage des conflits entre le personnel et la technologie, et la gestion efficace des employés dans le processus de changement sont des éléments clés pour la réussite de la mise en œuvre d'ERP [27].

- **Gestion de projet ERP**

La gestion efficace de projet ERP est un facteur fondamental pour que l'implantation de l'ERP réussie. Les auteurs Bingi, [28] ont constatés que « un manque de compréhension des besoins du projet, l'incapacité d'attribuer le leadership et l'absence de l'orientation du projet » sont les principaux facteurs pour que la mise en œuvre de l'ERP échoue. Ainsi, la gestion efficace de projet devrait définir des objectifs clairs, élaborer un plan de travail et des ressources, et de suivre attentivement les progrès du projet.

- **Réingénierie des processus d'affaires et la personnalisation de système ERP**

Il existe deux approches pour la mise en œuvre des systèmes ERP dans une organisation : la réingénierie des processus d'affaires et la personnalisation de système ERP [29]. La réingénierie des processus d'affaires crée des changements profonds dans les processus organisationnels afin de les adapter aux fonctions de l'ERP. D'autre part, quand une organisation souhaite maintenir ses processus existants à l'aide d'un système ERP, il est possible de personnaliser les fonctions de l'ERP. Cependant, de nombreuses recherches indiquent que la personnalisation de l'ERP devrait être évitée ou réduite au maximum afin d'assurer le plein des avantages offerts par les systèmes ERP [30] et [31].

- **Formation des utilisateurs**

La formation de l'utilisateur final est considérée, comme étant, un facteur crucial pour la mise en œuvre de l'ERP. En raison de la complexité du système intégré de l'ERP, la formation de l'utilisateur final est essentielle pour une solide compréhension de la façon dont le système ERP fonctionne et comment l'utiliser. Par conséquent, l'éducation et la formation de l'utilisateur final approprié permettra de maximiser les avantages de l'ERP et d'accroître la satisfaction des utilisateurs.

- **Composition de l'équipe de projet ERP**

La composition de l'équipe d'ERP est également importante pour la mise en œuvre d'ERP réussie ; une équipe de projet ERP devrait être composé de représentants de toutes les unités fonctionnelles liées à l'ERP.

- **Sélection des consultants et des relations**

Les consultants ERP jouent un rôle essentiel dans la mise en œuvre de l'ERP. Durant l'implantation de l'ERP, les consultants peuvent être des ressources de connaissances essentielles pour les matériels, les logiciels et les personnels de l'ERP. Ils peuvent aussi aider l'équipe de projet ERP et vérifier le projet ERP dans sa phase finale. D'autre part, et afin de

réussir la maintenance du système après post-mise en œuvre, le transfert des connaissances par les consultants est crucial pour l'organisation.

- **Plan de communication**

Une communication forte au sein de l'organisation au cours du processus de la mise en œuvre augmente le pourcentage de réussite de l'implantation l'ERP. Il permet aux utilisateurs de l'organisation à comprendre le but et les avantages attendus du projet, ainsi que de partager les progrès du projet.

I.12. Impact de l'implantation de l'ERP sur la performance dans l'entreprise

L'implantation croissance des ERP dans les entreprises s'accompagne d'une attente en termes d'amélioration de leur performance. D'ailleurs, cette idée est déjà introduite par Marciniak en 2001 [219] qui voient que la raison principale derrière le choix de la solution ERP est l'augmentation de la performance. Elle est expliquée par une optimisation des coûts et un accroissement de la réactivité et de la flexibilité de l'entreprise. Cependant, les auteurs dans [25] révèlent qu'il existe une problématique pour mesurer cette performance. Ils ont retenu une vision sur quatre dimensions liées à l'implantation des ERP dans l'entreprise : une dimension projet, une dimension technique, une dimension comportementale, une dimension organisationnelle. En plus, les auteurs dans [218] ont ajouté la dimension sociétale.

L'étude de CHAABOUNI en 2006 [25] présente l'impact de l'implantation de l'ERP sur la performance de l'entreprise, qui est mesuré sur un ensemble des critères d'évaluations, sur trois dimensions : (1) Dimension économique et financière, (2) Dimension organisationnelle, (3) Dimension humaine.

1)- l'impact d'ERP sur la performance économique et financière

Selon les travaux effectués par les auteurs de [33] lors de l'analyse des impacts financiers de la mise en œuvre de l'ERP, une amélioration considérable de la performance de l'entreprise a été constatée après trois ans d'implémentation du système ERP, par une baisse dans certains ratios, tel que le ratio de coût des marchandises vendues par revenus et le ratio d'employés par revenus pour chacune des trois années examinées suite à la mise en œuvre du système ERP.

En outre, l'implantation de l'ERP dans l'entreprise peut acquérir des avantages concurrentiels et ce, en offrant des produits à faibles coûts et en améliorant la relation de l'entreprise avec ses clients et avec les différentes parties prenantes [32].

Par conséquent, on peut résumer que l'impact de l'implantation de l'ERP sur la performance économique et financière de l'entreprise réside dans les points suivants : (1) Optimisation des ressources et des coûts, (2) Une réduction des délais et un accroissement de la productivité. (3) Diminution des prix et une amélioration de la qualité des prestations fournies au client. (4) Augmentation des ventes de l'entreprise qui le deviendra plus compétitive.

2)- L'impact d'ERP sur la performance organisationnelle

La performance organisationnelle est définie comme étant « la manière dont l'entreprise est organisée pour atteindre ses objectifs et la façon dont elle parvient à les atteindre ». En se référant aux travaux de Kalika [213] et Chaabouni [212], les critères d'évaluation de la performance organisationnelle sont les suivants : la qualité de la circulation de l'information, les relations entre les services, la coordination, la coopération, le degré de contrôle, la communication, la décentralisation, la flexibilité et l'intégration. En plus, l'auteur Reix [211] a ajouté le processus de décision.

Par ailleurs, l'impact de l'implantation de l'ERP sur la performance organisationnelle de l'entreprise est accomplis sur plusieurs points : (1)- Modification de la structure de l'organisation par la création de nouveaux services (2)- la réorganisation des services informatiques (3) - Modification de la nature, la circulation et la création de l'information. (4)- L'affectation de processus de décision plus rapide par l'implantation de l'ERP. (5)- Facilitation de processus de contrôle et la culture de l'organisation [34].

On peut conclure que l'ERP permet d'améliorer la qualité des informations communiquées, de favoriser la coordination, de décentraliser les décisions et de faciliter le contrôle.

3)- L'impact des ERPs sur la performance humaine

L'implantation de l'ERP va être l'occasion d'améliorer des connaissances du personnel. En effet, par l'introduction de l'ERP, les utilisateurs de ce progiciel doivent acquérir de nouvelles compétences pour mieux la manipuler. Ils arrivent ainsi un certain niveau de confiance et d'efficacité lors de l'utilisation de l'ERP leur permettant d'améliorer leurs productivités [32] et [35].

En outre, l'ERP supporte l'intégration avec les outils décisionnels tels que le Data Mining, qui fait l'objet du chapitre II, pour aider les dirigeants de l'entreprise dans leurs tâches de planification et de prise de décision. D'ailleurs, l'ERP est conçu principalement sur une base de données centralisé, ce qui nous permet de l'intégrer rapidement avec l'outil décisionnel « Data Mining » dont le but est d'analyser intuitivement la richesse des données ERP Métiers et de répondre aux problématiques de plusieurs fonctions simultanément : la production, la logistique, les finances, les ventes...etc.

I.13. Conclusion

L'implémentation d'une nouvelle solution de type ERP devient un défi majeur pour les entreprises, aussi bien sur le plan financier qu'à la qualité de son fonctionnement. De ce fait, Il représente l'enjeu stratégique de l'entreprise. Cela implique la nécessité de donner tous les moyens et ressources nécessaires afin de faire réussir le processus d'implémentation de l'ERP, en commençant évidemment par les ressources humaines et la qualité du projet.

Il est aussi d'une importance primordiale de prendre conscience, une fois que le progiciel ERP a été mis en place, et de s'assurer que les connaissances extraites et la maîtrise du système ERP continueront à être assurées durant tout son cycle de vie.

Chapitre II : KDD et Data Mining

II.1. Introduction

Aujourd'hui, les potentialités de collecte et de stockage des données ont augmenté d'une manière plus rapide au fil du temps. Elle nous permet de recueillir une énorme quantité de données par des usages quotidiens de nombreuses applications de gestion sur leurs bases de données, comme le cas précis de l'ERP, qui gère plusieurs domaines fonctionnels et opérationnels sur une grande base de données centrale partagée. Cette augmentation exceptionnelle de bases de données a dépassé nos capacités et notre habilité humaine de vouloir comprendre sans utiliser aucuns outils. En plus, elle a produit des besoins urgents pour analyser, synthétiser et extraire les connaissances cachées dans ces réservoirs de données. Des nouvelles techniques sont venues pour répondre à ces besoins qui peuvent intelligemment et automatiquement transformer les données traitées en informations et connaissances utiles, en occurrence le Data Mining.

Le Data Mining est un processus permettant de découvrir des relations cachées, représentées par des connaissances « intéressantes » ou des motifs (patterns), à partir d'une grande base de données. La découverte de ces relations comprend plusieurs avantages au profit des entreprises tels que la prédiction de la production qui va être vendue, la réalisation des campagnes de publicité ciblées, etc. Selon [58] le Data Mining constitue un support important d'aide à la décision tant dans des secteurs concurrentiels (domaine industriel, domaine commercial, domaine bancaire, assurances) que dans des secteurs tels que la santé, l'environnement... Il supporte un ensemble de traitements différents pour la découverte de connaissances variées. Ces traitements sont la classification, la segmentation et les règles d'association. En effet, les connaissances découvertes peuvent être exprimées sous forme d'un ensemble de règles associatives, qui permettent d'identifier une relation conditionnelle entre un ensemble d'attributs appelés itemsets qui sont enfouies dans les bases de données.

Ce chapitre a pour objectif de présenter le processus d'extraction de connaissances à partir des données (KKD), la Data Mining et ses tâches. Nous nous intéresserons plus précisément aux techniques du Clustering de données et l'extraction des règles d'association afin de mieux comprendre leurs fonctionnements.

II.2. Motivation et définition du Data Mining

Depuis le début des années 1990, plusieurs facteurs sont liés pour le développement en croissance du Data Mining telles que : la puissance de calcul importante est disponible sur les ordinateurs ; le volume des bases de données augmente énormément ; des réseaux à grande échelle ayant un débit croissant, ce qui permet de soutenir la distribution des calculs et des informations, etc. En outre, la prise de conscience de l'intérêt commercial de l'entreprise pour l'optimisation des processus de fabrication, vente, gestion, logistique ….etc.[24].

De nombreuses définitions, issues de la presse et la littérature spécialisée, sont proposées sur le terme de « Data Mining », nous citons quelques-unes d'entre elles :

Selon le Groupe Gartner, le terme « Data Mining » appelé aussi la fouille de données, , est le processus de découverte de nouvelles corrélations, modèles et tendances en analysant une grande quantité de données, tout en utilisant les technologies de reconnaissance des formes ainsi que d'autres techniques statistiques et mathématiques [41].

• Dans [42] plusieurs auteurs considèrent que le Data Mining est un domaine interdisciplinaire utilisant dans le même temps des techniques d'apprentissage automatiques, des statistiques, des bases de données, de reconnaissance des formes, et de visualisation pour déterminer les façons d'extraction des connaissances à partir des bases de données volumineuses. (Figure II-1)

• Une autre définition est citée dans la référence [43] soulignant que le Data Mining, est un processus inductif, itératif et interactif dont l'objectif est la découverte de modèles de données valides, nouveaux, utiles et compréhensibles dans de larges bases de données.

Selon la référence [44] le Data Mining est un ensemble des méthodes et techniques destinées à l'exploration et l'analyse de grandes bases de données informatiques, de façon automatique ou semi-automatique, en vue de détecter dans ces données des règles, des associations, des tendances inconnues ou cachées, des structures particulières restituant l'essentiel de l'information utile tout en réduisant la quantité de données permettant d'étayer des prises de décisions ».

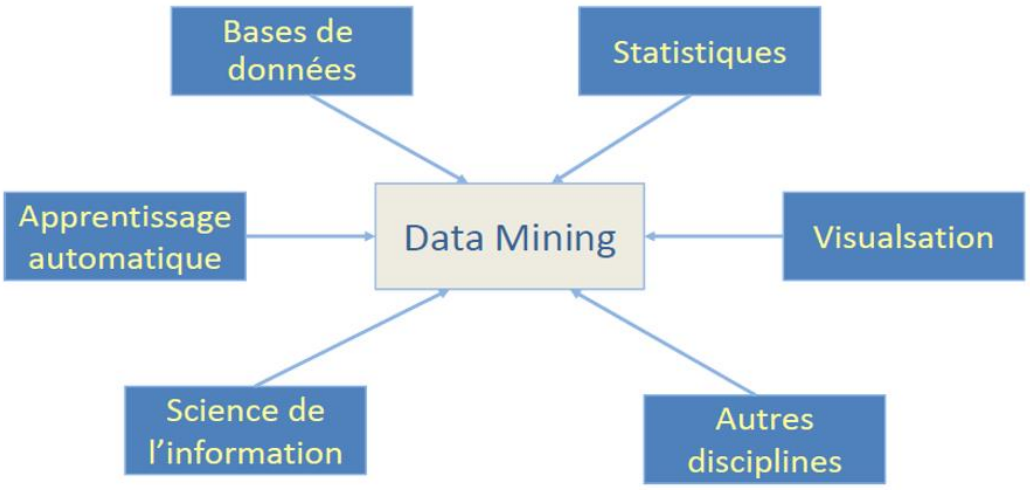


Figure II-1 : Data Mining : Union de disciplines variées [43]

II.3. Processus KDD et Data mining

Le Data Mining est souvent confondu par des non-spécialistes qui le voient comme un processus équivalent au KDD (Knowledge Discovery in Database). Cependant, la plupart des chercheurs considèrent le Data Mining comme une étape essentielle dans un processus plus général de l'extraction des connaissances à partir des données. Dans ce qui suit, nous utilisons terme KDD pour désigner l'extraction des connaissances à partir des données (ECD).

Avant de présenter le processus général de KDD, nous essayons de montrer la différence entre le Data Mining et le KDD, par les définitions suivantes :

D'après [48] «Le KDD est un processus itératif et interactif d'analyse d'un grand ensemble de données brutes afin d'extraire des connaissances exploitables par un utilisateur, analyste, qui y joue un rôle central».

Le Data Mining est : « le cœur du processus KDD, c'est le module dont le rôle est de fouiller dans les grandes bases de données pour extraire les connaissances cachées, son importance est due à la disponibilité d'énormes quantités de données dans un état de croissance permanente ». [46]. On peut dire que le KDD est un véhicule dont le Data Mining est le moteur.

Pour récapituler, nous pouvons fusionner les deux définitions [47] et [52] : Le KDD est un processus non-trivial, semi-automatique et itératif d'extraction des informations implicites, précédemment inconnues et potentiellement utiles concernant les données stockées dans les bases de données, composé de plusieurs étapes allant de la sélection et préparation des données jusqu'à l'interprétation des résultats, en passant par la découverte proprement dite, le Data Mining.

Avant de finaliser cette section, il est important de synthétiser les points suivants :

- Il faut bien noter que le KDD est un processus, c.à.d. un ensemble d'étapes et d'actions dont la finalité est l'extraction de connaissances corrélées au sein des données.
- La non trivialité, reflète au fait que, contrairement à la statistique qui est confirmatoire, le Data Mining est plutôt exploratoire [49]. Elle se justifie également par le fait que le processus KDD passe par plusieurs étapes.
- Le KDD est un processus itératif, c.à.d. une seule étape peut s'appliquer plusieurs fois. Cependant cela arrive assez rarement dans la pratique.
- Le KDD est un processus interactif, d'où on trouve l'utilisateur au cœur de ce processus puisque le Data Mining n'est pas un robot qui seul doit fouiller de larges bases de données afin d'extraire des connaissances utiles pour l'entreprise [50].
- Les résultats du Data Mining devraient être non seulement utiles mais aussi, visible et compréhensibles par les utilisateurs du domaine pour les aider à donner une décision finale. [51]

Les déférentes étapes de ce processus KDD sont présentées dans la figure ci-dessous :

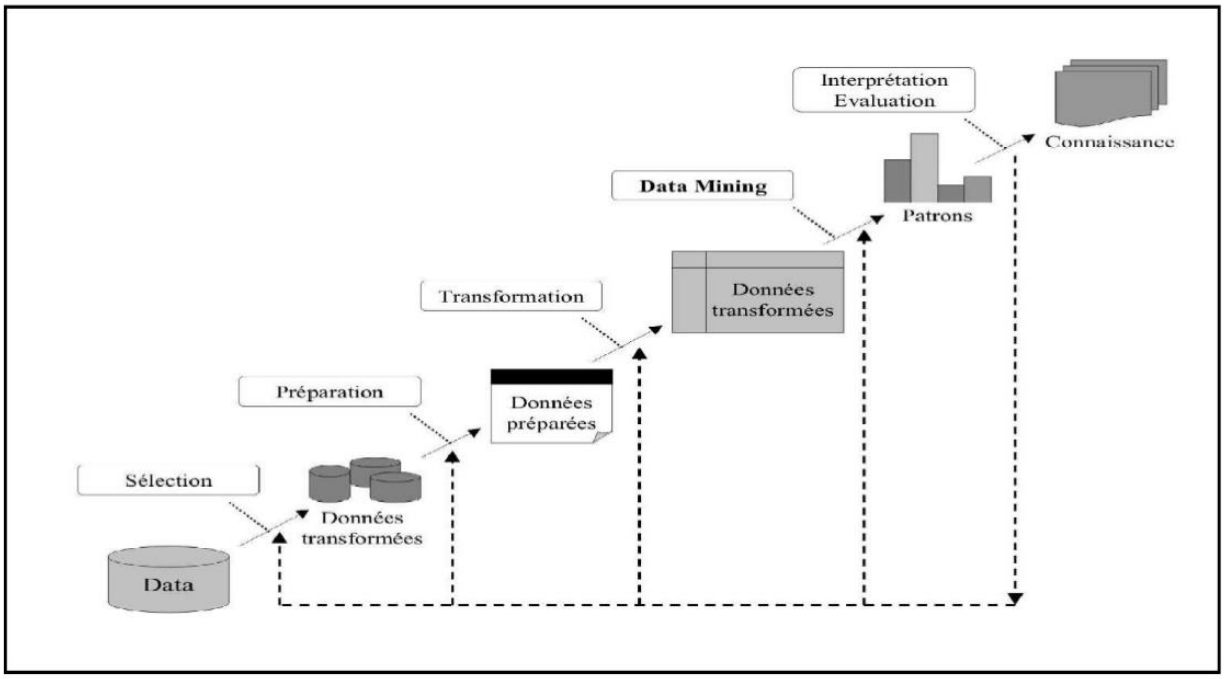


Figure II-2 : Processus KDD et Data Mining [52]

• **Sélection de données** : le but de cette phase est l'extraction à partir d'un plus grand stock de données seulement celles qui sont appropriées à l'analyse d'exploitation de données. Cette extraction de données aide à rationaliser et accélérer le processus.

• **Préparation de données** : cette phase de KDD est concernée par les données nettoyant et les tâches de préparation qui sont nécessaires pour assurer des résultats corrects.

• **Transformation de données** : Les données sélectionnées dans l'étape précédente vont subir une transformation dont le but est de les rendre dans une forme appropriée pour les méthodes et les techniques de Data Mining.

• **Data Mining** : le but de la phase d'exploitation de données est d'analyser les données par un ensemble approprié d'algorithmes afin de découvrir les modèles et les règles significatifs et produire les modèles prédictifs. C'est l'élément de noyau du cycle de KDD.

• **Interprétation et Évaluation** : tandis que les algorithmes de Data Mining ont le potentiel de produire un nombre illimité de modèles cachés dans les données, beaucoup de ces derniers peuvent ne pas être significatifs ou utiles. Cette phase finale est visée choisissant ces modèles qui sont valides et utiles pour prendre de futures décisions économiques. [53]

N'oublions pas de noter ici que dans la littérature, il existe des travaux qui parlent sur le modèle de référence CRISP-DM (Cross-Industry Standard Process for Data Mining). Le modèle CRISP-DM a été conçu en fin 1996, qui prouve sa réussite sur le terrain par l'expérience pratique dans l'entreprise et permet d'orienter les travaux de Data Mining. Les travaux persistants dans la littérature discutent sur ce modèle ont deux visions :

- En tant que méthodologie, CRISP-DM comprend des descriptions des phases typiques d'un projet DM et les tâches entrées dans chaque phase ainsi qu'une explication des différentes relations entre eux.
- Comme modèle de processus, CRISP-DM présente une vision claire de cycle de vie du Data mining..

II.4. Tâches de Data Mining

En réalité, le Data Mining représente un ensemble de traitements très divers afin de découvrir des connaissances variées. Selon [54], les traitements de Data Mining peuvent être exprimés en termes de six tâches suivantes :

• **La classification** : consiste à étudier les caractéristiques d'un nouvel objet pour lui attribuer une classe prédéfinie dont l'objectif est de créer un modèle qui peut être appliqué aux données non classifiées dans le but de les classifiées. [55]

• **L'estimation (**ou Régression) : similaire à la classification à part que la variable de sortie est continue au lieu d'être catégorielle. [44]

• **La prédiction** : est identique à la classification et à l'estimation, sauf que dans la prédiction on prend en charge la relation temporelle entre les variables d'entrée et les variables de sortie.

• **Le clustering** : (ou la segmentation) correspond au regroupement d'enregistrements ou des observations en groupes (classes) d'objets similaires.

• **La découverte de règles d'association** : (ou groupement par similitude) consiste à déterminer quels attributs "vont ensemble", ce qui signifie, la rechercher des implications entre les attributs.

• **La description** : Permettre à l'analyste d'interpréter les résultats d'un modèle de Data Mining, soit d'un algorithme, de manière la plus transparente et efficace possible. [57]

II.5. Techniques de Data Mining

Dans la littérature il existe plusieurs techniques du Data Mining issus des disciplines scientifiques diverses (Apprentissage automatique, statistiques, intelligence artificielle et base de données) pour effectuer les taches du Data Mining citées dans la section précédente. Nous catégorisons les techniques du Data Mining les plus connues en deux types :

II.5.1. Les techniques prédictives ou supervisées

Les techniques prédictives ou supervisées sont utilisées pour découvrir de nouvelles connaissances à partir des données présentes, elles sont utilisées aussi pour prédire des variables cibles. De nombreux algorithmes Data Mining appartenant à ce type de techniques ont été utilisés, parmi les plus connus on peut citer : les arbres de décision [67][68][69][70][71], les k- plus proches voisins [44] [61][74] et les réseaux de neurones [44]

II.5.2. Les techniques descriptives ou non supervisées

Elles s'intéressent à trouver des modèles intelligibles mais cachés par le volume de données. Elles sont utilisées pour réduire, résumer et synthétiser les données dont il n'y a pas de variable cible à prédire. Ces techniques descriptives sont basées sur plusieurs algorithmes, nous citons les principaux de ceux-ci : Clustering [61][62][63][64][65][66] et les règles d'association[72][73]

Le domaine Data Mining est très vaste, il regroupe plusieurs techniques très différentes les unes des autres. Les deux sections suivantes examinent deux techniques de Data Mining, le Clustering et les règles d'association qui ont été utilisés pour conduire et concentrer la recherche présentée dans cette thèse.

II.6. Le Clustering

Le Clustering consiste à la division de l'ensemble de données en groupes d'objets similaires appelés Clusters. Il est considéré comme une technique d'apprentissage non supervisé, c'est-à-dire sans savoir de connaissance à priori sur les groupes existants (y compris leur nombre). Cette technique a été exploitée dans plusieurs domaines tels que l'apprentissage machine, le traitement d'image/vidéo, la bio-informatique et la biochimie. Selon, le but de clustering ou les caractéristiques de données, différents types d'algorithmes de clustering ont été développés et qui feront l'objet de la section II.6.4.

II.6.1. Principe du clustering

Le principe du clustering consiste à partitionner l'ensemble de données en sous-ensembles de données pour construire des clusters, en respectant les deux critères suivants :

• diminuer la distance entre les données d'un même cluster ;

• augmenter la distance entre données de clusters différents.

Ce principe nécessite donc l'utilisation d'une fonction de mesure de distance pour mesurer la similarité entre les instances de données, qui est généralement liée au type de données traitées. [58]

II.6.2. But de clustering

Selon l'objectif à atteindre, les applications de clustering sont classées en trois catégories principales : la segmentation, la classification et l'extraction de connaissances [79].

A – La segmentation

Le principal but de la segmentation d'une base de données est de réduire l'ensemble de données traitées afin de faciliter leur traitement. Dans ce cas, nous parlons de la condensation ou de la compression de données. Généralement, cette méthode est utile dans la segmentation des images dans les bases de données spatiales, pour identifier les différentes régions homogènes de l'espace décrit (maisons, routes, champs, rivières, etc.).

B– La classification Les primordiales applications de cette technique sont dans la gestion de la relation avec le client. Cette technique consiste à l'identification de sous-population ayant des caractéristiques similaires ou proches dans les bases de données. Comme exemple, dans une base de données client, le but principal est de mettre des profils de clients aux comportements similaires, afin de comprendre leurs attentes et leurs besoins en fonctions de leur appartenance à un profil donné.

C– L'extraction de connaissances

Il n'y a pas de but prédéterminé d'utiliser le clustering pour l'extraction de connaissances. Le clustering dans ce cas est destiné à utiliser dans le contexte de la génération d'hypothèses ou la modélisation prédictive, dans le but d'aider à comprendre la structure des données en regroupant les classes homogènes et en inférant les règles qui caractérisent les données de domaine. Il peut être utilisé dans l'analyse du web, l'analyse des données textuelles, étude de ventes, bio-informatique, analyse génétique, diagnostics médicaux...etc.

II.6.3. Etapes du processus de clustering

Le processus de clustering utilise plusieurs critères spécifiques afin de partitionner la base de données en clusters homogènes. Cependant, l'étape de prétraitement de données est nécessaire avant de lancer la tâche de clustering sur l'ensemble de données. Le processus de clustering est devisé en quatre étapes qui sont présentées dans la Figure II 3.

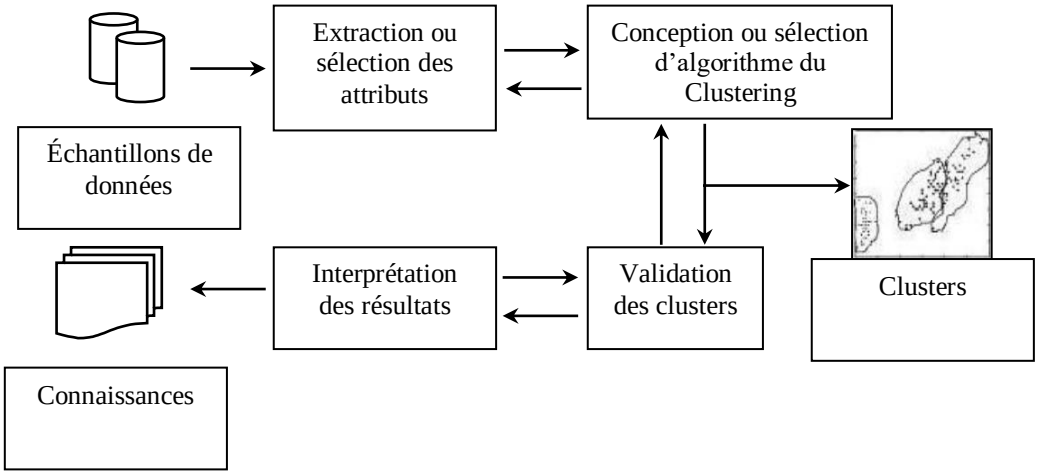


Figure II-3 : Les étapes du processus de Clustering [101]

1 - Extraction ou sélection des attributs

Une opération prétraitement des données est nécessaire pour assurer la qualité de données avant d'exécuter la tâche de clustering. L'objectif de cette étape est d'identifier les attributs dans lesquels le clustering doit être correctement lancé afin d'encoder autant que possible les informations concernant la tâche désirée.

2 - Conception ou sélection d'algorithme du clustering

Selon la définition d'un bon schéma de clustering de données, le choix de l'algorithme de clustering approprié est effectué. L'algorithme de clustering est caractérisé par deux

paramètres importants qui sont la mesure de proximité et le critère groupant pour définir les clusters représentant la base de données.

1. La mesure de proximité. C'est une mesure qui détermine comment les deux points sont similaires.
2. Critère groupant. Il peut être exprimé par l'intermédiaire d'une fonction de coût ou d'un autre type de règles.

Le choix de la mesure de proximité et le critère groupant influencent directement à la qualité de partitionnement de l'ensemble de données.

3 - Validation des clusters

Cette étape est basée sur un ensemble des techniques et des critères pour vérifier l'exactitude des résultats de l'algorithme de clustering. Le partitionnement final des données demande un certain genre d'évaluation dans la plupart des applications du clustering puisque ses algorithmes produisent des clusters qui ne sont pas connus à priori.

4 - Interprétation des résultats

Dans cette étape, Le rôle de l'expert de domaine d'application est primordial pour intégrer et affiner les résultats de clustering par d'autres analyses expérimentales afin d'extraire des bonnes connaissances. Dans la section suivante nous décrirons les différents algorithmes utilisés pour la tâche clustering de données.

II.6.4. Algorithmes de clustering

De nombreux algorithmes ont été développés pour la tâche du clustering. Ils peuvent être catégorisés sur plusieurs groupes tels que : Les algorithmes de partitionnement, hiérarchiques, basés sur la densité, basés sur la grille et basés sur les modèles.

Les algorithmes de partitionnement créent un ensemble initial de k partitions, où le paramètre k représente le nombre de partitions (clusters ou groupes) à construire, qui est fixé a priori ; ensuite, Ils utilisent un processus itératif en fonction du nombre k qui consiste à affecter chaque enregistrement au cluster le plus proche selon une fonction de distance ou un indice de similarité, que nous abordons dans la section suivante. Les algorithmes typiques de partitionnement incluent K-Means [203] et [65], KMedoids, KModes [206], CLARANS [204] et [66]). Dans notre approche que nous allons proposer dans le chapitre IV, nous nous intéresserons à modéliser l'algorithme « KMeans » en utilisant le paradigme du système multi-agents dans le but de regrouper efficacement les enregistrements de données métiers ERP en groupes d'objets homogènes.

Les algorithmes hiérarchiques créent une décomposition hiérarchique de l'ensemble de données. Ces algorithmes peuvent être classés comme étant soit agglomératifs (Bottom up), soit divisifs (Top down), en fonction de la façon dont la décomposition hiérarchique s'est formée. Les algorithmes agglomératifs considèrent initialement une partition constituée par des clusters, regroupant au départ une seule instance de données, et qui regroupent ensuite les clusters "voisins" jusqu'à un critère d'arrêt. Les algorithmes connus dans ce type sont : (Single-Link, CompleteLink, Average-Link[208], Agnes [157], Chameleon [64], Birch [62],

Cure [204]). Les algorithmes décisifs considèrent initialement qu'une partition constituée d'un seul cluster contient l'ensemble des instances de données, ainsi qu'ils découpent, par la suite, ces clusters de manière itérative jusqu'à un critère d'arrêt. L'algorithme le plus utilisé dans ce type est : DIANA [157]).

Les algorithmes basés sur la densité regroupent les objets en fonction de la notation de densité. Dans ce type d'algorithmes, les clusters se développent en fonction de la densité des objets de voisinage et de connectivité ou selon une certaine fonction de densité. Les algorithmes les plus connus sont DBSCAN [183], CLIQUE [159], MAFIA [161], et DENCLUE [188].

Les algorithmes basés sur les grilles utilisent la structure de la grille pour partitionner l'espace de description des données en différentes cellules et effectuent un regroupement sur la structure de la grille pour former les clusters. Parmi Les algorithmes du clustering basés sur les grilles, nous pouvons citer, STING [189] et WaveCluster [193].

Les algorithmes basés sur les modèles ont émis l'hypothèse d'un modèle pour chacun des clusters afin de trouver le meilleur résultat qui ajuste les données avec le modèle. Ce type d'algorithmes est plus proche des algorithmes basés sur la densité, concernant la naissance des classes particulières qui améliorent un certain modèle préétabli. Deux familles d'approches des algorithmes basés sur les modèles sont distinguées : les approches probabilistes [207] et [209] et les approches neuronales [205].

II.7. Règles associatives

L'extraction des règles d'association est considérée parmi les tâches les plus populaires dans le domaine du Data Mining, qui a attiré plus d'attention des chercheurs, et ce, depuis les travaux de [69], [70], [72] et [75]. Les règles associatives sont des règles intelligibles extraites d'une base de données transactionnelles, règles exprimant des associations entre items ou attributs dans une base de données.

L'une des applications typiques de l'extraction des règles d'association est l'analyse du panier de la ménagère (Market Basket Analysis) dont le principe est l'extraction d'associations entre produits sur les tickets de caisse de clients particuliers (on recherche des produits qui tendent à être achetés ensemble). Nous pouvons trouver comme exemple : une règle associative de la forme « 80% des clients qui achètent du fromage et du lait ont tendance à acheter des œufs». Fromage et lait constituent l'antécédent de cette règle, œuf est la conséquence et 80% est la confiance. Le but de cette technique est l'étude de ce que les clients achètent pour obtenir des informations pertinentes afin de comprendre leurs habitudes de consommation, agencer les rayons du magasin, organiser les promotions, personnaliser les catalogues et gérer les stocks etc.

Cette technique peut être appliquée à tout secteur d'activité pour lequel il est intéressant de rechercher des groupements potentiels de produits ou de services. Parmi les domaines d'application de la recherche de règles d'association nous pouvons citer comme exemple : le marketing, l'aide au diagnostic médical, les services de télécommunications, les services bancaires, l'analyse de données spatiales, la téléphonie, l'analyse des logs web... etc. On peut noter ici que dans nos expérimentations –chapitre 5- nous appliquons la recherche de règles

d'association sur un nouveau secteur d'activité, qui est le secteur pétrolier, et plus précisément sur les données de services aux puits accumulées dans la base de données ERP de l'Entreprise Nationale de Services aux Puits (ENSP).

II.7.1. Concepts et définitions

Nous présentons ici, les définitions de plusieurs concepts utilisés pour la recherche et l'extraction des règles d'association [43] et [91].

1. Transaction : On considère un ensemble I = {i_1, i_2,…, i_m} de m éléments (items) distincts. On appelle une transaction T le sous ensemble I'inclus dans I (T⊂I).

Dans une base de données transactionnelle D = {t_1, t_2, ..., t_n} de n transactions, chaque transaction t est identifiée par une clé unique TID.

2. Itemset : On appelle itemset un ensemble d'items i⊂I , le nombre d'items d'un itemset constitue sa longueur, un itemset contenant k items est appelé un k-itemset. Chaque instance d'une base de données est un itemset.

3. Règle d'association : est une implication de la forme A=>B où A et B sont inclus dans I et A ∩ B =Ø. A est appelé la condition ou l'antécédent et B la conclusion ou la conséquence. La pertinence d'une règle est mesurée par son support **S** et sa confiance **C**.

4. Support : est le nombre de transactions contenant à la fois tous les items de A et tous

$$Support(A \Rightarrow B) = \frac{\left|\{t \in D/(A \cup B) \subseteq t\}\right|}{|D|}$$

les items de B, par rapport au nombre total de transactions de D.

5. Confiance : est le nombre de transactions contenant à la fois tous les items de A et tous les items de B, par rapport au nombre de transactions contenant les items de A :

6. Itemset fréquent : un itemset dont le support est supérieur ou égal au seuil minimal de support (min support) définit par l'utilisateur est appelé itemset fréquent. On peut noter ici, plus le support est élevé, plus la règle est fréquente. Plus la confiance est élevée, moins il y a

$$Confiance(A \Rightarrow B) = \frac{\left|\{t \in D/(A \cup B) \subseteq t\}\right|}{\left|\{t \in D/A \subseteq t\}\right|}$$

de contre-exemples de la règle.

7. Autres mesures d'intérêt des règles d'association existent, nous citons : la conviction, le lift, le coefficient de corrélation, la JMeasure.

8. La recherche de règles d'association consiste à déterminer l'ensemble des règles dont le support et la confiance sont supérieurs à certains seuils fixés au départ par l'utilisateur.

9. L'extraction de règles d'association est décomposée en deux sous-problèmes :

- La recherche des ensembles fréquents d'items (les itemsets fréquents).
- La génération des règles d'association à partir de ces itemsets fréquents.

II.7.2. Etapes d'extraction des règles d'association

La recherche et l'extraction des règles d'association peuvent représenter par un processus itératif et interactif, qui se déroule en quatre phases successives [47] et [77] :

1. **Sélection et préparation des données :** Cette phase comporte deux étapes principales, **la sélection des données,** consiste à réduire la quantité des données en gardant seulement celles les plus pertinentes, et la **transformation de ces données** en contexte d'extraction, qui est représentée par un Triplet de type B= (O, I, R), dans lequel O et I sont des ensembles finis d'objets et d'items respectivement, R O ×I est une relation binaire entre les objets et les items. Un couple (o,i)ϵ R dénote que l'objet o ϵ O est en relation avec l'item i ϵ I. La sélection des données est nécessaire, elle permettra d'améliorer la qualité des règles d'association et la transformation des données est indispensable, elle permettra d'appliquer les algorithmes d'extraction des règles d'associations sur divers types de données.
2. **Découverte des ItemSets fréquents :** Une fois les données sont sélectionnées et préparées, il vient la phase de découverte des itemsets fréquents qui consiste à parcourir itérativement l'ensemble des itemsets. Durant chaque itération, un ensemble d'itemsets candidats est créé. Les supports de ces itemsets sont calculés et les itemsets non fréquents sont supprimés. Cette phase est couteuse en termes de temps d'exécution car le nombre d'itemsets fréquents dépend exponentiellement du nombre d'items manipulés, pour n items on a 2n itemsets potentiellement fréquents. De plus, le nombre d'itemsets fréquents est faible par rapport au nombre total d'itemsets. Il convient donc d'utiliser des approches plus complexes pour déterminer l'ensemble d'itemsets fréquents, on peut distinguer deux types d'algorithme [84] :
    - Des algorithmes qui proposent une réduction de l'espace de recherche (par exemple l'algorithme de base Apriori).
    - Des algorithmes qui proposent une réduction du nombre de balayages nécessaires du jeu de données (par exemple l'algorithme DIC).
3. **Génération des règles d'association** : Cette phase consiste à déduire les règles d'association en se basant sur l'itemsets fréquents obtenus. Elle va rechercher les règles d'association qui possèdent une confiance supérieure ou égale à une certaine confiance minimale définis par l'utilisateur. Pour chaque itemset, On va examiner les différentes combinaisons condition-conclusion possibles et garder la combinaison la plus pertinente celle qui répond au critère de seuil minimal de confiance.

   Soit k-itemset Fk fréquent, nous cherchons les règles de type : Fi→Fj de support et de confiance supérieure au seuil minimal fourni par l'utilisateur. Fi et Fj sont des itemsets de tailles respectives i et j, tels que : i+j = k, Fi $\subset$ Fk, Fj $\subset$ Fk et Fi $\cap$ Fj = $\emptyset$.

   On peut prendre l'exemple suivant, si XYZW et XY sont des itemsets fréquents, alors on peut engendrer la règle XY => WZ. Pour connaitre si cette règle est à maintenir ou à

rejeter, on calcule la confiance de XY => WZ qui n'est autre que la relation : conf = sup (XYZW) / sup (XY). Si conf ≥ minconf, alors la règle à retenir [84].

4. **Visualisation et Interprétation des résultats** : Cette phase permet à la visualisation des règles d'association extraites à l'utilisateur final ainsi que leur interprétation afin de déduire des connaissances utiles pour prendre des bonnes décisions dans son domaine d'activité.

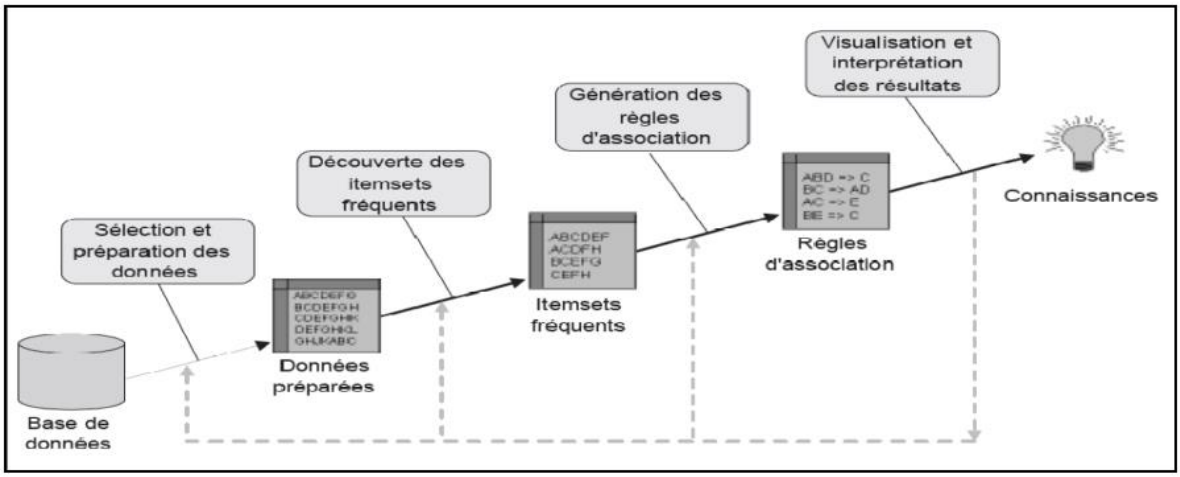


Figure II-4 : Etapes du processus d'extraction de règles d'association [84]

II.7.3. Algorithmes d'extraction de règles d'association

Il existe de nombreux algorithmes dans la littérature pour extraire les règles d'association à partir d'une base de données transactionnelle. L'algorithme le plus connu pour la recherche et l'extraction des règles d'association est l'algorithme Apriori [75]. La plupart des algorithmes existants sont basés sur Apriori. Celui-ci procède au traitement des données de façon itérative. Lors de la Kème itération, l'algorithme Apriori fait des calculs pour obtenir les K-itemsets fréquents. C'est-à-dire les itemsets de longueur K qui sont fréquents. A partir de l'itération (K+1) ème, il continue son fonctionnement en vue d'obtenir les (K+1)-itemsets fréquents et ainsi de suite.

II.7.3.1. L'algorithme Apriori

Dans la première itération de cet algorithme (Figure II 5 et Figure II 6), nous calculons le support des 1-items puis nous déterminons ceux qui sont fréquents. À partir de là, chaque itération suivante va utiliser la liste des ensembles d'éléments (itemsets) trouvés dans l'itération précédente, pour générer les nouveaux itemsets fréquents que nous appelons itemsets candidats. Cependant, durant chaque itération, les supports de ces itemsets candidats sont calculés. A la fin de chaque itération, nous déterminons l'ensemble des candidats fréquents à utiliser dans la prochaine itération. Ces traitements sont répétés jusqu'à ce que nous puissions avoir de nouveaux itemsets. Cet algorithme se base sur les deux propriétés suivantes pour limiter le nombre de candidats considérés lors de chaque itération, ce qui va réduire dynamiquement l'espace de recherche des itemsets fréquents.

■ **Propriété 1: Tous les sous-ensembles d'un itemset fréquent sont fréquents:** Elle limite le nombre d'itemsets candidats de tailles k +1 engendrés lors de la k +1 éme itération en effectuant une jointure conditionnelle des itemset fréquents de taille k, qui est découvris lors de l'itération antécédente.

■ **Propriété 2: Tous les sur-ensembles d'un itemset infréquentable sont infréquentables :** Elle permet d'éliminer un itemset candidat de taille k +1 lorsque au moins un de ses sous-ensembles de taille k ne figure pas dans les itemsets fréquentés, qui sont à découvrir lors de l'itération antécédente.

```
Algorithme Apriori
Debut
L1={Large 1-itemsets};
Pour (k=2; Lk-1 ≠∅; k ++) faire
/*Génération des Itemsets candidats*/
Apriori-Gen(Lk-1)
Ck={Ensemble des Itemsets candidats}
/*calcul du support*/
Pour toute transaction t∈D faire
Pour tout k-sous-ensemble s de t faire
si s ∈ Ck alors s.count++;FinSi
Finpour
/*Génération des Itemsets larges*/
Lk={c∈ Ck / c.count ≥ minsup};
Finpour
Finpour
Retourner ( Uk Lk = {Ensemble des Itemsets larges}
)
Fin
```

Figure II-5 : L'algorithme Apriori [75]

```
Fonction Apriori-Gen
/*Génération des Itemsets larges*/
Pour chaque Itemsets L1 ∈ Lk −1 faire
Pour chaque Itemsets L2 <> L1 de Lk −1
faire
Si (préfixe (L1)= préfixe (L2)) alors
Ck={ Ck U(c= L1 U L2)}}
FinSi
Finpour
Finpour
```

Figure II-6 : la procédure Apriori-Gen [75]

Cet algorithme va exécuter sur une base de données de type transactions D, dans l'intérêt est de découvrir les règles d'association qui satisfont un niveau de support (minsup) et de confiance minimum (minconf), qui sont entrés par l'utilisateur Data Miner.

Le support permet de mesurer la fréquence de l'association de la règle : Support = freq(condition et résultat) = D/M, tandis que D est le nombre de transactions de la base données où l'antécédent et la conclusion apparaissent et M le nombre total de transactions.

La confiance permet de mesurer la force de l'association de la règle : Confiance= freq(condition et résultat)/freq(condition) = D/C, tandis que « C » est le nombre de transactions où l'antécédent est déjà apparu.

Plusieurs variantes de cet algorithme ont été proposé pour réduire le temps d'extraction des itemsets fréquents, qu'ils effectuent plusieurs optimisations et structures de données permettant d'améliorer l'efficacité de l'algorithme de base Apriori. Nous présentons l'essentielles de ces algorithmes dans les sections suivantes.

II.7.3.2. DHP (Direct Hashing and Pruning)

DHP [85] est un léger changement d'Apriori qui utilise des tables d'hachage afin de réduire le nombre d'itemsets candidats produits. Cet algorithme fournit aussi une réduction de taille de la base de données, progressivement des itérations. À partir des deux premières itérations (gestion des 1-itemsets et des 2-itemsets), nous travaillons avec une table d'hachage sur les numéros des itemsets (c.à.d. joindre un numéro à chacun des itemsets possible pour l'itération en question). Pour toute case de la table d'hachage, nous jointons un compteur. A chaque fois qu'une procédure comporte l'un des itemsets associés (par l'activité d'hachage) à une case de la table d'hachage, nous incrémentons le compteur de cette case.

Nous conservons les itemsets adjoints aux cases de façon que le compteur soit suffisant. En illustrant, pour une case adjointe à 3 itemsets, nous conservons les itemsets si le compteur adjoint à la case a une valeur est supérieur de trois fois le seuil de support donné.

A partir de la seconde itération. Nous exécutons un pruning des difficultés de la base. A toute difficulté, si le nombre de bits de cette difficulté est supérieur de k, elle peut être nécessaire à la $k^{ème}$ itération, si non, nous la retirons.

II.7.3.3. AprioriTid

L'algorithme AprioriTID est introduit par Agrawel et Srikant [72] en 1994, dans le but de borner les accès directs à la base de données. En pratique les temps d'accès se manifestent beaucoup pénalisant plus que l'algorithme classique Apriori qui exécute une passe sur la base à tout échelon du treillis. C'est pourquoi, la totalité de la base est tenue dans la mémoire, et à tout échelon, les protocoles sont évoqués par les K-itemsets candidats qu'elle possède.

II.7.3.4. Partition

Cet algorithme [73] a été développé par Ashok Savasere, Edward Omiecinski et Shamkant Navathe afin de diminuer le nombre de voie de tri au niveau de données à deux tris uniquement. Ainsi que son nom le montre, le principe d'algorithme Partition existe pour partitionner la base de données en n partitions disjointes P1…Pn, la sélection du volume des partitions est en fonction du volume de la mémoire principale.

A toute partition Pi, les itemsets fréquents locaux sont créés lors d'un premier tri de P. Après, les itemsets fréquents de toute partition Pi sont associés pour avoir un groupe d'itemsets fréquents globaux. Cet algorithme contient deux étapes :

− Recherche des itemsets fréquents locaux sur chaque Pi,

− Dans la seconde étape un tri est réalisé ; les supports des itemsets fréquents globaux sont comptés par croisement des Tids et non par énumération également dans Apriori pour identifier les itemsets fréquents sont globaux.

II.7.3.5. Sampling

Cet algorithme [89] exécute la génération des itemsets sur un exemplaire de la base. Leurs supports sont après calculés sur toute la base. Et tout au long du parcours de la base.

Nous raffinons les résultats (la valeur des supports). Nous éliminons des itemsets qui se manifestent non fréquents. Cet algorithme est doté d'un exemplaire identifiant de la base.

II.7.3.6. DIC (Dynamic Itemsets Counting)

Les auteurs de [90] affirment que l'algorithme « DIC » réduit le nombre de tris des données, sachant que ledit algorithme est une méthode similaire à celui d'Apriori. Le jeu de données est partagé en sous-ensembles de volume donné, et pendant chaque itération, un sous-contexte (un sous-ensemble de problèmes) est choisi. Aussi, plusieurs ensembles d'itemsets candidats de volumes différents sont traités ensemble pendant chaque itération. Ce qui permet de minimiser le nombre total de tris du contexte. L'algorithme exécute une règle de fenêtrage de la base dont les auteurs ont composé des blocs de M problèmes. Ils ont aussi effectué un type d'itemsets : fréquent confirmé, fréquent potentiel, infréquentable confirmé, infréquentable potentiel.

Le trajet du premier bloc permet de trouver les fréquences des 1-itemsets. Quand ils ont franchi tout ce premier bloc, ils ont engendré par la suite les 2-itemsets candidats depuis des estimations de fréquences créées pour les 1-itemsets. Le trajet du second bloc a permis d'affiner les estimations de fréquences des 1-itemsets et de commencer l'estimation des fréquences des 2itemsets et ainsi de suite.

II.7.3.7. Eclat/ MaxEclat

Ces algorithmes sont introduits par M. J. Zaki en 1997 [88], ils sont formés sur l'élagage de la base de données en classes d'équivalences et distribution de la charge de travail sur tous les processeurs. En premier, il compte les items fréquents depuis de la base de données verticale tid-liste, après il compte les 2-itemsets. Ensuite il génère des sous-treillis, ces derniers sont alors traités isolément pour créer des itemsets fréquents maximaux.

Au contraire à Apriori, Eclat ne connait pas tous les itemsets fréquents à un niveau donné avant de considérer les candidats du niveau postérieur, ce qui minimise l'efficacité, puisque la propriété d'antimonotonicité n'est plus usée pour retrancher l'espace de recherche.

Cela demeure valable pour les petites bases de données, mais l'élagage reste pauvre et altère les performances lorsqu'il s'agit de traiter des bases de données de grande taille.

L'avantage de cette théorie comme le soulignent ses fondateurs, est qu'elle reste simplement parallelisable, notant que nous pouvons chercher les itemsets fréquents dans les différentes classes d'équivalents isolément. [88]

II.7.3.8. FP-Growth (Frequent-Pattern Growth)

C'est un algorithme [86] entièrement innovateur par rapport aux autres algorithmes de recherche de règles associatives souvent tous basés sur Apriori. L'algorithme utilise une structure de données dense nommée Frequent-Pattern tree et qui apporte une solution au de la fouille de motifs fréquents dans une grande base de données de transaction. En stockant l'ensemble des éléments fréquents de la base de transactions dans une structure dense, nous annulons la nécessiter de scanner de façon répétée la base de transactions. Encore, en triant les éléments dans la structure dense, nous pressons la fouille des motifs.

Un FP-tree est formé d'une racine nulle et d'un groupe de nœuds défini par l'élément indiqué. Un nœud est formé par : le nom de l'élément, le nombre d'occurrence de transaction où figure la portion de trajet jusqu'à ce nœud, un lien inter-nœud vers les autres occurrences du même élément figurant dans d'autres séquences transactionnelles. Une table d'en-tête pointe sur la première occurrence de tout élément. Le bienfait de cette représentation des données est qu'il suffit de poursuivre les liens inter-nœuds afin de savoir toutes les associations fréquentes où figure l'élément fréquent.

II.7.4. Avantages et inconvénients des règles d'association

II.7.4.1. Avantages

Les règles d'association représentent plusieurs avantages parmi lesquels nous citons : [76, 77]

- Application dans plusieurs secteurs d'activités, pour extraire les connaissances utiles, cachées dans les grandes bases de données.
- Simplicité de la méthode et des calculs, efficacité et facilité de compréhension.
- Facilité de l'interprétation par des résultats clairs.
- Aucune hypothèse préalable (Apprentissage non supervisé).
- Adaptation facile avec les séries temporelles.

II.7.4.2. Inconvénients

Malgré les avantages cités précédemment, les règles d'association peuvent représenter aussi des faiblesses qu'on peut résumer sur :

- La méthode est coûteuse en temps de calcul pour la recherche des ltemSets fréquents. [80].
- Parfois, elles génèrent une grande quantité des règles d'association [81].
- La production des règles triviales et inutiles n'apporte pas de nouvelles connaissances. [83]
- La difficulté d'évaluer les règles d'associations par des indices statiques ou par l'expert du domaine. [82]
- Méthode non efficace pour les articles rares. [78]

II.8. Data Mining distribué

La plupart des algorithmes séquentiels du Data Mining offrent un gain important pour l'extraction des connaissances, mais leurs performances se dégradent lorsque la taille de la

base de données est augmentée, ce qui influence négativement la vitesse des taches du data mining. Pour faire face à ces problèmes, le recours au parallélisme s'avère indispensable pour le développement étendu des algorithmes du Data Mining classique. Cela permet d'analyser les bases de données volumineuses de Téraoctet par des machines parallèles qui comportent des centaines de processeurs [79]. A partir de ces contraintes, le Data Mining distribué (DDM : Distributed Data Mining) est apparu pour répondre à une des perspectives suivantes :

La première perspective est la fouille de données principalement distribuées où les données doivent être fouillées dans leur site, à cause de plusieurs contraintes comme le coût de stockage, de communication, de calcul et de sécurité.

La deuxième est le besoin de préserver l'efficacité des algorithmes en présence d'une énorme quantité de données, dans ce cas, les données peuvent être partitionnées et distribuées sur des différents sites, afin d'effectuer le processus du Data Mining distribué en parallèle, et sur des portions de données moins volumineuses.

Autrement dit, le Data Mining distribué est le processus du Data Mining classique qui consiste à extraire une nouvelle connaissance à partir des sources de données partitionnées, distribuées sur des différents processeurs. Chaque processeur applique un algorithme du Data Mining sur ses données locales. Les résultats sont ensuite combinés avec le minimum d'interaction entre les sites de données [79]. En plus de cet aspect distribué qu'il faut prendre en considération, le Data Mining distribué ainsi que celui le classique sont toujours face à deux défis majeurs : augmentation du taux de précision et diminution du temps de calcul.

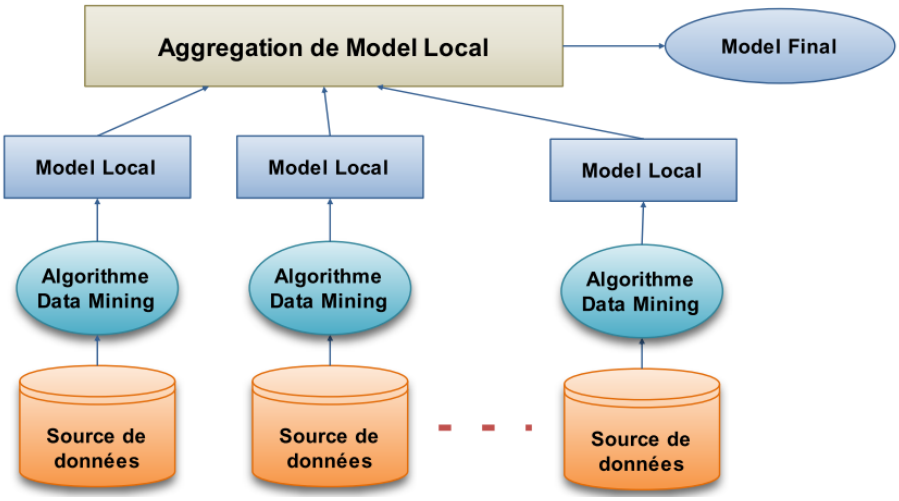


Figure II-7 : Architecture générale du Data Mining distribué [79].

II.8.1. Concepts du parallélisme

Le Data Mining distribué se base essentiellement sur les concepts de parallélisme, ce qui permet de traiter efficacement l'énorme quantité de données dans un environnement parallèle. Dans ce qui suit nous présentons les différents concepts liés au parallélisme tels que [92] : les architectures des parallélismes, les paradigmes de parallélisme, la stratégie d'équilibrage de charge et la fragmentation des données :

II.8.1.1. Architectures du calcul parallèle

Les architectures du calcul parallèle se basent principalement sur les plateformes matérielles utilisées pour le parallélisme, qui sont divisées en deux types : (1) Systèmes à Mémoire Partagée(SMP) et (2) Réseaux de stations et grille de calcul. La table suivante résume les différences entre les deux architectures [58] [79] :

Tableau II-1 : Architectures du calcul parallèle

Réseaux de stations et grille de calcul	Systèmes à mémoire partagée(SMP)
• Utilisation simultanée de plusieurs stations de travail.	• Utilisation simultanée de plusieurs processeurs.
• Absence de mémoire commune.	• Existence d'une mémoire commune ou partagée par les processeurs.
• Présence d'un réseau d'interconnexion lent par rapport à la puissance des machines.	• Disponibilité d'un réseau d'interconnexion interne rapide.
• Hétérogénéité d'un système composé d'une grappe de stations ou d'une grappe de grappes.	• Homogénéité d'un système d'exploitation unique.
• La synchronisation par le passage de messages.	• La synchronisation se fait via les verrous.
• les E/S parallèles sont libres.	• Problématique pour les systèmes SMP, qui sérialisent généralement les E/S.
• Bonne décomposition de données entre les nœuds, et minimiser la communication.	• Bonne localité des données, c.-à-d. optimiser les accès à l'antémémoire locale, et d'éviter ou réduire les faux partages.

Actuellement, l'accès est non uniforme à la mémoire pour les machines de type hybride ou hiérarchique (Ex, le cluster de SMPs), les paramètres d'optimisation tirent de deux architectures à la fois [59][60].

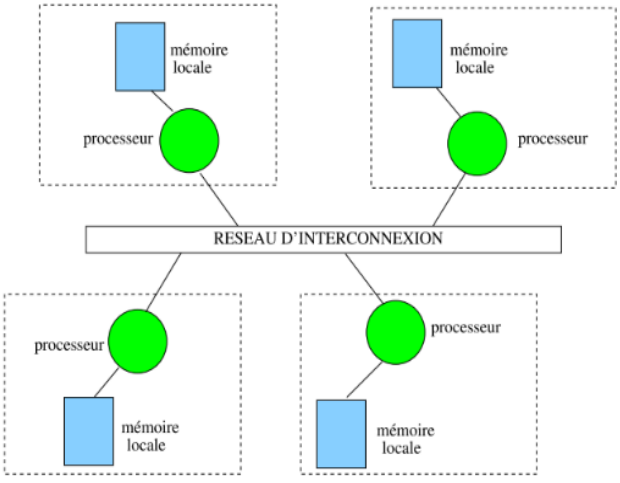


Figure II-8 : Architecture de type réseau de situations et grille de calcul [58]

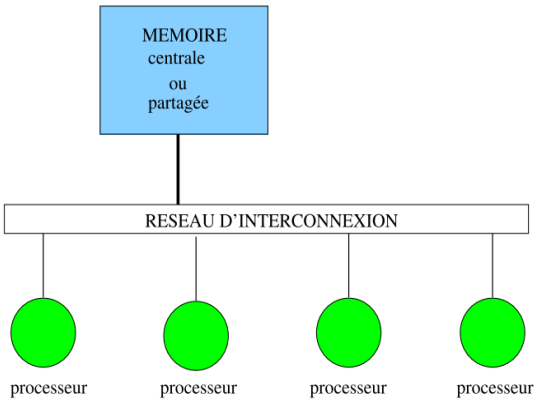


Figure II-9 : Architecture SMP [58]

II.8.1.2. Modèles du parallélisme

Les algorithmes à forte intensité exploitent les modèles du parallélisme, pour améliorer leurs performances et accroître l'évolutivité. Les deux principaux modèles du parallélisme sont : (1) le parallélisme de données et (2) le parallélisme des tâches. Le parallélisme des données partage la base de données entre les processeurs. Chaque processeur effectue le même calcul sur sa partition locale de la base de données. Le parallélisme des tâches distribue la charge de travail impliquée dans l'exploration de données sur plusieurs processeurs où les processeurs effectuent des calculs différents de manière indépendante. Le parallélisme hybride est également possible, il combine les deux à la fois le parallélisme des tâches et des données.

II.8.2. Stratégie d'équilibrage de charge

L'implémentation des tâches en parallèle souffre de problèmes d'équilibrage de charge causés par la distribution inégale de la charge de travail entre les différents processeurs. Dans la littérature il existe de types d'équilibrage de charge statique et dynamique. L'équilibrage de charge statique est fondé sur le partage des tâches entre les processeurs au début du traitement, en utilisant une fonction de coût heuristique, et il n'y a pas des traitements ultérieurs pour corriger les déséquilibres de charge qui résultent de la nature dynamique des algorithmes. L'équilibrage de charge dynamique est fondé par réaffectation dynamique des tâches des processeurs lourdement chargés à ceux faiblement chargés. Ce type d'équilibrage de charge dynamique expose des coûts supplémentaires pour les mouvements des tâches et de données, mais il est bénéfique si le déséquilibre de charge est important et si la charge est avec le temps. L'équilibrage de charge dynamique est essentiellement important dans les environnements multiutilisateurs avec des charges transitoires et dans les plates-formes hétérogènes. La plupart des algorithmes du Data Mining parallèles existants actuellement utilisent une approche d'équilibrage de charge statique qui est faite de la répartition de la base de données au début du traitement entre les sites disponibles.

II.8.3. Fragmentation de données

Dans le contexte du parallélisme, la base de données peut être partitionnée horizontalement (par ligne) ou verticalement (par colonne) dont le but d'augmenter les

performances pour la distribution de données entre différents sites disponibles. Dans la fragmentation horizontale, la base de données est distribuée par instances, tandis que dans la fragmentation verticale, les données sont distribuées par attributs. Plusieurs algorithmes de Data Mining utilisent une fragmentation horizontale (Figure I-5) de la base de données, où ils stockent comme unité, chaque transaction (tid), ainsi que les valeurs d'attribut pour cette transaction. D'autres algorithmes utilisent une fragmentation verticale (Figure I6), où ils associent à chaque attribut une liste de tous les tids (appelé Tidlist) contenant l'item et la valeur d'attribut correspondant à cette transaction. Certains algorithmes fonctionnent plus efficace en utilisant un format horizontal, tandis que d'autres sont plus efficaces en utilisant un format vertical.

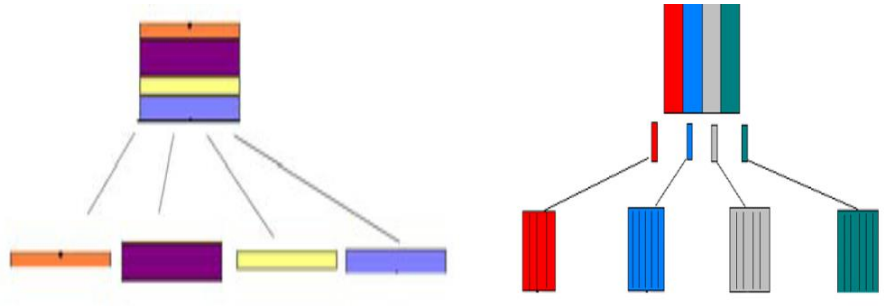


Figure II-10 : Fragmentation horizontale Vs Fragmentation verticale [79].

II.8.4. Défis de Data Mining distribué

Le développement du Data Mining dans les environnements parallèles et distribués, est confronté à plusieurs défis qui sont les suivants [60] :

• Distribution de données : l'un des avantages de Data Mining distribué est que chaque nœud peut potentiellement travailler avec un sous-ensemble de taille réduite de la base de données totale. Un algorithme parallèle dans un environnement distribué doit effectivement distribuer des données pour permettre à chaque nœud de faire des progrès indépendants avec sa vue incomplète de la base de données entière. ;

• Minimisation des E / S : même avec une bonne distribution des données, les algorithmes parallèles d'exploration de données doivent essayer de minimiser les opérations de E / S qu'ils effectuent sur la base de données.

• Équilibrage de charge : pour maximiser l'effet / l'efficacité du parallélisme, chaque poste de travail doit avoir approximativement la même quantité de travail à effectuer. Bien qu'une bonne distribution initiale des données puisse aider à fournir un équilibrage de charge, avec certains algorithmes, une redistribution périodique des données est requise pour obtenir un bon équilibrage de charge global.

• Éviter les doublons : idéalement, aucun poste de travail ne devrait effectuer de travail redondant (travail déjà effectué par un autre nœud).

• Minimiser la communication : un algorithme d'exploration de données parallèle idéal permet à tous les postes de travail de fonctionner de manière asynchrone, sans devoir se bloquer fréquemment pour les barrières globales ou pour les retards de communication.

• Optimisation de la localité : comme pour toute programmation de performance, les algorithmes d'exploration de données parallèles à haute performance doivent être conçus de manière à accroître le potentiel de performance du matériel. Cela implique de maximiser la localité pour un bon comportement de cache, en utilisant autant que possible la bande passante de la mémoire de la machine, etc.

La réalisation de tous les objectifs ci-dessus dans un algorithme est presque impossible, car il existe des compromis entre plusieurs points ci-dessus. Les algorithmes existants pour l'exploration de données parallèles tentent d'obtenir un équilibre optimal entre ces facteurs.

II.9. Techniques utilisées dans le Data Mining Distribué

Trois techniques sont souvent utilisées dans un processus de Data Mining distribué telles que la classification distribuée, Le clustering distribué et les règles d'association distribuées :

II.9.1. La Classification distribuée

Elle permet la classification d'un ensemble de données sur plusieurs machines distribuées. Il existe plusieurs techniques pour effectuer la classification distribuée, à savoir : les arbres de décision, réseaux de neurones et les algorithmes génétiques.

II.9.2. Le Clustering Distribué

Cette technique est utilisée pour mettre en commun la puissance de plusieurs ordinateurs (clusters), elle comprend au moins deux sites appelés « nœuds ». Le clustering Distribué travaille sur des données de nature distribuée avec un paradigme de distribution des tâches comme MapReduce. Ce dernier est produit par Google Corp, comme un modèle de programmation parallèle adapté au traitement des données massives (c.-à-d. plus de 10 GB). L'implémentation de ce type de solution est coûteuse car elle nécessite des infrastructures très puissantes à savoir : une grande capacité de stockage en format des fichiers spécifique comme HDFS (Hadoop Distributed File System), réseau de haut débit et des unités de traitement très performantes. D'ailleurs, le HDFS est le système de fichier distribué de Hadoop Apache, qui est très adapté au Big Data. Il stocke les données sous forme de fichiers et les découpe en blocs de 64 MB pour être répliqués sur différents serveurs composant le cluster. Ces caractéristiques permettent donc de paralléliser les traitements sur ces données, d'être résistant aux pannes (grâce à la réplication) et d'être facilement évolutif. C'est l'avantage de HDFS par rapport à un RDBMS (Relational DataBase Management System) classique.

II.9.3. Les Règles d'Association Distribuées

Elles sont utilisées pour trouver des relations entre des items dans un ensemble de données non centralisées. Dans la section suivante, nous somme intéressé à présenter les algorithmes de règles d'association distribuées, qui représentent des fondements théoriques pour notre approche présentée dans le chapitre V.

II.10. Algorithmes de règles d'association parallèles et distribués

Des nombreux algorithmes d'extraction de règles d'association parallèles et distribués ont été graduellement adaptées aux systèmes parallèles afin d'améliorer le temps de réponse et tirer profit de la capacité de stockage étendue qu'offrent les systèmes distribués et parallèles. Différentes approches ont été proposées, elles se divisent en deux catégories, parallélisme de données et parallélisme de tâches. Les deux paradigmes diffèrent par le fait que l'ensemble de candidats soit distribué ou non à travers les processeurs. Dans le paradigme de parallélisme de données, chaque nœud compte le même ensemble de candidats alors que dans le paradigme de parallélisme de tâches, l'ensemble des candidats est divisé et distribué à travers les processeurs, et chaque nœud compte un ensemble différent de candidats. Ces parallélisations nécessitent une nouvelle propriété (Propriété 3) en plus des deux propriétés présentées précédemment dans l'extraction séquentielle des règles d'association :

Propriété 3 : Pour qu'un itemset soit globalement fréquent il faut qu'il soit localement fréquent sur au moins dans un site.

II.10.1. Algorithmes basés sur le parallélisme de données

Dans ce type de paradigme, chaque nœud dénombre le même ensemble de candidats. Les ensembles sont dupliqués sur tous les processeurs et la base de données est distribuée à travers ces processeurs. Chaque processeur est responsable du calcul des supports locaux de tous les candidats qui sont des supports dans sa propre partition de la base de données. Tous les processeurs calculent ensuite les supports globaux des candidats qui sont les supports totaux des candidats dans la base de données entière par échange des supports locaux (la réduction globale).Par la suite, les itemsets fréquents sont calculés par chaque processeur indépendamment. Parmi ces algorithmes, on peut citer Count Distribution, Fast Distributed Mining…etc

II.10.1.1. Count Distribution (CD)

Proposé par R.Agrawal et J.C Shafer en 1996, cet algorithme [93] est basé sur l'algorithme Apriori et sur le parallélisme de données, on effectue un découpage horizontal des données (répartition des instances sur les processeurs). Chaque processeur compte les supports des mêmes itemsets candidats. A chaque itération, les supports locaux de chaque itemset sont calculés, puis une synchronisation permet de sommer ces supports locaux, de manière à prendre les décisions sur des supports globaux. Cette synchronisation de supports à lieu entre chaque itération.

II.10.1.2. Fast Distributed Mining (FDM)

Cet algorithme [94] est basé sur l'algorithme Apriori, il a été proposé par Daving Cheung pour l'extraction des règles d'association. Ce dernier propose ainsi de nouvelles techniques pour réduire le nombre des candidats considérés pour le calcul du support. Après avoir généré l'ensemble global de candidats CG(k), il faut échanger les supports de cet ensemble entre les sites pour trouver les itemsets globalement fréquents. Toutefois, il est observé que certains ensembles de candidats dans CG(k) peuvent être élagués en utilisant des informations locales avant que l'échange des supports commence. L'élagage local consiste

alors à supprimer l'élément X de l'ensemble de candidats sur le site si X n'est pas localement fréquent par rapport à sa base de données locale.

Une fois que l'élagage local est effectué, chaque site diffuse les supports des candidats restant aux autres sites. A la fin de chaque itération chaque site somme ces supports locaux pour générer les itemsets globalement fréquents par rapport à la base de données globale, cette technique est appelée l'élagage global. L'algorithme FDM minimise ainsi l'ensemble des candidats générés en se basant sur ces deux techniques.

II.10.1.3. Distributed Mining of Association rules (DMA)

L'algorithme DMA [95] est une parallélisation de l'algorithme Apriori. Il utilise une technique d'élagage au niveau de chaque site et ne garde que les itemsets fréquents. Les supports locaux des itemsets fréquents sur chaque site sont employés pour décider si un itemset fréquent est lourd (localement fréquent dans une partition et globalement fréquent dans la base de données entière).

DMA introduit une nouvelle technique d'optimisation de la communication afin de réduire la taille des messages et éviter leur duplication. Au lieu de diffuser les supports locaux de tous les candidats comme dans le cas de CD, DMA associé à chaque itemset un site de vote qui est le responsable de collecte de ses supports locaux.

Dans cet algorithme, pour chaque candidat X, son site de vote (polling site) est responsable d'envoyer les demandes (polling request) pour collecter les supports locaux, et déterminer si le candidat X est fréquent. Puisqu'il y a un seul site de vote pour chaque candidat X, le nombre de messages échangés pour calculer le support global de X est n. Dans la keme itération, après la fin de la phase d'élagage local dans le site S.

II.10.1.4. Optimized Distributed Association Mining (ODAM)

ODAM [96] suit le paradigme de CD et de FDM. Il partitionne la base de données horizontalement sur les sites distribués. ODAM calcule d'abord les supports des 1-itemsets sur chaque site de la même manière que l'algorithme séquentiel Apriori, il diffuse ensuite cet ensemble d'itemsets aux autres sites et identifie les 1-itemsets globalement fréquent. Par la suite, chaque site génère les 2-itemsets candidats à partir de cet ensemble et calcule leurs supports. En même temps, ODAM élimine tous les 1-itemsets non fréquents globalement de chaque transaction et insère la nouvelle transaction dans la mémoire. Après la génération des supports des 2-itemsets candidats à chaque site, ODAM génère les 2-itemsets globalement fréquents. Il recommence alors sur les nouvelles transactions en mémoire principale et il génère les supports des itemsets candidats de longueur successive puis les itemsets fréquents globaux correspondant cette longueur en diffusant les supports des itemsets candidat après chaque passage.

ODAM propose une optimisation dans le schéma de communication en réduisant au minimum les échanges de messages en envoyant des comptes de support d'itemsets des candidats à un emplacement simple appelé le récepteur. Le récepteur annonce les itemsets globalement fréquent de nouveau aux emplacements distribués.

Pour calculer efficacement les supports des candidats générés, ODAM élimine tous les ite s non fréquents après la première itération et stocke les nouvelles transactions dans la mémoire. Cette technique permet non seulement de réduire la taille moyenne des transactions mais également réduit la taille de l'ensemble de données de manière significative, ainsi on peut accumuler plus de transactions dans la mémoire. Le nombre d'items dans l'ensemble de données pourrait être grand, mais seulement quelques-uns satisferont le seuil de support. En plus, le nombre des itemsets non fréquents augmente proportionnellement pour un seuil de support plus élevé.

II.10.2. Algorithmes basés sur le Parallélisme de taches

Dans ce type de paradigme, l'ensemble de candidats est partitionné et distribué à travers les processeurs ainsi que la base de données. Chaque processeur est responsable de maintenir seulement les supports globaux d'un sous-ensemble de candidats. Cette approche exige deux passages de communication dans chaque itération. Dans le premier, chaque processeur envoie sa partition de la base de données à tous les autres processeurs. Dans le second passage, chaque processeur diffuse les itemsets fréquents qu'il a trouvés à tous les autres processeurs pour calculer les candidats de l'itération suivante. Plusieurs variantes du parallélisme de tache ont été proposées, elles diffèrent dans la façon de diviser la base de données et les candidats :

II.10.2.1. Data Distribution (DD)

Il s'agit encore d'un algorithme [93] issu de l'algorithme Apriori et proposé par R. Agrawal et J.C. Shafer en 1996. Il se diffère de Count Distribution par le fait que les processeurs ne distribuent pas le travail par portion de base de données mais le partage se fait par des candidats. Les supports des itemsets sont toujours calculés de façon locale, à chaque itération. Puis, entre deux itérations on effectue une diffusion des partitions locales de données. Dans cet algorithme DD, chaque processeur est responsable de calculer les supports d'un ensemble de candidats qui est différent des ensembles de candidats assignés aux autres processeurs.

II.10.2.2. Intelligent Data Distribution (IDD)

Cet algorithme [97] a été proposé pour résoudre les problèmes de l'algorithme Data Distribution. Premièrement, Il adopte une architecture en anneau pour améliorer les performances et réduire les couts de communication, c'est-à-dire il emploie la communication asynchrone de point en point entre les voisins dans l'anneau au lieu du broadcast.

Deuxièment, IDD résout le problème du travail redondant dans DD par le partitionnement intelligent de l'ensemble de candidats. Afin d'éliminer le calcul redondant dû au partitionnement des itemsets, IDD trouve une manière rapide pour vérifier si les transactions données peuvent potentiellement contenir les candidats stockés dans chaque processeur. Ceci n'est pas possible par le partitionnement aléatoire de l'ensemble de candidats global Ck. IDD partitionne les candidats à travers les processeurs en se basant sur le premier item des candidats, c'est-à-dire les candidats avec le même item préfix (premier item) seront mis dans la même partition. Donc, avant le traitement d'une transaction chaque processeur doit s'assurer qu'elle contient les préfixes appropriés qui sont assignés à ce processeur. Sinon, la transaction peut être rejetée.

La base de données entière est toujours communiquée, mais une transaction ne pourrait pas être traitée si elle ne contient pas les itemsets appropriés. Ce qui réduit le calcul redondant dans DD (Dans DD, chaque processeur doit vérifier tous les sous-ensembles de chaque transaction).Pour réaliser un équilibre de charge dans la distribution des candidats, il calcule d'abord pour chaque item le nombre de candidats commençant par cet item et il assigne les itemsets aux partitions des candidats de telle sorte que le nombre de candidats dans chaque partition soit le même.[98]

II.10.2.3. Hash Partioned Apriori (HPA)

L'algorithme HPA [99] est conçu dans le même esprit que l'algorithme IDD, il utilise la fonction de hachage pour partitionner les itemsets candidats entre les différents processeurs. Ce qui élimine le broadcast de toutes les transactions et permet de réduire la charge de travail de manière significative.

II.10.2.4. D'autres types de parallélisme

Il existe d'autres algorithmes qui ne peuvent pas être classifiés dans les deux paradigmes. Ils ont des particularités distinctes. Ces algorithmes incluent :

II.10.2.5. Hybrid Distribution (HD)

L'algorithme Hybrid Distribution [97] combine Count Distribution et Intelligent Data Distribution. Il partitionne les P processeurs en G groupe de tailles égales, où chaque groupe est considéré comme un super-processeur contenant P/G processeurs. L'algorithme IDD est utilisé au sein des groupes et l'algorithme CD entre les groupes. La base de données est partitionnée horizontalement entre les super processeurs G, et les candidats sont repartis entre les P/G processeurs dans un groupe. En outre, la distribution hybride ajuste le nombre de groupes de manière dynamique à chaque passage.

II.10.2.6. ParEclat

Cet algorithme [100] repose sur le découpage de la base en classe d'équivalence, où chaque processeur calcule tous les itemsets fréquents d'une classe d'équivalence. Il découple les dépendances entre les processeurs, dont chacun peut procéder indépendamment, il n'a y pas une synchronisation coûteuse à la fin de chaque itération. En outre, Eclat utilise une disposition verticale de la base de données les Tid-listes. Notons que les tid-listes élaguent automatiquement les transactions non pertinentes, quand la taille de l'itemset augmente, la taille de tid-liste diminue, résultant par des intersections rapides. Ainsi la base de données locale est parcourue une fois seulement.

II.11. Discussion et étude des algorithmes distribués présentés

Après avoir présenté les algorithmes d'extraction de règles d'association dans un environnement parallèle et distribué. Nous observons que la majorité de ces algorithmes est basée sur l'algorithme Apriori. Selon les paradigmes de parallélismes et les architectures du calcul parallèle citées dans la section II.8.1, nous catégorisons les algorithmes ARM distribués en deux classes :

II.11.1. Parallélisme de taches

Les algorithmes ARM distribués de ce type de parallélisme partitionnent et distribuent les candidats et la base de données entre les différents processeurs. Chaque processeur envoie sa base de données locale à tous les autres processeurs, en même temps il calcule les supports de sa portion de l'ensemble des itemsets candidats sur sa base de données locale et sur les partitions de la base de données envoyées par les autres processeurs dans chaque itération. Ces algorithmes distribués impliquent une quantité non négligeable de communication entre tous les processeurs pour échanger leurs partitions locales à chaque étape. Ils souffrent ainsi d'un fort coût de communication dû à cette diffusion, surtout si la base de données est très grande (comporte beaucoup d'instances) comme c'est généralement le cas.

Ce type d'algorithme est adapté au parallélisme à mémoire partagée et à réseau d'interconnexion rapide, à cause des échanges nécessaires aux partitions de données entre les nœuds. Il est non satisfaisant sur une architecture parallèle de type réseau de stations de travail, les communications y étant trop pénalisantes.

II.11.2. Parallélisme de données

Dans ce type d'algorithme, chaque site calcule le même ensemble de candidats qui sont dupliqués sur tous les processeurs et la base de données est distribuée à travers ces processeurs. Il doit échanger les supports calculés sur les différents processeurs, d'où énormément de communications que l'on peut considérer assez courtes ; dans le meilleur des cas, on ne communique que des entiers (les supports).

Ici encore, ce type d'algorithme est adapté au parallélisme à mémoire partagée et à réseau d'interconnexion rapide, à cause des échanges d'informations (supports) nécessaires. La solution ne peut pas être utilisée de manière satisfaisante sur une architecture parallèle de type réseau de stations de travail, puisque les communications nécessaires entre les itérations sont bloquantes (pour passer aux K+1-itemsets, il faut recevoir les supports des k-itemsets de tous les sites).

Les algorithmes présentés dans les deux types de parallélisme s'inscrivent dans une mémoire logique partagée et se basent toujours sur le problème type de l'analyse du panier de la ménagère. Sur ce type de problèmes (données de consommation simple), généralement, la taille des itemsets fréquents reste limitée, (le plus grand nombre k pour lequel on génère les k−itemsets reste faible), on n'a donc peu d'itérations à faire. Sur des données de services aux puits de la base de données ERP (comme celles que l'on souhaite traiter), il serait intéressant de ne pas se limiter pour les jeux de données ; la longueur, la largeur de la base ainsi que le nombre moyen d'items par instance ne doivent pas être soumis à des bornes supérieures trop faibles à cause de la nature du domaine traité. De ce fait, une vision par mémoire distribuée pourrait permettre d'accroitre la performance des traitements par : (1) l'utilisation simultanée de plusieurs stations de travail ;(2) La synchronisation par le passage de messages ; (3) Bonne décomposition de données entre les nœuds ; et (4) minimiser la communication.

II.12. Conclusion

Le Data Mining est une technologie puissante qui offre aux entreprises la possibilité de se focaliser sur les connaissances les plus importantes dans leurs bases de données afin de prédire les futurs tendances et actions, permettant de prendre les bonnes décisions aux bons moments.

Dans ce chapitre nous avons présenté la notion du Data Mining. Plusieurs définitions ont été étudiées selon plusieurs points de vue. Après, nous nous intéresserons à la technique du Clustering de données, une des techniques les plus utilisées dans le domaine du data mining. Ensuite, un intérêt particulier est donné à l'extraction de règles d'association qui constitue l'un des champs de recherche les plus actifs dans ce domaine. Cependant, cette technique souffre de dégradation de performance lorsque la taille de la base de données est augmentée, ce qui influence négativement sur la vitesse des taches du Data Mining.

Pour remédier à ces problèmes, de nombreux algorithmes parallèles et distribués ont été proposés dans la littérature pour l'extraction distribuée des règles d'associations. L'objectif de ces algorithmes est de réduire le coût de communication qui constitue un facteur important pour mesurer leurs performances. Néanmoins, la plupart de ces algorithmes présentent habituellement un nombre élevé de balayage des données et des étapes multiples des synchronisations et de communication qui dégrade leurs performances.

Dans le chapitre suivant nous allons présenter quelques approches de Data Mining distribuée à base du système multi-agents (SMA), ce dernier constitue aujourd'hui une nouvelle technologie pour la conception et le développement de systèmes complexes. Le SMA nous permettra de résoudre la problématique citée précédemment pour l'extraction de règles d'association distribuée et du clustering de données. Il permet l'exécution autonome et asynchrone de plusieurs taches de Data Mining simultanément, sur une très grande quantité de données distribuées, ce qui permet d'améliorer le temps de réponse et de minimiser le coût de communication entre les différentes stations du travail.

Chapitre III : Etat de l'art sur les systèmes de Data Mining à base d'agents

III.1. Introduction

De nos jours, les systèmes récents sont développés dans des environnements ouverts, hétérogènes et sont souvent distribués sur plusieurs machines. Ils interagissent et communiquent entre eux et s'exécutent indépendamment les uns des autres. Ils sont généralement divisés en sous-systèmes (indépendants) dont chacun d'eux exécute une partie du travail pour un but commun. Cette nouvelle tendance a donné lieu à la naissance des systèmes multi-agents.

Les systèmes multi-agents sont apparus comme un sous-domaine d'Intelligence Artificielle Distribuée (IAD) pour traiter des problèmes complexes que l'IA classique a échoué à résoudre. Ainsi, ils offrent une approche privilégiée afin de remédier à plusieurs types de problèmes d'une manière distribuée et de proposer des solutions modulaires et robustes. Ces systèmes ont dominé le domaine de l'intelligence artificielle, celui des systèmes informatiques distribués, de la robotique et du génie logiciel. Ils s'intéressent aux comportements collectifs générés par les interactions entre plusieurs entités autonomes et flexibles appelées agents [102]. Par ailleurs, la complexité du processus de data Mining et ses divers algorithmes ainsi que la diversité des profils d'utilisateurs ont motivé les chercheurs d'avoir recours aux systèmes multi-agents pour développer des plateformes Data Mining plus avancées, et des systèmes plus intelligents et plus robustes [103].

Dans ce chapitre nous abordons la contribution des systèmes multi-agents dans le domaine de Data Mining. Avant de commencer, il serait intéressant de comprendre quelques concepts sur les systèmes multi agents. Ensuite, nous montrons la motivation et les avantages de la contribution du système multi-agents dans le Data Mining. Nous présentons par la suite une étude des travaux de Data Mining qui exploitent le paradigme agent quant à la tâche du Clustering et les règles d'association. Enfin, nous clôturons ce chapitre par la présentation des approches existantes qui consistent à intégrer la technologie du Data Mining avec le système ERP dont l'objectif est de mieux situer nos propositions.

III.2. Motivation de couplage du Data Mining avec les SMA

Le domaine d'application de Data Mining (DM) et ses techniques ont été grandement développés au cours de ces dernières années. La technologie d'exploration de données est un moyen d'identification des modèles destinés au traitement de grandes quantités de données. Cependant, ces techniques sont appliquées généralement dans des environnements complexes pour faire face aux changements pouvant affecter l'ensemble de la performance du système. Étant donné que les systèmes de DM sont composés d'un certain nombre de tâches discrètes et néanmoins dépendantes, ils peuvent être considérés comme des réseaux d'unités collaboratives, mais autonomes. D'ailleurs, ces systèmes régulent, contrôlent et organisent toutes les activités distribuées impliquées tout au long du processus de l'extraction de

connaissances tel que : le nettoyage de données, la transformation et la réduction des données, l'application des algorithmes de DM et l'évaluation des résultats. [104]

En outre, la littérature de recherche sur les architectures des systèmes multi-agents a prouvé que de tels types de problèmes nécessitant la synergie d'un certain nombre d'éléments distribués et autonomes, peuvent être remédiés efficacement grâce aux systèmes multi-agents [105]. C'est pour cette raison que la technologie multi-agents est utilisée comme un outil puissant pour le développement des systèmes de Data Mining [106, 107 et 108]. Ces systèmes traitent souvent des applications complexes qui nécessitent la résolution des problèmes distribués dont le comportement individuel et collectif des agents dépend des données observées. Dans ce contexte, les systèmes multi-agents possèdent deux méthodes de résolution de problèmes, soit par la distribution de données entre plusieurs sites, ou bien par la division de la complexité de système en sous-systèmes (indépendants) [109]. L'utilisation du SMA est primordiale dans notre travail, puisqu'il permet de créer des modèles de connaissances clairs et robustes. De ce fait, l'intégration des agents coopératifs au sein du processus de Data Mining est essentielle pour l'extraction efficace des connaissances utiles et, en même temps, cachées dans une grande base de données centrale du progiciel ERP. Dans la section suivante, nous allons mettre en évidence quelques notions de l'agent et des systèmes multi-agents.

III.3. Généralité sur les Systèmes Multi-Agents (SMA)

Aujourd'hui, les systèmes multi-agents sont devenus un nouveau paradigme qui fournit une architecture très appropriée pour la conception et la mise en œuvre de plusieurs systèmes dans divers domaines comme : le domaine de la simulation et de la vie artificielle, la robotique, le traitement d'images, les bases de données et les bases de connaissances distribuées coopératives, les applications distribuées comme le Web, etc. Les SMAs sont principalement issus de l'intelligence artificielle distribuée (IAD) qui est une branche de l'intelligence artificielle classique. Ils offrent une approche privilégiée s'intéressant aux systèmes composés de nombreux éléments qui interagissent fortement entre eux dans un environnement souvent décentralisé. La thématique du système multi-agents se propose pour l'étude des comportements collectifs ainsi sur la répartition de l'intelligence entre différents agents plus ou moins autonomes et capables de s'organiser. Dans un premier temps nous allons définir et décrire le concept d'agent.

III.3.1. Concept d'agent

Dans la littérature des SMAs, il n'existe pas une définition universelle de l'agent parce que ce concept peut être vu de plusieurs angles. Chaque définition est généralement influencée par une vision particulière de l'agent ou par le contexte de l'utilisation. Toutefois, il est possible d'identifier quelques caractéristiques qu'un agent doit posséder pour qu'il puisse être qualifié d'agent. Deux définitions fréquemment utilisées dans la littérature de système multi-agents sont abordées :

La première définition est celle de Jacques Ferber [113], qui est très acceptée dans la communauté SMA. Il définit un agent comme étant *« une entité physique ou virtuelle qui :*

- ✓ *est capable d'agir dans un environnement,*
- ✓ *peut communiquer avec d'autres agents,*
- ✓ *possède des ressources propres et des tendances (des objectifs individuels ou d'une fonction de satisfaction),*
- ✓ *est capable de percevoir son environnement,*
- ✓ *ne dispose qu'une représentation partielle de cet environnement (et éventuellement aucune),*
- ✓ *possède des compétences et offre des services,*
- ✓ *peut éventuellement se reproduire,*
- ✓ *tend par son comportement à satisfaire ses objectifs.»*

Selon la définition de Jennings et Wooldridge, [114] « *Un agent est un système informatique, situé dans un environnement, et qui agit sur cet environnement de façon autonome pour atteindre les objectifs pour lesquels il a été conçu.* »

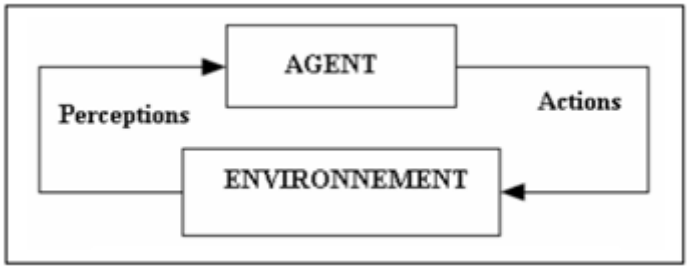


Figure III-1 : Action de l'agent sur l'environnement [118]

A partir des deux définitions précédentes, nous citons les principales caractéristiques d'un agent :

- L'autonomie : est la faculté d'avoir ou non le contrôle de son comportement sans l'intervention d'autres agents ou d'êtres humains.
- La réactivité : l'agent doit être capable d'élaborer des réponses aux changements de son environnement dans les temps requis.
- La proactivité : l'agent prend des initiatives, et choisit des actions en fonction de ses objectifs et des connaissances qu'il possède et ce aux moments opportuns.
- La sociabilité : l'agent interagit avec d'autres agents pour satisfaire les tâches qui lui sont confiées, coopère et résout des conflits.

III.3.2. Les différents types d'agent

Dans la littérature des SMAs, il existe plusieurs classifications des agents. Conformément aux références [115] et [116], la taxonomie la plus complète des agents comprend :

III.3.2.1. Agents d'interface

Les agents d'interface proviennent des domaines de l'intelligence artificielle et de l'interface homme-machine. Ils tentent toujours de simplifier la vie de l'utilisateur en automatisant ses tâches à réaliser habituellement.

III.3.2.2. Agents Mobiles

Les agents mobiles sont hérités du domaine des réseaux dans le but d'effectuer des tâches réparties sur plusieurs machines interconnectées ; et ce dans un environnement distribué pour le compte d'un utilisateur ou d'une application. Un agent mobile a la capacité de se déplacer dans son environnement, il possède donc des méthodes assurant sa mobilité.

III.3.2.3. Agents collaboratifs

Les agents collaboratifs se communiquent et collaborent entre eux dans l'environnement pour accomplir leurs tâches dont le but est la résolution des problèmes communs. Ils exécutent des tâches multiples et agissent en synergie pour diffuser l'information et traiter des conflits dans des environnements multi-agents ouverts. A l'effet de coordonner leurs activités, ils négocient dans le souci de conclure des accords mutuellement acceptables. Selon [116], les agents collaboratifs permettent :

- L'interopération et Intercommunication des anciens systèmes existants,
- Le traitement des problèmes très complexes, en raison des limitations de ressource ou le risque d'avoir un système centralisé ;
- La fourniture des solutions aux problèmes de nature distribuée.

III.3.2.4. Agents d'Information/Internet

Ces agents sont conçus pour supporter les situations de surcharge d'information. Ils sont responsables de la gestion, la manipulation ou le rassemblement de l'information à partir de plusieurs sites distribués. Les agents d'information sont capables de filtrer des grandes quantités d'information et de choisir des données appropriées [116].

III.3.2.5. Agent réactif

Un agent réactif interagit selon des règles du type stimulus-réponses et il ne possède pas de représentation interne explicite dans son environnement. Les prises de décision d'un tel agent peuvent être représentées par une machine à états finis.

III.3.2.6. Agent cognitif

A l'inverse de l'agent réactif, l'agent cognitif se distingue par des capacités de raisonnement développées. Il possède une représentation explicite de ses objectifs, une représentation évoluée de l'environnement et une capacité à manipuler ces représentations pour anticiper ou réévaluer ces objectifs. D'ailleurs, il dispose d'une base de connaissance préservant l'ensemble des informations et des savoir-faire nécessaires à la réalisation de sa tâche et à la gestion des interactions avec les autres agents.

III.3.2.7. Agents hybrides

Les architectures hybrides combinent des agents réactifs et des agents cognitifs pour produire un modèle plus puissant.

III.3.2.8. Agents hétérogènes

Le système d'agents hétérogènes rassemble dans son environnement aux moins deux agents qui appartiennent à deux classes différentes ou plus. Il peut contenir aussi des agents hybrides. Les différents agents de ce type peuvent communiquer entre eux en utilisant un langage de communication d'agent (ACL).

III.3.2.9. Agents réellement intelligents

Un agent réellement intelligent hérite tous les aspects d'intelligence d'agents mentionnés précédemment. Il représente une forme idéale d'un agent intelligent, comme il est autonome avec un modèle interne de raisonnement. [116].

III.3.3. Systèmes multi agents

Un système multi agents (SMAs) est un ensemble d'agents qui représentent des entités actives du système, généralement situés dans un environnement et sont en interaction entre eux selon certaines relations. Les agents sont dotés de connaissances, d'intentions et de capacité d'évolution différente pour résoudre des problèmes complexes [113]. Il peut être considéré comme une extension de l'intelligence artificielle. Son intelligence est en fait obtenue par le passage du comportement individuel à un comportement collectif des agents afin d'engendrer de nouveaux comportements.

Les SMAs résultent de l'intersection de plusieurs domaines scientifiques : les systèmes distribués, les interfaces homme-machine, les bases de données et les bases de connaissances distribuées, les systèmes de compréhension du langage naturel, les protocoles de communication et les réseaux de télécommunication, la programmation orientée agents et le génie logiciel, la robotique cognitive et la coopération entre robots, etc [113]. La puissance d'un système multi-agents provient du comportement collectif des agents qui se traduit à partir de leurs interactions : communication, coopération, négociation et/ou coordination.

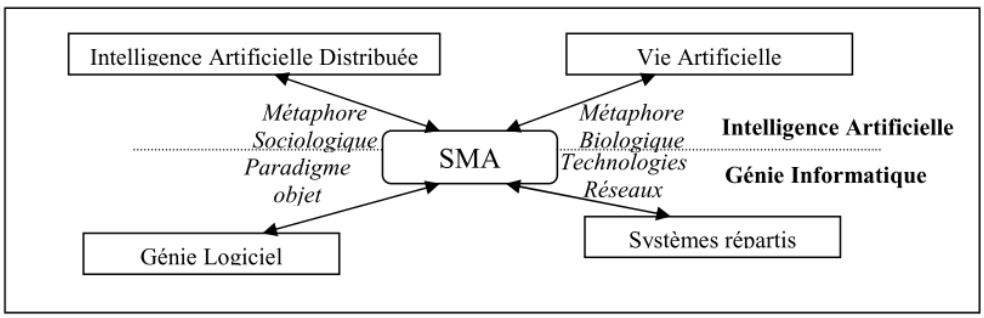


Figure III-2 : Positionnement des SMAs dans l'IA [118]

Yves Demazeau propose une approche intégrée qui décompose un SMA en quatre parties [117] :

- **Agents A** : qui concernent les modèles et les architectures utilisées pour les agents.
- **Environnement E** : représente le milieu où évoluent les agents.
- **Interactions I** : il s'agit des infrastructures, des langages et des protocoles d'interactions entre agents, depuis de simples interactions physiques à des interactions langagières par actes de langage.
- **Organisation O** : qui structurent les agents en groupes, hiérarchies, relations, etc.

Cette approche a deux principes :

1. Approche déclarative : **SMA** = Agents + Environnement + Interactions + Organisation
2. Approche fonctionnelle : La fonctionnalité d'un SMA s'exprime par la somme des fonctionnalités individuelles de ses agents ainsi que de leur fonctionnalité collective.

Fonction(SMA) = (∑ Fonction(Agents) + Fonction collective)

Selon Jennings, le terme SMA est utilisé pour tous les types de systèmes formés de plusieurs composants autonomes qui respectent les caractéristiques suivantes [115] :

1- chaque agent a des compétences de résolution de problèmes limitées ;

2- il n'y a aucun contrôle global dans le système.

3- les données sont décentralisées ;

4- l'exécution est asynchrone.

Les Systèmes multi-agents sont des systèmes typiques créés pour résoudre des problèmes complexes parce qu'ils héritent les caractéristiques traditionnelles de la résolution distribuée de problèmes comme la modularité, la vitesse, et la fiabilité [120]. Ils bénéficient aussi d'autres avantages de l'IA classique comme la facilité de la maintenance, le traitement symbolique, la réutilisation et la portabilité. Ils ont particulièrement l'avantage de faire intervenir des schémas d'interaction sophistiqués [119].

III.3.4. Coopération entre agents

La coopération entre les agents est une caractéristique principale du système multi-agents puisque la résolution distribuée d'un problème est le résultat de l'interaction coopérative entre les différents agents. Dans la littérature du SMAs, il existe plusieurs méthodes de coopération telles que : [113][119]:

1. *La communication :* elle est une caractéristique très importante dans les SMAs et représente la base des interactions des agents et la résolution coopérative des problèmes. Par la communication [120], les agents sont capables de coopérer, de négocier, de coordonner leurs actions ou de réaliser des tâches en commun.
2. *La spécialisation* : cette méthode permet à chaque agent de se spécialiser dans un seul type de tâches spécifiques.

3. *Le regroupement et la multiplication* : Ils reposent sur la constitution d'un bloc d'agents qui permet de produire un comportement collectif en vue d'effectuer un ensemble d'actions qu'un seul agent ne peut pas mener.
4. *La collaboration par partage de tâches et de ressources* : il existe plusieurs mécanismes pour le partage des tâches et des ressources, nous les citons comme suit :
    - les mécanismes d'élection : les tâches sont attribuées à des agents suite à un accord ou un vote,
    - les réseaux contractuels : les tâches sont attribuées aux agents suite à des cycles d'appels d'offres ou de propositions.
    - La planification multi-agents : la responsabilité de la distribution des tâches est attribuée aux agents planificateurs, où les agents ont des responsabilités pour des tâches particulières.
5. *Coordination des actions :* Elle repose sur la coordination des actions entre les agents pour assurer un fonctionnement cohérent et adéquat du système.
6. *La résolution de conflit par arbitrage et négociation :* elle joue un rôle fondamental dans les activités de coopération en permettant aux individus de résoudre les conflits et empêcher les situations de désaccords entre les agents qui pourraient les produire par leurs comportements coopératifs.

III.3.5. Communication entre agents

Dans les SMAs, la communication entre agents représente un pivot crucial dans la résolution coopérative des problèmes. Elle a pour objective de coopérer, négocier et coordonner les actions des agents pour résoudre les conflits des ressources. Dans la littérature du SMA, il y a deux types principaux de la communication entre les agents [136] : communication par partage de mémoire et communication par envoi de messages.

III.3.5.1. Communication par partage d'information

Ce type est appelé aussi le tableau noir (Blackboard). Il est utilisé dans l'IA classique pour spécifier une mémoire partagée [121]. Le tableau noir est une structure de données partagée entre différents agents qui l'utilisent pour écrire des messages, déposer des résultats de calculs et de solutions, obtenir des informations sur l'état d'un problème [122].

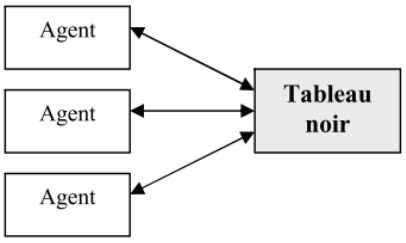


Figure III-3 : Communication par partage d'information

III.3.5.2. Communication par envoi des messages

Dans ce type de communication, les agents disposent d'une représentation de l'environnement. Ils envoient leurs messages directement et explicitement à l'agent destinataire selon divers protocoles (TCP/IP, HTTP, IIOP) et à la base d'un langage commun. Nous distinguons deux modes de transmission de messages : la mode point à point où l'émetteur du message connait l'adresse des agents destinataires et la mode par diffusion dont le message est diffusé à tous les agents du système. Selon la référence [160], plusieurs stratégies ont été mises en œuvre pour l'échange des messages entre agents telles que : l'appel de procédure à distance (Remote Procedure Call «RPC»), le bus logiciel (Object Request Brocker «ORB») et la stratégie basée sur la théorie des actes de langage.

Selon la référence [107], la stratégie basée sur la théorie des actes de langage est une base pour les communications entre agents. Ladite théorie a été conclue à partir de la philosophie du langage relevant d'un grand intérêt pour l'analyse des communications symboliques dans les SMAs. Elle fait appel aux blocs primaires du langage naturel appelés actes de langage dont la signification est claire et la structure bien formée [156].

III.3.5.3. Langages de communication entre agents (ACL)

Afin de faciliter la communication et l'interopérabilité entre les agents, de nombreux langages de communication ont été développés. Ces langages occupent une place importante dans une couche logiquement supérieure à celle des protocoles de transfert (TCP/IP, HTTP, IIOP) et adressent le niveau intentionnel et social des agents [124]. Les deux ACL les plus employés sont KQML (Knowledge Query and Manipulation Language) et FIPA-ACL (FIPA Agent Communication Language) qui est une extension du langage KQML [123].

III.3.5.4. Le langage de communication KQML

Knowledge Query and Manipulation Language (KQML) n'a vu le jour qu'en 1992 et a atteint sa maturité en 1993. Il est fondé sur un ensemble d'actes de langage standards et utiles pour que les agents cognitifs puissent coopérer. KQML repose principalement sur la séparation de la sémantique liée au protocole de communication (Indépendante du domaine d'application) de celle du contenu des messages (dépendante du domaine d'application). Malgré les avantages offerts par le langage KQML, sa sémantique n'est pas fortement formalisée, ce qui conduit à des différentes interprétations [125] et [113]. Des efforts consentis autour de ce langage ont abouti à de nouveaux langages plus fiables, en particulier le FIPA-ACL.

III.3.5.5. Le langage FIPA-ACL

Le langage FIPA-ACL développé par FIPA (Foundation for Intelligent Physical Agents) en 1996, est un langage standard de communication entre agents qui s'appuie essentiellement sur la théorie des actes de langages comme KQML. FIPA a défini des standards pour la structuration des messages, leur représentation et les mécanismes de leur transport. La spécification de ce langage consiste en un ensemble, d'une part, de types de message et, d'autre part, de protocoles d'interaction de haut niveau [126]. Le FIPA-ACL est composé de quatre primitives principales : confirm, disconfirm, inform, request. Tous les autres actes de

langage sont construits sur la base de ces quatre primitives [119]. Ce langage dispose 22 actes de langage exprimés par des performatives, et ils peuvent être classés selon leur fonctionnalité [126] (tableau III.1).

Tableau III-1 : Actes de communication FIPA-ACL, regroupés par catégories

	Transmission d'information	Demande d'information	Négociation	Action	Gestion d'erreurs
Accept-proposal			x		
Agree				x	
Cancel				x	
Cfp (call for proposal)			x		
Confirm	x				
Disconfirm	x				
Failure					x
Inform	x				
Inform-if	x				
Inform-ref	x				
Not-understood					x
Propagate				x	
Propose			x		
Proxy				x	
Query-if		x			
Query-ref		x			
Refuse				x	
Reject-proposal			x		
Request				x	
Request-when				x	
Request-whenever				x	
Subscribe		x			

Le langage FIPA-ACL qui repose sur une sémantique beaucoup plus formalisée que KQML, permet d'offrir un ensemble de protocoles de communication où chacun implique plusieurs actes de langage [124] tels que : contract net, demande d'action,… etc.

La structure d'un message FIPA-ACL comprend plusieurs éléments illustrés dans le Figure III.4.

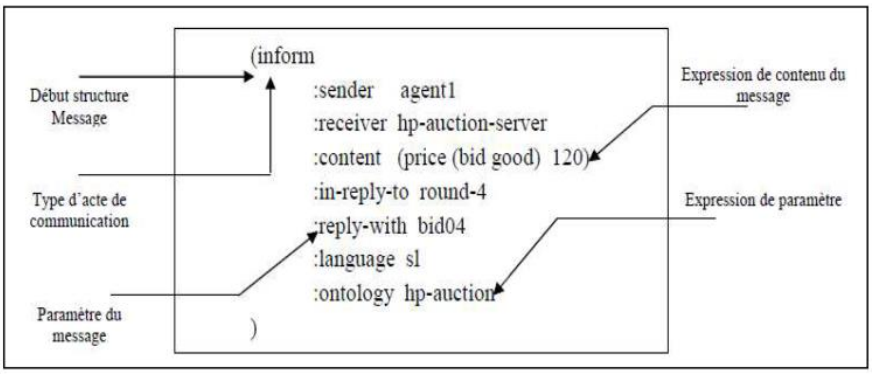


Figure III-4 : Structure d'un message FIPA-ACL

III.4. Avantages de la contribution de SMA dans le Data Mining

Le système multi-agents peut contribuer efficacement à l'amélioration du fonctionnement du processus de Data Mining. Les avantages que le système multi-agents peut fournir au système de Data Mining sont :

Décentralisation du SMA : La nature décentralisée d'un système multi-agents autorise l'exécution parallèle du processus Data Mining à partir d'une très grande quantité de données distribuées. Celles-ci sont souvent dispersées géographiquement et il est impossible de les regrouper sur un site central de traitement en raison des coûts de communication, de sécurité et des issus de préservation de la vie privée.

Efficacité des calculs : elle concerne la réduction du temps d'exécution et l'augmentation de la performance globale du système de Data mining dont son exécution se fait de façon parallèle sur des sites différents grâce au système multi-agents.

L'autonomie SMA : C'est une caractéristique très importante dans le système multi-agents, permettant aux agents de contrôler leurs actions et comportements sans l'intervention humaine ou d'un logiciel. Elle est également applicable au système Data Mining par les deux capacités suivantes :

* La capacité réactive : les agents peuvent effectuer des tâches (Ex : lancement d'un algorithme des règles d'association) et produire des résultats (Ex : l'affichage des règles d'association).

* La capacité sociale : les agents communiquent et échangent des messages entre eux tout en réalisant une tâche spécifique.

Adaptabilité et extensibilité : Le modèle agents rend le système Data Mining facilement extensible. En effet, les agents de données et les agents Data Mining peuvent être simplement ajoutés au système développé. En plus et grâce à l'utilisation des agents, le système Data Mining s'adapte aisément aux nouveaux changements lors de l'extraction des modèles (patterns) qui changent au fur et à mesure, ou bien à travers les progrès technologiques de l'apprentissage automatique qui donne lieu à de nouveaux algorithmes. En ce qui concerne l'extensibilité, de nouvelles techniques peuvent s'ajouter dynamiquement au système Data Mining, néanmoins celles anciennes peuvent être supprimées [110].

La maintenabilité : le système multi-agents offre aux utilisateurs du système de Data Mining la capacité de suivre l'avancement du processus de l'extraction de connaissances durant les différentes phases. Par exemple, un utilisateur peut avoir l'affichage du modèle de connaissance obtenue par un agent avant de faire l'intégration du résultat [111,112].

III.5. Quelques travaux du Data Mining basé Agents

L'un des défis de Data Mining actuel est de savoir comment traiter efficacement les grandes bases de données croissantes qui sont souvent distribuées sur plusieurs sites. Il s'agit d'extraire des connaissances utiles quant à la prise de bonnes décisions en temps opportun au sein des entreprises. Ce problème peut être résolu par l'exploitation de la puissance de traitement des ordinateurs d'une manière optimale. Dans ce contexte, plusieurs approches de

Data Mining ont été proposées par l'utilisation des techniques de traitement distribuées [129, 130, 131, 132, 133, 134] ou parallèles [135, 136, 137]. De la sorte, la prise en charge de ce problème peut faire appel à l'exploitation efficace de plusieurs processeurs (Voir la section II.8.7). Cependant, le contrôle centralisé sur lequel ces approches reposent est une faiblesse.

Les techniques Data Mining distribuées et parallèles nécessitent généralement un processus «maître» qui dirige la tâche d'exploration de données. Par conséquent, le contrôle est toujours centralisé au niveau du processus maître qui diminue la robustesse de ces systèmes.

La complexité du processus de Data Mining et de ses divers algorithmes a motivé les chercheurs à s'intéresser aux systèmes multi-agents afin de développer des plateformes Data Mining plus avancées, intelligents et robustes [103]. Les systèmes multi-agents (SMA) offrent une alternative pour le traitement de grandes quantités de données en exploitant la puissance du nombre de processeurs, avec un avantage supplémentaire de contrôle décentralisé. Ces systèmes sont devenus comme un nouveau paradigme qui fournit une approche très appropriée pour la conception, le développement et la mise en œuvre de systèmes de nature hétérogène, complexe, distribué et/ou autonome [127].

Tant de travaux basés sur des approches de Data Mining à base d'agents avec plusieurs techniques DM ont été réalisés. Dans la section suivante, nous allons présenter en premier lieu les travaux de Data Mining basé agents pour la tâche de clustering tels que les travaux [110], [128], [139], [140], [142], [143], [144], [145], [173] et [147]. Nous nous penchons, par la suite, sur la présentation des travaux de Data Mining basé agents en ce qui concerne l'extraction des règles d'association tels que les travaux [168], [169], [170], [167], [166], [165],[155] et [171].

III.5.1. Approches basés agents pour la tâche du Clustering

Dans ce contexte, il existe dans la littérature plusieurs approches, et nous allons les présenter dans les sections qui viennent.

III.5.1.1. Le système PADMA

Les plus anciens systèmes de clustering à base d'agents sont PADMA [139] et PAPYRUS [140] développés à la fin de des années 1990. Ces systèmes visaient à réaliser l'intégration des connaissances découvertes à partir de différents sites avec un minimum de communication réseau et un maximum de calcul local. Selon les auteurs de [138] et [139] le système PADMA (PArallel Data Mining Agents) est conçu sur une architecture centralisée qui utilise des agents logiciels pour l'accès et l'analyse des données locales via une interface Web pour la visualisation interactive des données. Ce système prend en charge la distribution naturelle des données et des calculs entre plusieurs sites de données pour répondre au problème d'échelle (scaling problem) de Data Mining. Les développeurs de système PADMA suggère que la nature très distribuée de données et des calculs dans les environnements informatiques jouera un rôle important dans la conception de la prochaine génération des systèmes de Data Mining basés agents.

Les trois principaux composants du PADMA sont : (1) interface utilisateur pour lancer les requêtes DM, agent facilitateur pour la coordination des agents, et agents Data Mining pour exécuter une tâche particulière de Data mining. Ces derniers ont accès direct aux données sur le disque pour exécuter une tâche précise de Data Mining afin d'extraire des connaissances. En plus, parmi les facteurs permettant l'augmentation de la performance de traitement, c'est bien l'utilisation des techniques d'optimisation des entrées et sorties locales par les agents DM. La réduction du temps d'exécution parallèle et autonome dans la plateforme PADMA en découle automatiquement.

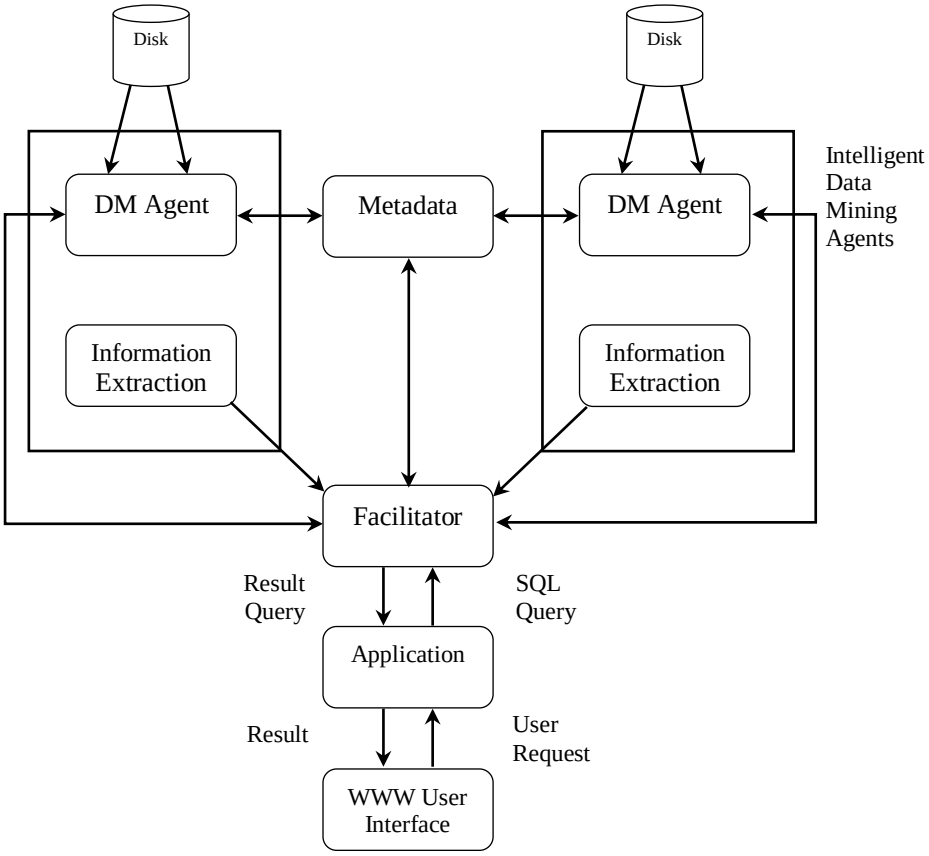


Figure III-5 : Architecture PADMA [139]

Le rôle de l'agent facilitateur est le partage des connaissances extraites entre les agents. Il affiche le résultat du processus Data Mining à l'interface de l'utilisateur et il transmet également les interactions de l'utilisateur vers les agents Data Mining.

L'indépendance entre les sites permet d'accélérer le processus Data Mining. De ce fait, lors de traitement d'une requête utilisateur, l'agent facilitateur ordonne les agents Data Mining pour exécuter un algorithme de clustering sur leurs sources de données locales. Après, l'agent facilitateur intègre le résultat de l'extraction global tout en minimisant la communication inter-agents et avec un maximum de calcul local.

L'architecture de PADMA a été expérimentée sur le clustring de textes d'un corpus afin d'identifier des relations entre textes basés sur les n-grams. Cette expérimentation suppose que les textes analysés ne contiennent pas d'erreurs typographiques et orthographiques.

Le résultat de cette expérience a montré que PADMA a donné des résultats de clustring acceptables avec un bon temps d'exécution.

III.5.1.2. Le système PAPYRUS

PAPYRUS [140] est un système de type Standard MADM (Multi Agents Data Mining) développé sous java, destiné au clustering distribué. Il manipule des méta-clusters et des super-clusters distribués sur des sites de données hétérogènes. Les méta-clusters sont des nœuds connectés par des réseaux classiques tandis que les super-clusters sont des nœuds connectés par des réseaux de haute performance.

Les fondateurs de ce système suggèrent que le clustering à base d'agents fournit une alternative compétitive aux ordinateurs spécialisés de haute performance pour traiter les grands ensembles de données distribuées [140]. Papyrus constitué de couches d'outils logiciels et de services réseau dédiés aux systèmes distribués pour l'exploration de données et le calcul intensif. Les concepteurs de Papyrus admettent qu'il existe N sites différents connectés à un réseau. Le K_{eme} nœuds est considéré comme la racine du réseau où le résultat global sera calculé. Chaque nœud peut utiliser l'une des trois stratégies suivantes :

- *MR : Move Result*, la possibilité de déplacer l'ensemble des résultats d'un processus DM local vers un site central dans le réseau.
- *MM : Move Model*, la capacité à déplacer des modèles prédictifs d'un site vers un autre dans le réseau.
- *MD : Move Data,* la possibilité de déplacer des volumes de données importants pour les traiter d'un site à un autre.

PAPYRUS a adopté un modèle Peer-to-Peer par l'utilisation des agents mobiles qui sont capables de transporter les données (MD), résultats intermédiaires (MR) et modèles de données (MM) entre les clusters pour un traitement local, ce qui va diminuer le coût de communication dans le réseau.

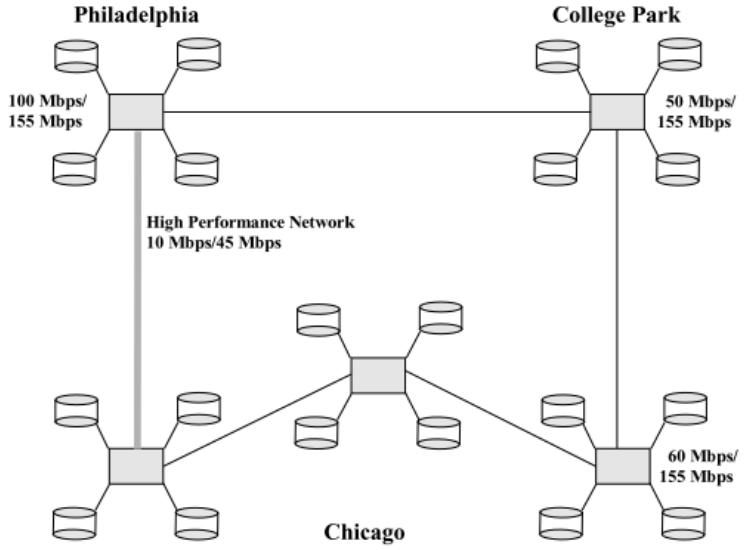


Figure III-6 : Clusters de stations de travail de Papyrus [140]

III.5.1.3. Approche basée agents pour la tâche du Clustering

Les auteurs des travaux [110], [128] et [173] ont dévoilé une approche basée agents concernant le clustering afin de fournir une bonne solution au problème générique de data mining, ce qui permet de produire un meilleur ensemble de clusters.

Dans cette approche, l'opération de clustering est effectuée par un certain nombre d'agents en utilisant des différents algorithmes de clustering qui sont appliqués sur le même ensemble de données. Une meilleure configuration de cluster est choisie par l'agent de validation en utilisant une panoplie de mesures de clustering. Par conséquent, les résultats de clustering ont été produits avec un meilleur ensemble de clusters. Ce travail a été expérimenté sur des données de type benchmark UCI pour démontrer que l'architecture proposée pourrait être utilisée avec succès dont le but est de trouver une meilleure configuration de cluster.

L'architecture proposée « Figure III.7 » comporte 5 types d'agents cognitifs et chaque agent peut réaliser une tâche spécifique de data mining, nous les citons comme suit :

1. ***L'agent d'utilisateur* :** représente une interface graphique entre l'utilisateur et les agents de tâches.
2. ***Les agents de tâches*** : permettent l'exécution des tâches de data mining. Il existe trois types d'agents distincts qui sont identifiés pour les trois tâches de data Mining suivantes : le clustering, la classification et l'extraction des règles d'association. Ces agents assurent la gestion, la planification d'une requête de Data Mining et l'identification des agents appropriés pour accomplir une tâche de Data Mining à l'aide des services de "pages jaunes".
3. ***Les agents de données*** : disposent des méta-données sur les données utilisées dans l'opération de data mining, ce qui permet à l'agent DM d'accéder à ces données.

4. ***Les agents Data Mining*** : assurent l'exécution de la tâche de Data Mining qui est déjà choisi par l'agent de tâche afin de générer des résultats DM. Les résultats sont ensuite envoyés à l'agent de validation.
5. ***Les agents de validation*** : permettent l'évaluation et la validation des résultats de data mining. Plusieurs agents de validation sont activés et associés aux différents agents de tâches déjà lancés pour l'une des tâches DM suivantes : le clustering, la classification et l'extraction de règles d'association.

Cette architecture a été implémentée en utilisant une plateforme « JADE » quant au développement des systèmes multi-agents. Elle procure un ensemble d'agents supplémentaires tels que : (1) l'agent AMS (Agent Management System) spécialisé dans la gestion et le contrôle du cycle de vie des agents de la plate-forme, (2) l'agent DF (Directory Facilitator) pour fournir un service de "pages jaune" facilitant aux agents l'enregistrement de leurs capacités et aux agents de tâches l'identification des agents appropriés.

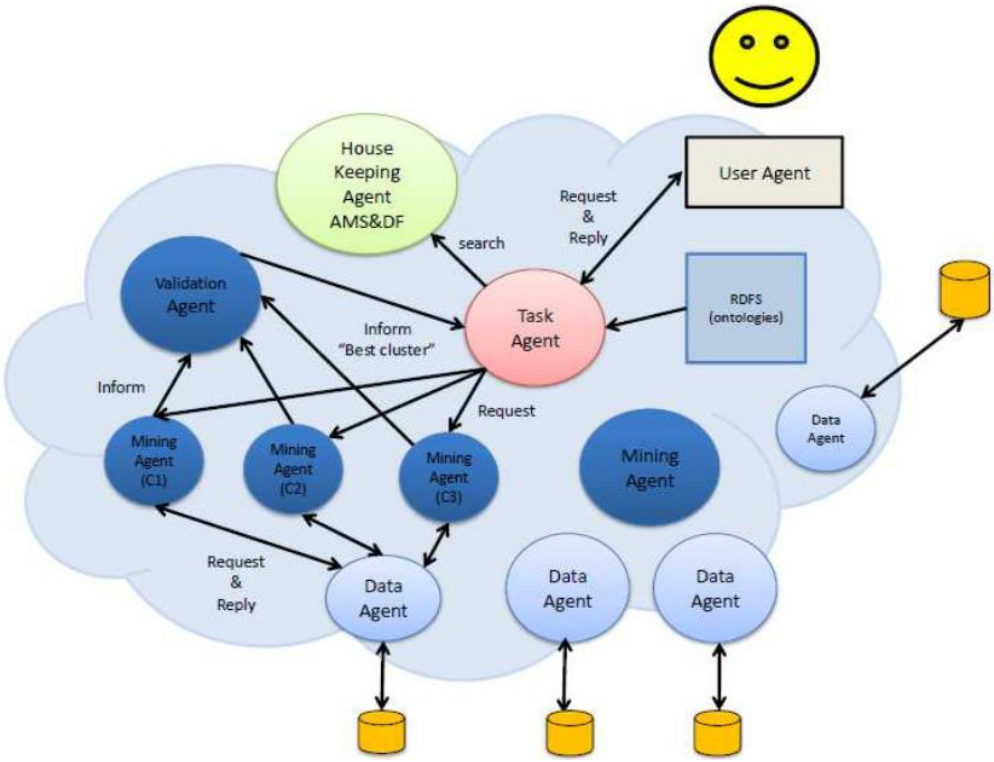


Figure III-7 : L'architecture du système proposé pour le clustering. [110]

La figure ci-dessus décrit un scénario d'exécution d'une tâche Data Mining basé agents pour réaliser le clustering des données. Le flux de contrôle commence par l'agent utilisateur qui crée un agent de tâche de clustering. L'agent de tâche créé, interagit avec l'agent DF pour découvrir les agents disponibles et accomplir la tâche de clustering des données, puis faire la sélection à partir de ces agents. Dans l'architecture présentée (dans Figure III.7), cinq agents sont activés sur la plateforme « Jade » : Trois agents de data mining, un agent de validation et un agent de données. L'agent de tâche commence à communiquer avec les trois agents Data Mining qui interagissent avec un seul agent de données (dans d'autres cas si nécessaire, il

peut interagir avec plusieurs agents de données). Par la suite, les agents Data Mining exécutent les algorithmes nécessaires pour la tâche de clustering et transmettent les résultats à l'agent de validation, qui à son tour traite les résultats et transmet le résultat final à l'agent utilisateur via l'agent de tâche.

III.5.1.4. Approche IDS fondée sur Data Mining basé agents

Les auteurs de [142] ont proposé un système de détection d'intrusion (IDS) qui est conçu sur une architecture de Data Mining basé agents. Ils suggèrent que l'utilisation de la technologie de Data Mining et des agents mobiles semblent très appropriée pour résoudre le problème des intrusions dans un environnement distribué. Ce système baptisé MAD-IDS (Mobile Agent using Data Mining based Intrusion Detection System), est conçu pour aider à surveiller les événements qui se produisent dans un environnement distribué d'ordinateurs ou d'un réseau. Plusieurs travaux de recherches ont été menés sur la détection de l'intrusion dans un environnement distribué pour pallier les inconvénients de l'approche centralisée. Cependant, l'IDS distribué porte aussi sur des inconvénients par exemple : des taux élevés de faux positifs, la faiblesse de l'efficacité, etc.

Le MAD-IDS repose sur un ensemble d'agents intelligents qui recueillent et analysent les connexions réseau par des techniques de Data Mining pour détecter les intrusions et effectuer des expertises. L'utilisation de cette approche a montré l'efficacité des IDS distribués par rapport aux systèmes de détection décentralisés classiques.

Ce système est décomposé essentiellement en deux parties : IDS pour la détection d'anomalies et IDS pour l'utilisation abusive. La Figure III.8 montre une architecture distribuée de MAD-IDS, et elle comprend différents agents qui sont : Sniffer Agent, Filter Agent, Misuse Detection Agent, Anomaly Detection Agent, Rule Mining Agent et Reporter Agent.

1- Sniffer Agent : collecte les paquets réseau et les stocke dans un « Sniffing file ». Les avantages marquant ce type d'agents sont : i) le clonage et la distribution dans tout le réseau ; et ii) la duplication afin d'alléger la charge du réseau.
2- Filter Agent : L'agent de filtrage agrège et fusionne les événements stockés dans le Sniffing file. Il effectue ses tâches comme une phase de prétraitement data mining.
3- Misuse Detection Agent (l'agent de détection d'abus) : détecte les attaques connues dans les connexions réseau. Par conséquent, s'il existe une similarité entre les paquets filtrés et les signatures d'attaque, l'agent envoie une alerte à l'agent Reporter, puis supprime ces paquets anormaux. Bien que les attaques connues soient détectées, il reste néanmoins le problème de la détection des nouvelles attaques. Une réponse au problème pourrait s'appuyer sur des techniques d'exploration de données.
4- Anomaly Detection Agent : L'agent de détection d'anomalie se focalise sur la combinaison d'IDS distribué avec des techniques de clustering. L'algorithme de détection d'anomalies basé clustering passe par les étapes suivantes :
    - Étape 1 (Initialisation) : partitionner les données d'apprentissage en k clusters;
    - Étape 2 (Affectation): Attribuer chaque instance à son centre le plus proche;
    - Étape 3 (Mise à jour) : Remplacer chaque centre par la moyenne de ses membres ;

- Étape 4 (Itération): Répéter les étapes 2 et 3 jusqu'à ce qu'il n'y ait plus de mise à jour;
- Etape 5 (Recherche d'anomalies): Pour chaque instance de test Z :

1) Calculer la distance euclidienne entre Z et un cluster initial Ci;
2) Trouver le cluster Ci le plus proche de Z;
3) Classifier Z comme une anomalie ou une instance normale en utilisant un seuil prédéfini.

5- Rule Mining Agent : L'agent d'extraction de règles construit un résumé des connexions anormales détectées par Anomaly Detection Agent. Pour exploiter les règles d'association, il applique la base générique informative (IGB) [141]. En plus, l'application de cette base IGB lors d'un processus de détection d'intrusion permet d'augmenter la couverture globale des attaques détectables et le transfert maximal de connaissances utiles. A cet effet, la base de données de signatures de l'agent Misuse Detection peut être mise à jour régulièrement en ajoutant l'ensemble de règles extraites.

6- Reporter Agent : l'agent Misuse Detection et l'agent Anomaly Detection envoient leurs résultats sous forme d'alertes à l'agent Reporter qui les transmet à l'administrateur du système.

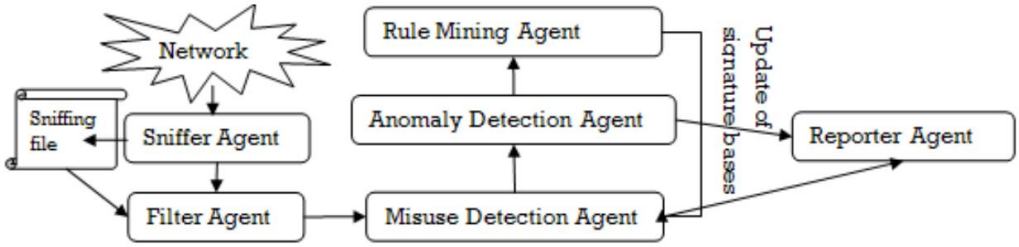


Figure III-8 : Architecture de système MAD-IDS de [142]

III.5.1.5. Autres approches

Dans cette section nous décrivons d'autres travaux pour le clustering à base d'agents plus moins importants. Les auteurs de [143] ont proposé un schéma KDEC pour minimiser les communications entre les sites dans le but de préserver la confidentialité des sites de données. Le choix de l'utilisation des agents dans le Data Mining distribué est justifié par l'autonomie, l'interactivité et la possibilité de la sélection dynamique de source de données dans un environnement dynamique. Le travail de [144] a proposé une architecture SMA pour la classification distribuée des documents textuels qui affecte de nouveaux documents entrants aux clusters. Cette proposition a pour but d'améliorer l'exactitude et la pertinence du processus de recherche d'information. Le travail de [145] décrit une application d'E-Commerce pour le clustering qui est fondé sur le Data Mining distribué à base d'agents. Ces auteurs utilisent l'algorithme de règles d'association « Apriori » pour analyser les intérêts des utilisateurs du site WWW eCommerce, en ce qui concerne les caractéristiques et le temps passé sur le site. Ensuite, un mécanisme de clustering a été utilisé pour regrouper les utilisateurs en fonction de cette analyse. Les auteurs de [147] ont créé un système multi-agents pour le clustering distribué et non distribué (JABAT basé sur l'algorithme K-means). JABAT utilise également des ontologies pour définir les vocabulaires et la sémantique du contenu lors de l'échange de messages entre agents.

III.5.2. Approches basées agents pour l'extraction des règles d'association

Nous avons confirmé à travers le chapitre précédent que l'extraction des règles d'association a reçu une attention considérable dans la communauté de recherche. En plus, elle a montré des résultats très intéressants lors de l'application dans de nombreuses problématiques [148]. L'extraction des règles d'association permet de découvrir les relations conditionnelles entre des ensembles d'attributs appelés Itemsets. Ceux-ci sont enfouis dans des grandes bases de données à l'effet d'extraire des connaissances sous forme d'un ensemble de règles associatives. Tenant compte du chapitre précédent, l'extraction de règles d'association à partir de sources de données distribuées est appelée DARM (Distributed Association Rule Mining) [149]. Cela signifie l'extraction des règles d'association dans un environnement distribué. La plupart des systèmes DARM confrontent d'énormes défis tels que le temps de réponse, les coûts élevés de communication et l'incapacité de s'adapter aux bases de données en constante évolution. (Voir Section II.8.5.)

Avec la croissance exponentielle des données stockées dans les bases de données des entreprises, l'extraction des règles d'association distribuée est très difficile dans un seul processus qui nécessite un énorme volume de travail et un temps d'exécution considérable afin de finaliser leurs traitements. En outre, les algorithmes utilisés dans ce cadre sont gourmands en ressources (en termes de consommation mémoire et processeur). C'est pour cela que les chercheurs proposent des méthodes plus efficaces pour mener à bien ces tâches minières. Ils pensent que le partitionnement et la distribution des grandes bases de données sur des petites partitions améliorera la performance globale de l'extraction des règles d'association. (Voir Section II.8.4.). Dans ce sens, plusieurs approches de DARM été également proposées pour le partitionnement de données telles que les travaux de Albashiri et al. [150], Coenen et Leng [151], Pudi [152] et Coenen, et Leng [129] (plus de détails dans la section II. 8.7). Eu égard à ces approches, ils ont essayé de réduire le nombre de faux Itemsets candidats et ont considéré la priorité du partitionnement de données comme un facteur principal pour améliorer les traitements du processus d'extraction de règles d'association distribuée. Dans la littérature, il existe deux types de partitionnement de données à savoir, le partitionnement horizontal et le partitionnement vertical de données. Certaines approches comme Albashiri [153], Coenen et Leng [151] ont travaillé sur la distribution et le partitionnement horizontal de données. La distribution et le partitionnement vertical de données ont également été explorés par les travaux de Coenen et Leng [129], [153] et [154]. Trois principales stratégies ont été identifiées par de nombreux chercheurs [155] et [136] pour les systèmes DARM qui utilisaient l'algorithme Apriori ou ses variantes. Ces stratégies sont la Data Distribution, la Candidate Distribution et la Count Distribution. Les algorithmes DARM basés sur la distribution de données (Data Distribution) partitionnent à la fois les itemsets candidats et la base de données, tout en utilisant le Système à mémoire partagée (SMP) [160]. Les algorithmes de règles d'association parallèles / distribués basés sur la distribution candidate (Candidate Distribution) partitionnent les ensembles itemsets candidats mais répliquent sélectivement les transactions de base de données pour une extraction indépendante de chacun des processus. Les algorithmes DARM basés sur la Count Distribution utilisent une méthode parallèle aux données avec un partitionnement horizontal de la base de données pour l'analyse locale et la détection de tous les ensembles des Itemsets candidats [151, 161, 162].

Actuellement, le partitionnement de données est devenu possible dans de nombreux contextes grâce à l'utilisation des agents, qui sont des logiciels situés dans un environnement particulier et capables d'effectuer des actions autonomes dans cet environnement afin d'atteindre leurs objectifs [58]. Le paradigme multi-agents a été utilisé comme une technologie puissante pour le développement de systèmes d'extraction de règles d'association distribuées (DARM) [106], [107] et [108]. Dans ce contexte, nous trouvons plusieurs approches de DARM basées agents telles que :

1. Mobile agent based distributed knowledge discovery system (MADKDS) [68]-(2003) pour un système de reconnaissance des connaissances distribuées basées agents mobiles (MADKDS),
2. Efficient Distributed Data Mining using Intelligent Agents [65]-(2004) il s'agit d'une approche d'extraction efficace de données distribuée en utilisant des agents intelligents
3. Mobile Agent based Distributed Data Mining [69]-2007 il concerne une approche d'extraction de données distribuée basée sur les agents mobiles
4. An Agent based Framework for Association Rules Mining of Distributed Data (AFARMDD) [70]-(2009) qui repose sur une plateforme de règles d'association basée agents pour les données distribuées,
5. Multi-Agent Distributed Association Rule Miner (MADARM) [67]-(2011) pour une approche d'extraction de règles d'association basée multi-agents,
6. Agent Based Data Distribution for Parallel Association Rule Mining (ABDDPARM) [55]-(2014) il s'agit d'une approche de distribution de données basée agents pour l'extraction de règles d'association parallèle,
7. Partition Enhanced Mining Algorithm (PEMA) [66]-(2015) : c'est une approche d'extraction de règles d'association distribuée améliorée par le partitionnement de données.
8. Agent enriched Mining of Globally Strong Association Rules (AeMGSAR) [71]-(2015) : une approche d'extraction de règles d'association fortes globalement améliorée par les agents.

La plupart de ces approches sont des projets de recherche académique. En ce qui suit, nous présentons brièvement quelques travaux modernes pour l'extraction de règles d'association distribuées (DARM) basés agents :

Le premier travail de [167] qui a proposé des modèles de coûts théoriques pour un DARM basé agent sur des données distribuées (MADARM). Ces modèles de coûts servent de modèle de base pour estimer et prévoir le temps de réponse d'une tâche de DARM. Plusieurs agents sont impliqués dans ce système tels que: ARMCA (Association Rule Mining Coordinating Agent) pour créer et coordonner les autres agents, MAARM (Mobile Agent-Based Association Rule Miner) quant à l'exécution de la tâche ARM sur chaque source de données distribuée, Mobile Agent Based Result Reporter (MARR) est créé par l'agent MAARM concernant la migration du résultat vers l'agent ARMCA, et, enfin, l'agent de coordination et intégration des résultats (RICA) pour l'intégration des connaissances ou des résultats. L'intégration des connaissances est optimisée sur les sources de données à l'aide de l'intégration des connaissances distribuées basée sur les agents (ADKI). Les modèles de coûts

théoriques constituent le cœur de cette étude qui a utilisé les algorithmes Apriori et FP-growth pour extraire les itemsets fréquents locaux. L'itinéraire parallèle pour MAARM, MARR et l'itinéraire en série pour l'agent RICA sont maintenus. Seuls les points de vue conceptuels sont présentés dans le document tandis que les chercheurs ont conclu que le travail a encore besoin d'amélioration et de validation expérimentale.

Les auteurs du travail [155]-(2014), ont proposé une approche de distribution de données basée agents pour l'extraction des règles d'association parallèle. Ce système comprend une collection d'agents coopérants pour traiter les tâches de Data Mining distribué. L'exploration du système est effectuée en considérant un scénario d'extraction de règles d'association (ARM) parallèle / distribué spécifique, en se basant sur le partitionnement de données (vertical ou horizontal). Pour faciliter le partitionnement de données, une structure de données d'arbre (T-Tree) est utilisée par un algorithme de règles d'association Apriori-T. Le but de ce scénario est de démontrer que cette approche est capable d'exploiter les avantages de l'informatique parallèle tels que le traitement de requête particulièrement parallèle et l'accès parallèle aux données. Les deux techniques de partitionnement des données (verticales ou horizontales) sont évaluées et comparées avec des données synthétiques. L'expérimentation indique que les méthodes de comparaison des mesures de partitionnement des données décrites sont extrêmement efficaces pour limiter les exigences de la mémoire lors de l'exécution des algorithmes de règles d'association, tandis que leur temps d'exécution ne varie que lentement et linéairement avec des dimensions de données croissantes.

Dans le travail de [166]-2015, une approche PEMA (Partition Enhanced Mining Algorithm) est proposée pour améliorer l'extraction de règles d'association distribuée en utilisant le partitionnement de données. Elle est basée essentiellement sur la combinaison du partitionnement horizontal et vertical de données. Dans cette approche, l'agent de coordination de règles d'association reçoit une requête pour choisir les sites de données appropriés, la stratégie de partitionnement et les agents d'exploration de données à utiliser. Le processus d'extraction est divisé en deux étapes. Dans la première étape, les agents de données segmentent horizontalement la base de données en petites partition en fonction du nombre de sites disponibles et la taille de mémoire disponible. D'un autre côté, si la base de données est de taille importante (Nombre de colonnes important), elle est partitionnée verticalement. Après cela, les agents mobiles de règles d'association, découvrent les Itemsets fréquents locaux. Dans la deuxième étape, les Itemsets fréquents locaux sont, de manière incrémentale, intégrés d'un site de données à un autre pour obtenir les Itemsets fréquents globaux. Les expériences de cette approche sont appliquées sur des ensembles de données réelles [172]. Elles ont montré que le temps de réponse moyen de PEMA a connu une amélioration par rapport aux algorithmes existants.

Dans le dernier travail de [171]-(2015), une plateforme basée agents mobiles est conçue pour l'extraction de règles d'association de données distribuées (AeMGSAR). Elle est appliquée en particulier dans le domaine de la bio-informatique pour étudier les associations entre les acides aminés dans les protéines. Ce système comprend deux modèles selon la stratégie de la mobilité. Le premier modèle est basé sur la mobilité en série alors que le deuxième est basé sur la mobilité parallèle des agents. Dans le modèle de calcul en série,

chaque agent mobile visite chaque nœud en séquence et revient finalement du dernier nœud visité à la station de lancement centrale. AeMGSAR a été amélioré en tant que modèle de calcul parallèle basé sur le modèle de mobilité parallèle. Dans ce modèle, les clones d'agent mobile d'extraction des règles d'association visitent chaque nœud en parallèle et reviennent immédiatement à la station de lancement centrale. Les performances du système comparées avec l'approche centralisée de l'entrepôt de données, ont montré que AeMGSAR fournit des caractéristiques améliorées et présente aussi des performances supérieures à celles des plateformes existantes telles que MADKDS [168]-(2003), AFARMDD [170]-(2009) et MADARM [167]-(2011).

III.6. Approches Data Mining pour les ERP

Plusieurs approches ont été proposées dans le cadre d'intégration de la technologie de Data Mining dans une plateforme ERP, mais peu d'entre elles qui utilisent des systèmes multi-agents. Dans ce qui suit, nous présentons brièvement les approches trouvées dans la littérature pour intégrer le Data Mining avec le système ERP, et ce dans le but de la prise de décision.

Les travaux [174], [177] et [178] montrent une meilleure intégration de l'ERP avec le Customer Relationship Management (CRM) par l'utilisation de techniques de Data Mining pour l'aide à la décision dans une entreprise. Ce modèle se compose de trois vues : une vue extérieure CRM, une vue intérieure ERP et une vue d'extraction de connaissances. Les techniques de Data Mining appliquées dans ces travaux sont le clustering et les règles d'association. Ainsi, Ces travaux ont été validés par une étude de cas en utilisant les données ERP de Madar et la mise en œuvre de l'algorithme « Apriori » sur celle-ci pour la génération de nouvelles règles afin de répondre aux futures perspectives de MADAR.

Dans l'approche [176], les auteurs abordent le domaine de Data Mining à partir d'une grande base de données ERP. La méthode de fouille choisie dans ce travail est « les règles d'association » en utilisant l'algorithme « FP Growth » qui extrait de nouvelles règles pour améliorer les décisions dans une telle organisation. Cette approche a été validée par une véritable étude de cas sur les données ERP d'organisation ABC. L'architecture proposée est appliquée sur les données relatives aux requêtes et réponses des clients. De ce fait, l'algorithme FP Growth est implémenté sur la base de données ERP pour trouver et générer de nouvelles règles et anticiper les prochaines requêtes des clients.

Les auteurs dans [175] présentent une architecture complète pour l'application de Data Mining sur la plateforme ERP afin de trouver des réponses aux requêtes clients. L'architecture proposée est composée de trois couches : «CRM», «DÉCOUVERTE DE CONNAISSANCES» et «ERP». Il s'agit essentiellement d'étendre la structure classique en couches par l'ajout d'une troisième couche de découverte de connaissances. La technique DM appliquée dans ce travail est la génération des règles d'association à partir de la base de données ERP, en utilisant l'algorithme CBA distribué qui montre un meilleur résultat par rapport à l'algorithme classique « Apriori ».

Dans nos travaux [201], nous avons proposé une approche coopérative multi-agents pour la découverte des connaissances sur le système ERP pour extraire les connaissances

utiles, cachées dans la base de données ERP. L'architecture proposée est composée de trois couches : «couche d'interface utilisateur», «couche ERP» et «couche de découverte des connaissances». Les objectifs de ce travail sont reposés sur deux dimensions : la première, est d'intégrer une nouvelle couche de découverte des connaissances sur la plateforme ERP. La deuxième est d'enrichir cette couche-là par l'utilisation du système multi-agents pour une extraction efficace des connaissances à travers la base de données ERP. Celle-ci permet d'aider les leaders de l'entreprise à prendre de bonnes décisions en temps opportun.

III.7. Conclusion

Dans ce chapitre nous avons décrit l'état de l'art associé aux systèmes de Data Mining basés agents. Nous nous sommes focalisés particulièrement sur les approches basées agents pour les tâches du clustering et de l'extraction des règles d'association en vue de positionner nos approches proposées qui seront présentées dans les chapitres suivants.

Nous avons présenté en premier lieu les systèmes multi-agents, qui sont devenus un outil fort pour pallier aux problèmes posés dans les systèmes de Data mining. Les caractéristiques intrinsèques des agents rendent leur utilisation dans des systèmes hétérogènes, complexes, distribués, ouverts, et dynamiques très appropriée. En effet, après avoir défini le concept d'agent en tant qu'entité autonome et le paradigme de systèmes multi-agents, et leurs positionnements comparativement aux autres domaines scientifiques tels que : les systèmes distribués, les interfaces homme-machine, les bases de données et les bases de connaissances distribuées, etc. Nous avons conclu que la puissance des SMAs provient du comportement collectif des agents et se traduit à partir de leurs interactions : communication, coopération, négociation, ou coordination.

Dans le chapitre suivant nous allons présenter une approche [221] multicouches basée agents à partir d'une base d'une donnée centralisée et partagée du système ERP. Elle consiste à distribuer la complexité de la tâche du clustering sur plusieurs entités coopérantes et autonomes. La solution proposée est dédiée à la tâche de clustering basée sur l'algorithme k-means pour regrouper les enregistrements ou les observations en classes d'objets similaires, afin d'aider les décideurs des entreprises dans la prise de bonnes décisions en temps réel.

Chapitre IV : Une approche multi-agents coopératifs pour le clustering de données via un ERP

(**CMAAC-ERP:** A **C**ooperative **M**ulti-**A**gent **A**pproach based **C**lustering in **E**nterprise **R**esource **P**lanning)

IV.1. Introduction

Les progiciels de type ERP visent à centraliser les données sur une grande base de données centrale. Cette architecture offre une image unifiée, intégrée, cohérente et homogène des informations entre les différents acteurs métiers de l'entreprise. Au fil du temps et à cause des usages quotidiens, la base de données ERP accumulera une énorme quantité de données qui est souvent sous exploitée, alors qu'elle peut renfermer des connaissances décisionnelles que des managers peuvent ignorer. Ce réservoir de données ERP représente une importante mine d'informations que l'entreprise doit exploiter et explorer pour découvrir des connaissances pertinentes et utiles à des fins de prédiction et de prise de décisions stratégiques dans les entreprises. A cet effet, le progiciel ERP demande désormais d'être renforcé par un outil décisionnel robuste dans le but d'analyser et d'interpréter la grande masse de données ERP afin d'en extraire des connaissances décisionnelles utiles.

L'intégration de cet outil décisionnel avec le système ERP offre la possibilité d'exploiter intuitivement la richesse des données ERP et de répondre aux problématiques de plusieurs fonctions simultanément : la production, la logistique, les finances, les ventes… etc. En outre, il facilite la vérification de la cohérence des données pour garantir une bonne qualité de décision dans l'entreprise. Il permet ainsi à l'ensemble des collaborateurs, quel que soit leur métier, de prendre les décisions pertinentes basées sur les indicateurs de performances et de suivre avec précision les objectifs initialement définis pour tous les départements de l'entreprise.

Dans ce contexte et pour soutenir la prise de décision dans l'ERP, plusieurs approches ont été proposées pour intégrer la technologie Data Mining sur la plateforme ERP afin d'atteindre les objectifs décisionnels tracés. La technologie de Data Mining est enfouie dans des grandes bases de données dans le but de valoriser les informations et l'extraction des nouvelles connaissances à caractères prévisionnels et/ou décisionnels. Cette technologie permet d'effectuer des traitements plus ou moins sophistiqués sur la base de données ERP pour extraire des connaissances utiles pour la prise de décision stratégique au sein des entreprises. L'objectif atteint par le Data Mining est donc celui de la valorisation des données contenues dans les systèmes d'information des entreprises.

Dans cette perspective, ce chapitre vise à proposer une approche multicouche à base d'agents pour intégrer la technologie de Data Mining avec la plateforme ERP. L'approche proposée **« CMAAC-ERP »** est destinée à la tâche du clustering de données pour une extraction efficace des connaissances à partir d'une grande base de données ERP. Elle est basée sur les concepts du système multi-agents dont le but de distribuer la complexité de l'algorithme du clustering « k-means » sur plusieurs agents coopératifs. Celle-ci permet de

regrouper efficacement les enregistrements de données métiers ERP en classes d'objets similaires. Cela vise à constituer des clusters homogènes et utiles pour aider les managers à prendre de bonnes décisions en temps opportun.

L'objectif de ce chapitre est de présenter notre approche « CMAAC-ERP » pour le clustering de données ERP dont le but d'améliorer la prise de décision dans les systèmes ERP. Nous nous intéresserons en premier lieu par la présentation des fondements théoriques liées à l'approche proposée. Nous montrons par la suite l'architecture générale du système proposé pour le clustering de données ERP. Ensuite, nous décrivons l'architecture fonctionnelle des agents du clustering basée K-Means ainsi qu'un scénario des interactions entre les agents développés. Enfin, nous clôturons ce chapitre par une analyse comparative entre notre approche proposée « CMAAC-ERP » et les travaux existants de clusting de données basés agents présentés dans le chapitre précédant.

IV.2. Fondements théoriques

Pour atteindre notre but dans le présent chapitre, on doit utiliser quelques techniques, approches, et outils. Dans cette section, nous abordons les fondements théoriques relatifs à l'approche proposée, à savoir : l'ERP, le Data Mining, le Clustering de données, les fonctions de mesure de similarité, et l'algorithme du clustering « K-Means ».

IV.2.1. Progiciels de gestion intégrés (ERP)

La référence (MARKESS, 2013) [12] définit une solution de gestion intégrée, également dénommée ERP (Entreprise Resource Planning) ou PGI comme étant :

- ✓ Une solution logicielle paramétrable et modulaire permettant la gestion de plusieurs processus fonctionnels ou opérationnels d'une entreprise (des achats, des approvisionnements, de la GRH, comptable et /ou financière, de la production, des stocks, des ventes, de la relation client, etc.).
- ✓ Les processus gérés sont intégrés de manière modulaire (module fonctionnel par module fonctionnel), tout en partageant une base de données unique et centrale.
- ✓ La solution doit couvrir au moins deux domaines fonctionnels différents (par exemple, RH et comptabilité/finance, ou encore comptabilité/finance et gestion commerciale…).

A partir de cette définition (MARKESS, 2013), nous remarquons que la centralisation de données sur une grande base de données représente une caractéristique primordiale du système ERP. Celle-ci assure une intégration complète de l'ensemble des processus fonctionnels clés de l'entreprise couvrant les domaines financier, ressources humaines, logistique, production, marketing et vente…etc. [185].

IV.2.2. Data Mining et Clustering de données

Comme nous avons vu dans le chapitre II que le KDD (Knowledge Discovery in Data) représente un processus non-trivial d'extraction des connaissances implicites, précédemment inconnues et potentiellement utiles concernant les données stockées dans des grandes bases de données. D'ailleurs, le KDD est composé de plusieurs étapes allant de la préparation des données jusqu'à l'interprétation des résultats en passant par la découverte proprement dite « le

Data Mining ». [191] et [192]. L'étape de Data Mining est le cœur de ce processus qui permet l'extraction, la découverte des connaissances utiles et des modèles cachés dans des grandes bases de données ou des entrepôts de données. [47].

La technologie de Data Mining utilise plusieurs méthodes et divers algorithmes pour accomplir ses tâches d'extraction des connaissances. Dans le cadre de ce chapitre, nous travaillons sur la tâche de clustering de données dans le but de découvrir de la grande base de données ERP, des groupes des données similaires formant des clusters de données ERP homogènes, non identifiés à l'avance, ayant les mêmes caractéristiques. Les clusters produits doivent contenir des objets partageant un haut degré de similarité (maximisation de la similarité intra-cluster) et avec une minimisation de la similarité inter-cluster. La similarité entre les objets est également mesurée par une fonction de distance, selon le type de données employées.

IV.2.3. Mesure de similarité

La mesure de similarité ou ressemblance (proximité) est une partie importante de la définition de la méthode de clustering. Elle a pour but de définir et formaliser une fonction de similarité, qui permet de mesurer les liens entre les objets (points, images, classes, phonème...), adaptés aux caractéristiques des données utilisées. A cet effet, plusieurs notations comme la similarité, la dissimilarité ou la distance peuvent être utilisées pour mesurer le lien entre les différents objets d'un même ensemble :

1. **Similarité** : si sa valeur est grande, le lien entre deux objets sera plus fort.
2. **Dissimilarité** : si sa valeur est petite, le lien entre deux objets sera plus fort.
3. **Distance** : si les mesures ont des propriétés de non-négativité, réflexivité, symétrie et respectent l'inégalité triangulaire, nous utilisons souvent la « distance » comme une mesure de similarité entre les objets.

Dans la littérature, il existe un nombre considérable des fonctions des mesures de distances entre les objets. Elles sont toujours liées au type de données employées telles que les données numériques ou bien les données nominales…etc. Dans ce qui suit, nous présentons quelques fonctions les plus populaires pour mesurer la distance entre deux objets (Xi et Xj) de type numérique (continues ou discrètes) :

- La distance d(Xi,Xj) de Minkowski d'ordre α définie par :

$$\forall X_i, X_j \in X;\ d(X_i, X_j) = \left(\sum_{h=1}^{m} \left| Y_h(X_i) - Y_h(X_j) \right|^{\propto} \right)^{\frac{1}{\propto}}$$

Où m décrit le nombre de données quantitatives discrètes ou continues et $\alpha \geq 1$

- Si $\infty = 1$, la distance d(Xi,Xj) est celle de City-block ou Manhattan :

$$d(X_i, X_j) = \sum_{h=1}^{m} \left| Y_h(X_i) - Y_h(X_j) \right|$$

- $\alpha = 2$, on trouve la distance Euclidienne classique :

$$d(X_i, X_j) = \sqrt{\sum_{h=1}^{m} \left(Y_h(X_i) - Y_h(X_j)\right)^2}$$

- $\alpha = \infty$, d(Xi,Xj) est la distance de Chebyshev :

$$d(X_i, X_j) = \max_{1 \le h \le m} \left|Y_h(X_i) - Y_h(X_j)\right|$$

Dans la plupart des cas on utilise la distance euclidienne. Cependant l'utilisation de distance de Manhattan est parfois utile, notamment pour amoindrir l'effet de larges différences dues aux points aberrants, car leurs coordonnées ne sont pas élevées au carré. Les résultats de distance de Manhattan sont similaires dans la plupart des cas à ceux de résultats de distance euclidienne.

IV.2.4. Algorithme K-Means

K-Means est l'algorithme de clustering le plus populaire et le plus utilisé dans la catégorie de l'apprentissage non supervisé. Il a pour but de diviser une population donnée en K groupes homogènes appelés clusters de telle sorte à avoir dans chaque cluster les données qui ont des caractéristiques de similarité forte, et que les clusters entre eux doivent être différents les uns des autres. K-Means est un algorithme du clustering itératif qui permet de minimiser la distance entre les individus et les centres des clusters. Le fonctionnement de l'algorithme k-Means se déroule comme suit [182], [200] et [206] :

1. *Choisir k objets formant ainsi k clusters qui représentent la position moyenne des partitions M1(1),…,Mk(1) initiales.*
2. *(Ré) affecter chaque objet Oj au cluster Ci de centre Mi tel que dist(Oj,Mi) est minimal*

   $Ci(t) = \{Oj : || Oj - Mi(t) || \le || Oj - Mi^*(t) ||$ *tel que* $i^* = 1,2, \ldots, k\}$
3. *Recalculer Mi de chaque cluster (le barycentre)*
4. *Aller à l'étape 2 s'il faut faire une affectation*
5. *Refaire les étapes (2) et (4) jusqu'à ce qu'il n'y est aucun changement du calcul des centres des clusters Ci(t) ou une stabilité des objets.*

Figure IV-1 : Fonctionnement de l'algorithme K-Means

Pendant la première étape, le choix des centres des clusters initiaux est extrêmement important puisqu'il a une influence directe sur le résultat final du Clustering. Il est donc très important de choisir des clusters bien séparés [200]. L'algorithme de base « K-Means » basé sur une initialisation aléatoire. Dans la littérature, il existe des travaux pour améliorer cette étape puisqu'elle influe sur le résultat final du clustering. Dans le reste de ce chapitre nous nous intéressons à modéliser cet algorithme par le paradigme du système Multi-agents afin de distribuer la complexité de traitement du clustering de données ERP entre plusieurs agents autonomes et coopératifs.

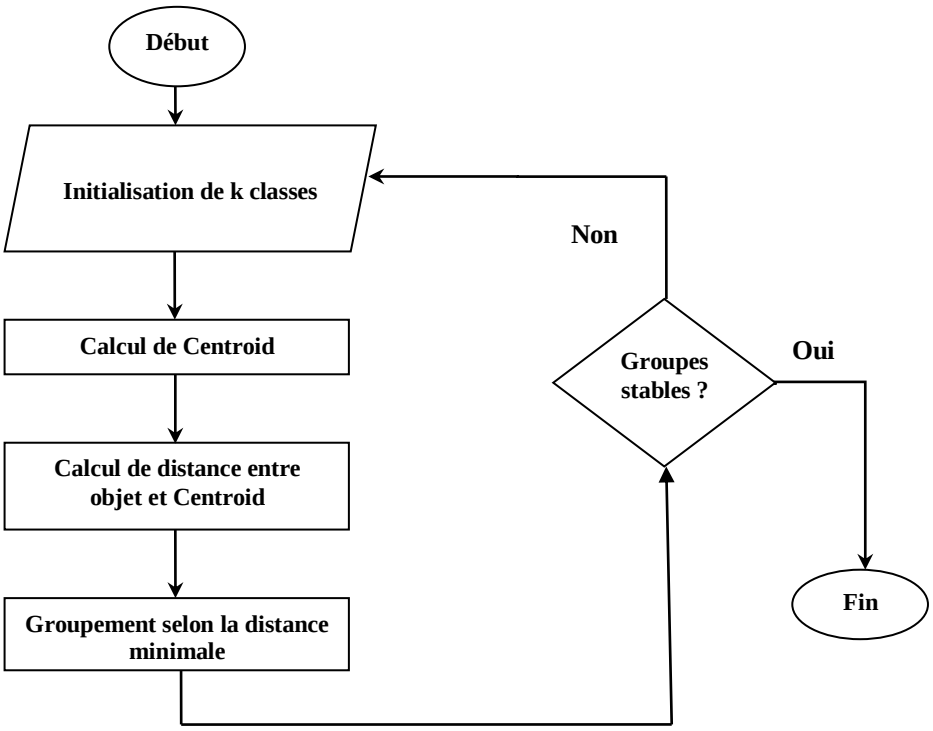


Figure IV-2 : Organigramme de fonctionnement du K-Means [200]

IV.3. Motivation et objectifs de l'approche proposée

Dans tous les travaux présentés (Voir la section III.6) [1, 2, 3, 4, 5 et 31], l'objectif est d'intégrer le module du Data Mining avec la plateforme ERP, afin d'extraire des connaissances cachées dans la base de données ERP. Ces travaux sont généralement basés sur des algorithmes spécifiques tels qu'Apriori, FP Growth, CBA et K-Means, qui sont gourmands en ressources que ce soit sur le volet de la consommation de processeur ou de mémoire. Par conséquent, ces approches ont pris un temps considérable lors des traitements de données pour l'extraction des connaissances à partir d'énorme quantité de données ERP. D'ailleurs, la base de données ERP augmente au fil du temps et avec l'utilisation quotidienne du système ERP. Selon la référence [202], le Data Mining et l'extraction des connaissances dans des très grandes bases de données de pétaoctets représentent des difficultés dans la mise en œuvre de système de Data Mining. Ils nécessitent généralement un énorme volume de travail et un temps d'exécution considérable pour extraire des connaissances pertinentes à partir de leurs bases de données. Dans ce contexte, nous proposons notre approche multicouches « CMAAC-ERP » [221] à base d'agents coopératifs pour la tâche de clustering de données via un système ERP.

IV.4. Architecture multicouches de l'approche proposée CMAAC-ERP

Dans cette section, nous décrivons l'architecture multicouches propre à notre approche proposée « CMAAC-ERP » pour la tâche clustering basée « K-means » et ce à partir d'une grande base de données ERP centrale.

D'ailleurs, l'algorithme K-Means est considéré comme l'un des meilleurs algorithmes dédiés à la tâche de segmentations de données dans le but d'extraire des clusters de données

homogènes. Néanmoins, il peut prendre un temps considérable lors de l'exécution de ses traitements du clustering de données, surtout avec la croissance de la quantité de données avec l'usage quotidienne.

Dans ce cadre, nous proposons notre approche à base d'agents « CMAAC-ERP » [221] pour le clustering de données ERP dont le but est de soutenir la prise de décision dans le système ERP. La puissance de notre approche « CMAAC-ERP » provient de l'utilisation du système multi-agents dont ses caractéristiques intrinsèques sont la collaboration la distribution, la flexibilité, l'évolutivité et l'efficacité. L'approche « CMAAC-ERP » assure la répartition des tâches effectuées par l'algorithme k-Means sur plusieurs agents coopérants et autonomes pour une extraction efficace des connaissances cachées dans la base de données ERP.

Les principaux objectifs de « CMAAC-ERP » reposent sur deux dimensions principales : la première est l'intégration de la plateforme ERP avec une nouvelle couche de découverte des connaissances basée sur la tâche du clustering. La deuxième est l'enrichissement de cette couche par l'utilisation du système multi-agent pour distribuer la complexité de l'algorithme k-Means sur plusieurs agents coopérants. Cela vise à réduire le temps global d'exécution nécessaire à la tâche de la segmentation de données ERP. Par conséquence, notre approche permet un regroupement efficace des enregistrements de données ERP en groupes de données similaires formant des clusters de données ERP homogènes dont le but est de servir les managers de l'entreprise à prendre des meilleures décisions en temps opportun.

L'approche « CMAAC-ERP » est fondée sur une architecture multicouches à base d'agents où chaque couche possède ses propres fonctionnalités ayant des responsabilités précises. Chaque agent du système possède également une compétence particulière et un sous ensemble de connaissances du domaine, qu'il utilise pour résoudre une partie du problème de clustering de données ERP. L'architecture multicouche proposée est illustrée dans la Figure IV 3.

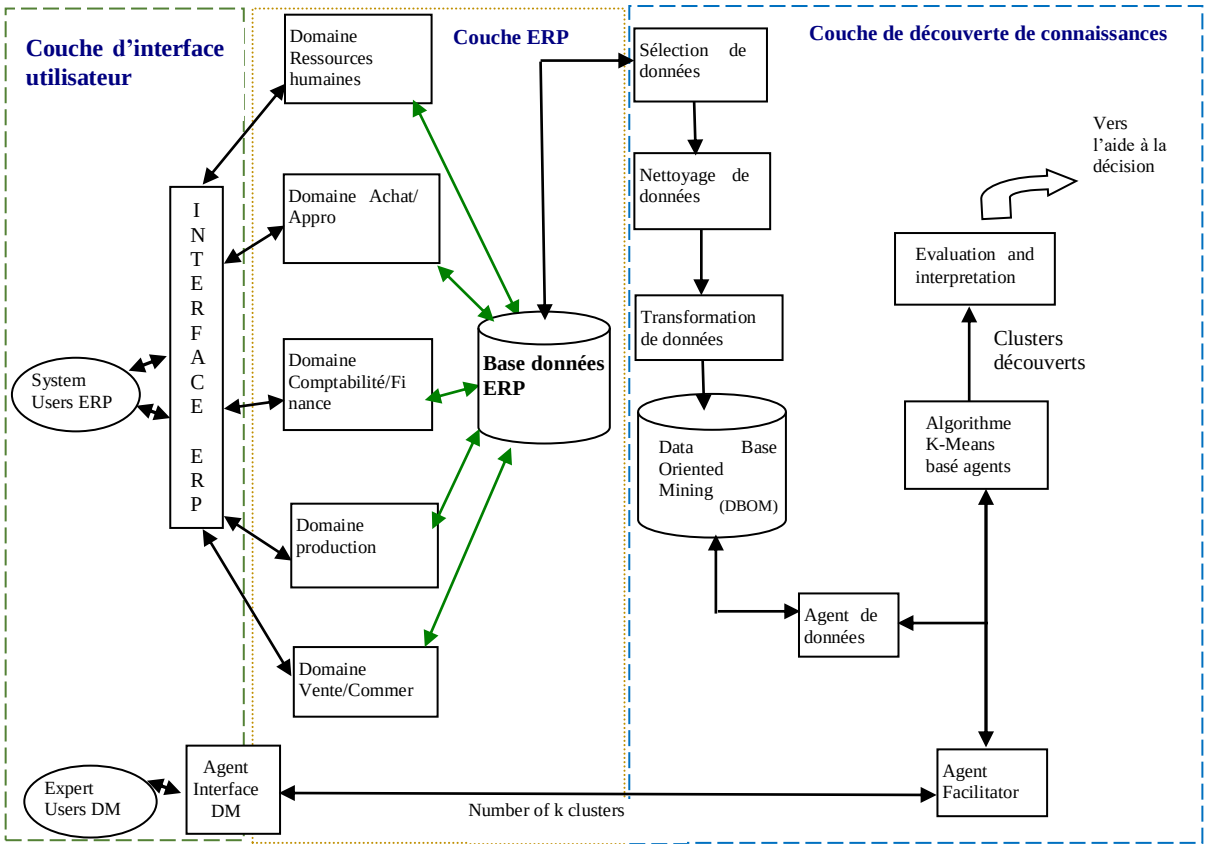


Figure IV-3 : Architecture générale du clustering de données basée agent pour l'ERP

Notre architecture proposée en Figure IV-3 est composée de trois couches telles que : la couche d'interface utilisateur, la couche ERP et la couche de découvertes de connaissances. Dans ce qui suit, nous décrivons les trois couches ainsi que les relations établies entre eux.

IV.2.5. Couche Interface Utilisateur

Elle comporte deux types d'interfaces, une première pour les utilisateurs du système ERP et une deuxième pour l'utilisateur Data miner. Le rôle de l'interface ERP est de capturer les besoins des utilisateurs ERP et de répondre le mieux possible pour réaliser une tâche bien précise de système ERP. L'agent d'interface utilisateur DM représente un point d'entrée vers la couche de la découverte de connaissances. Il interagit avec l'utilisateur Data miner pour l'assister lors de la formulation de sa requête de Data Mining telle que la saisie de nombre du k clusters souhaité, qui est nécessaire pour le lancement de l'opération du clustering de données ERP. Un autre rôle très important dans cette couche est la présentation des résultats de clustering de données ERP en face de l'utilisateur Data Miner dans le but de l'aide à la décision.

IV.2.6. Couche ERP :

La deuxième couche de notre architecture est consacrée à la modélisation du système ERP, qui est composé d'un ensemble de domaines métiers et une grande base de données ERP centrale, ayant une variété des données collectées à partir de n'importe quel domaine métier. Elle assure la communication et le partage des informations entre les différents

domaines tels que le domaine Achat/Appro, le domaine Vente/Commercial, le domaine Comptabilité/Finance…etc. Ainsi, la base de données ERP de cette couche représente une source de données central et partagée pour la couche de découvertes de connaissances afin d'entamer le processus du clustering de données ERP.

IV.2.7. Couche de découvertes de connaissances.

Elle est constituée d'un ensemble d'étapes successives pour la tâche du clustering de données ERP, dans le but de découvrir de nouvelles connaissances utiles et inconnues cachées dans la base de données ERP. Ces connaissances découvertes sont utilisées pour assister les utilisateurs DM à prendre les bonnes décisions au bon moment. Dans cette couche, les étapes suivies pour clustering de données ERP sont décrites comme suit :

1. **Sélection de données ERP** : la première étape consiste à sélectionner les attributs utiles à partir de la base de données ERP pour une tâche spécifique du clustering.
2. **Nettoyage de données ERP** : le principal but de cette étape est la normalisation des données ERP sélectionnées par l'élimination des bruits telles que les opérations de correction des erreurs, suppression des enregistrements en double et suppression des valeurs aberrantes (valeurs inhabituelles ou exceptionnelles).
3. **Transformation de données ERP** : Elle permet de transformer la structure de données ERP sélectionnées et nettoyées pour être adéquate à la tâche du clustering de données ERP. Les données transformées sont stockées dans une Base de Données Orientée Mining (DBOM).
4. **Algorithme K-Means basé agents** : Cette étape est consacré à la modélisation de la tâche du clustering de données ERP basée K-Means par l'approche du système Multi-Agents, que nous détaillerons dans la section suivante.
5. **Evaluation et interprétation** : Elle utilise des techniques de visualisation de données (histogramme, camembert, arbre, visualisation 3D) pour la découverte des modèles de données utiles. L'évaluation de ces modèles est intéressante en se basant sur des mesures données.

IV.5. Architecture fonctionnelle de l'algorithme K-Means basé agents

Le déroulement de l'étape de « Algorithme K-Means basé agents » nécessite une base de données orientée mining (DBOM) pour accomplir la tâche de clustering de données ERP. Cette base de données DBOM contient des données ERP nettoyées et transformées auparavant pour les exploiter par l'algorithme K-Means basée agents. La figure IV-4 suivante illustre l'architecture fonctionnelle de l'algorithme K-Means basé agents :

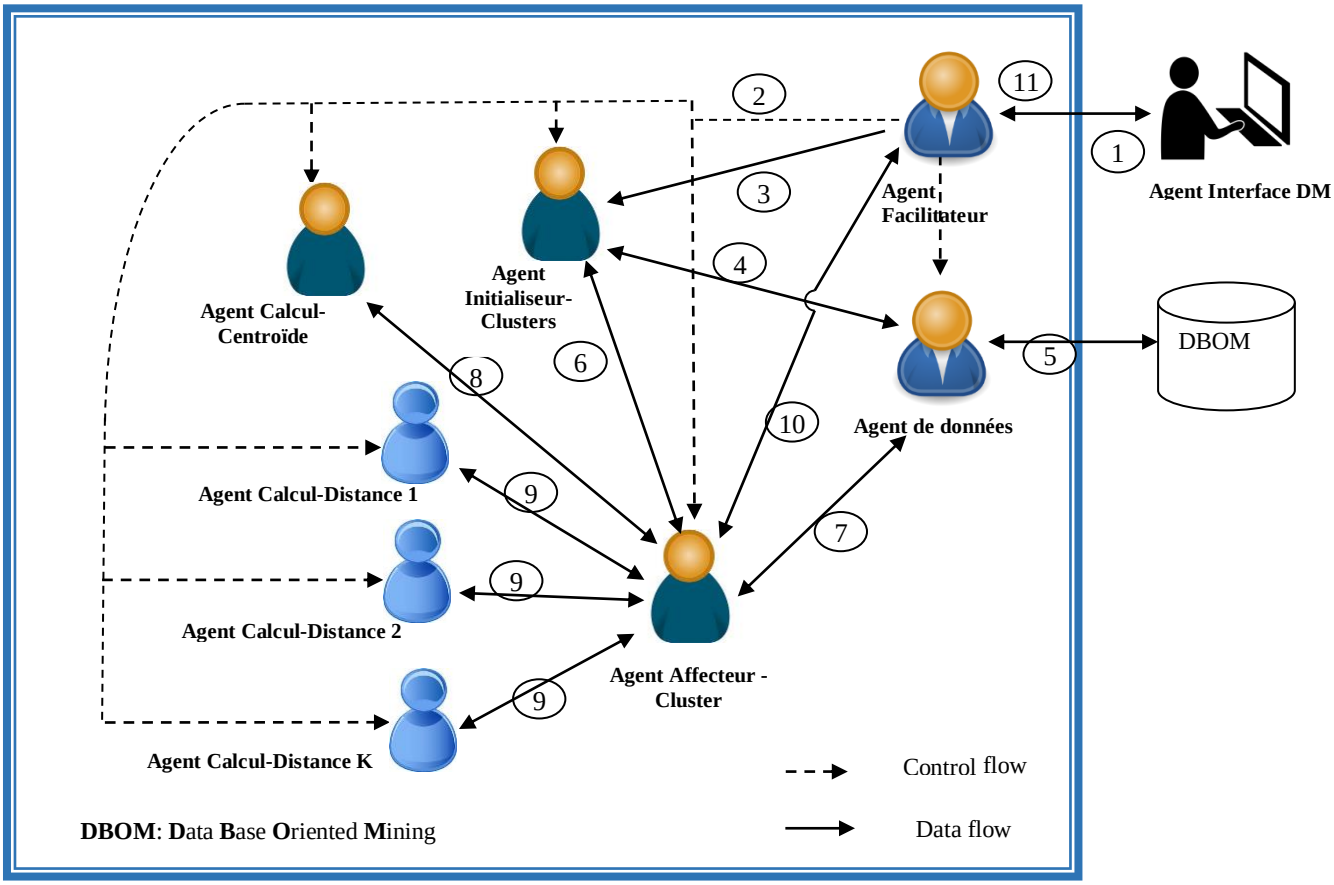


Figure IV-4 : Modèle Agent pour clustering basée K-Means

Nous nous intéressons dans cette section à la description de cette importante étape de la couche de découvertes de connaissances, qui est consacré à la modélisation de la tâche du clustering basée K-Means par le paradigme du système multi-agents. Le flux de contrôle indiqué dans la figure IV-4 représente l'activation et l'initialisation des agents du système de clustering par l'agent facilitateur. Le flux de données mentionné dans la figure ci-dessus assure la collaboration et la communication entre les agents du système. Notre architecture fonctionnelle comme elle présentée se compose d'une base de données DBOM et sept agents intelligents, à savoir : (1) Agent interface DM, (2) Agent facilitateur, (3) Agent de données, (4) Agent Initialiseur-Clusters, (5) Agent Affect-Cluster et (6) Agent Calcul-Distance, (7) Agent Calcul-Centroide.

IV.2.8. Agent d'interface DM

Cet agent est considéré comme une interface entre l'utilisateur Data Miner et la couche de découverte des connaissances. Il est responsable de la récupération du nombre de K clusters, entré par l'utilisateur Data Miner, qui est nécessaire pour le lancement du processus du clustering de données ERP basé k-means. En outre, Il permet de présenter les résultats du clustering de données ERP à l'utilisateur Data Miner. L'architecture interne de cet agent se compose de trois modules, comme illustrée dans la « Figure IV 5».

- Le module d'interface permet l'interaction avec l'utilisateur Data Miner, en présentant un ensemble de fonctionnalités sous forme d'une interface graphique.
- Le module de coordination et de planification coordonne son but local avec le but global du système en se basant sur un ensemble paramètres telles que : le nombre de K clusters saisi par l'utilisateur Data Miner et les informations reçues par le module de communication.
- Le module de communication contient des méthodes de communication inter-agents pour envoyer le nombre de K clusters à l'agent facilitateur, qui est nécessaire au démarrage de l'opération du clustering de données ERP. Il reçoit aussi les résultats de clustering de données ERP (Clusters terminaux envoyé par l'agent Affect-Cluster via l'agent facilitateur) et met à jour la base de connaissances.

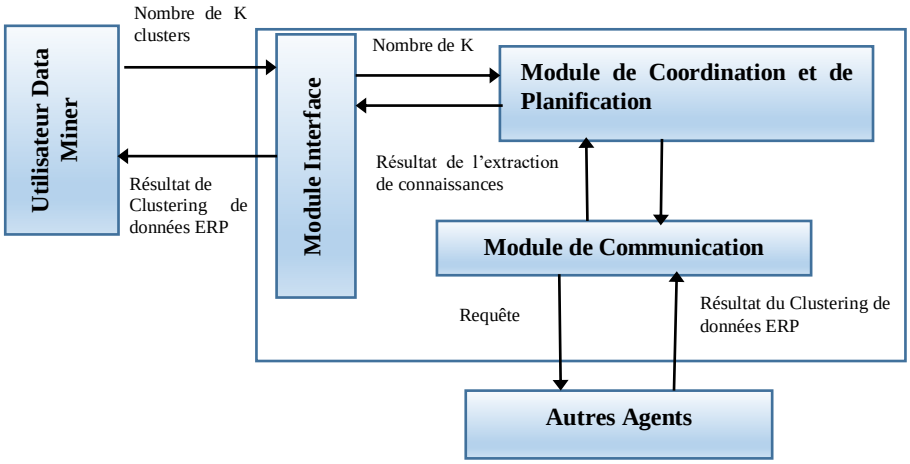


Figure IV-5 Architecture interne de l'agent d'interface DM

Le fonctionnement de cet agent décrit dans les étapes suivantes afin d'atteindre ses buts :

a) Saisie du nombre de K clusters.

b) Envoi le nombre K saisi aux agents Initialiseur-Cluster et Affect-Cluster via l'agent facilitateur pour entamer le processus du clustering de données ERP.

c) Présentation du résultat du clustering de données ERP à l'utilisateur Data Miner.

IV.2.9. Agent facilitateur

Il assure l'activation et la synchronisation des différents agents du système. En plus, il élabore un plan de travail et veille à son accomplissement. Cet agent reçoit les requêtes mining de l'agent interface DM et les transmet à l'agent Initialise-Clusters pour lancer l'opération de clustering de données ERP. Un autre rôle primordial de cet agent est la synthétisation et l'envoi du résultat de Clustering à l'agent d'interface DM pour l'affichage final. Son module d'interface est responsable à la communication inter-agents ; le module de processus contient des méthodes de contrôle et coordination de diverses tâches du système du clustering de données. Le module de connaissances contient également des méta-connaissances sur les capacités de services des autres agents du système. (Voir Figure IV 6)

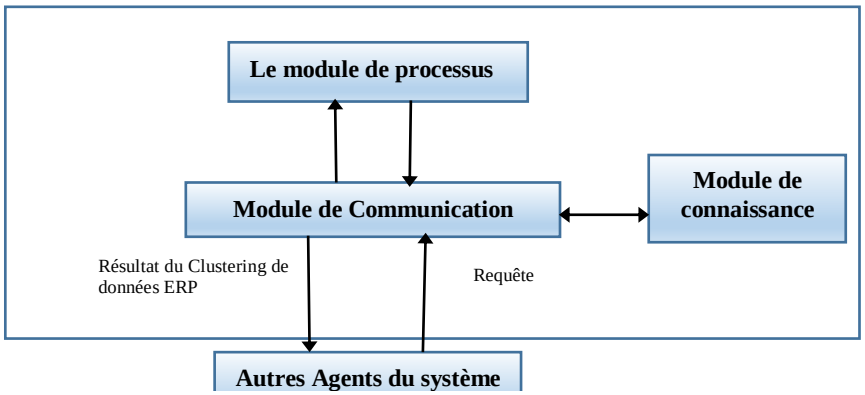


Figure IV-6 : Architecture interne de l'agent facilitateur

Les différentes fonctionnalités de cet agent sont présentées dans les étapes suivantes :

a) Activer et synchroniser les différents agents du système de clustering,

b) Recevoir le nombre de K clusters envoyé par l'agent interface DM,

c) Envoyer le nombre de K clusters reçu à l'agent Initialiseur-Cluster et l'agent Affect-Cluster pour lancer l'opération de clustering de données ERP,

d) Recevoir et transmettre les résultats finaux de Clustering à l'agent interface DM pour l'affichage finale.

IV.2.10. Agent de données

Cet agent contient des métadonnées relatives à la base de données orientée Mining (DBOM). Il s'occupe de récupérer l'ensemble de données ERP faites l'objet du clustering à partir de la base de données DBOM, qui sont nécessaires pour le fonctionnement de l'agent Initialise-Cluster et l'agent Affecte-Cluster. L'architecture interne de cet agent se compose de trois modules, citons : (Voir Figure IV 7)

- Le module d'interface qui prend en charge la communication inter-agent ainsi qu'une interface vers la base de données DBOM.
- Le module de processus qui prend en charge le contrôle et la coordination de diverses tâches de cet agent avec le système global du clustering.
- Le module d'extraction de données contient des fonctionnalités pour l'extraction des données ERP préparées, nettoyées et transformées précédemment stockées dans la base de données DBOM, en vue d'être segmentées.

Cet agent se base sur les requêtes de l'agent Initialise-Cluster et l'agent Affecte-Cluster pour extraire les données ERP nécessaires au déroulement de l'opération du clustering basée K-Means.

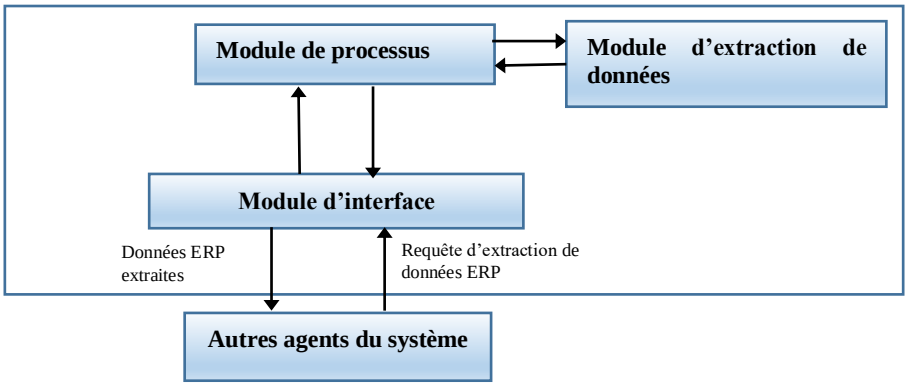


Figure IV-7 : Architecture interne de l'agent de données

L'agent de données suit les étapes suivantes afin d'accomplir ses tâches de l'extraction de données ERP :

a) Recevoir de l'agent Initialiseur-Clusters une requête pour l'extraction de k premiers enregistrements de données ERP.

b) Interroger la base de données DBOM pour extraire les k premiers enregistrements de données ERP.

c) Recevoir les requêtes de l'agent affecte-clusters pour extraire les données ERP concernées par l'opération du clustering.

d) Extraire les données ERP faites l'objet du clustering à partir de la base de données DBOM

IV.2.11. Agent Initialiseur-Clusters

Cet agent est activé par l'agent facilitateur. Son rôle est de déterminer les k premiers enregistrements de la base de données DBOM (Data Base Oriented Mining) à travers l'agent de données. Les k premiers enregistrements sont nécessaires pour engendrer les clusters initiaux. Par la suite, les coordonnées de k premiers enregistrements sont envoyées à l'agent Affect-cluster pour commencer le Clustering de données ERP. Le fonctionnement de cet agent est assuré par un module de communication et un module initialiseur des clusters qui sont illustrés dans la Figure IV 8.

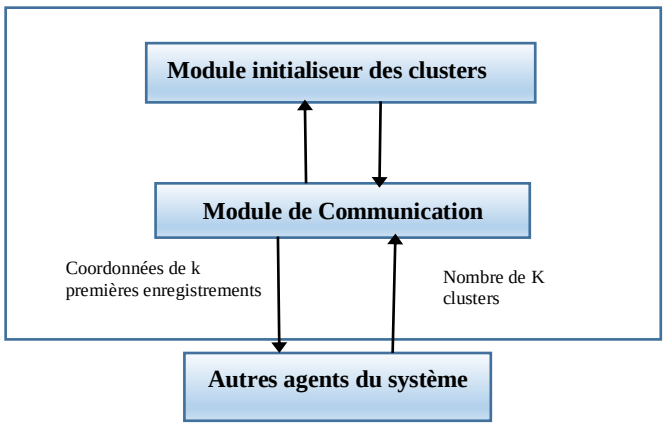


Figure IV-8 : Architecture interne de l'agent Initialiseur-Clusters

Les étapes ci-dessous montrent le fonctionnement de l'agent Initialiseur-Clusters :

a) Recevoir le nombre de K clusters à partir de l'agent facilitateur.

b) Interroger l'agent de données pour récupérer les k premiers enregistrements de la base de données DBOM.

c) Envoyer les coordonnées de k premiers enregistrements de données ERP à l'agent Affect-Cluster pour créer les clusters initiaux.

IV.2.12. Agent Affecteur-Clusters

Il affecte les données ERP concernées par l'opération du clustering dans les clusters adéquats. Son rôle principal est la création d'une nouvelle partition pour chaque groupe d'enregistrements de données ERP dont leurs centres de gravité sont plus proches. Cet agent se base sur une fonction de distance qui est effectuée par l'Agent Calcul-Distance et le calcul de Centreoid des clusters qui est effectué par Agent Calcul-Centroid. Il est fondé sur trois modules fondamentaux (tels que le module de communication, le module de choix de cluster, et module de processus) afin d'accomplir ses tâches d'affectation de données ERP dans les meilleurs clusters (Voir Figure IV 9).

- Le module de communication permet la communication inter-agents. Il reçoit le nombre de k clusters envoyé par l'agent facilitateur et les coordonnées de k premiers enregistrements choisis par l'agent Initialiseur-Cluster comme clusters initiaux. En plus, il reçoit les centres des clusters recalculés par Agent Calcul-Centroid et distances calculées par les distances calculées de l'agent Calcul-Distance. Aussi, ce module permet de capter les données ERP faites l'objet de clustering via l'agent de données.
- le module de processus contient des méthodes de contrôle et de coordination de diverses tâches de cet agent avec le système de clustering.
- le module d'affecte cluster : il se base sur les informations obtenues par le module de communication telles que les coordonnées de K clusters et les données ERP faites l'objet de clustering, ainsi que les centres des clusters et les distances calculées pour affecter chaque enregistrement de données ERP au cluster le plus adéquate (où la distance entre l'enregistrement et le centre de cluster est minimale).

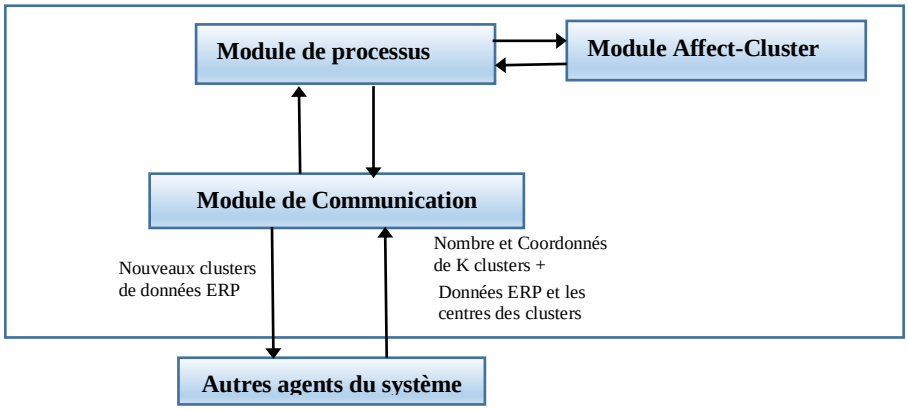


Figure IV-9 : Architecture interne de l'agent Affecteur-Clusters

Le fonctionnement de l'agent Affect-Cluster est décrit par les étapes suivantes afin de réaliser ses tâches :

a) Recevoir les coordonnées de k clusters initiaux envoyé par l'agent Initialiseur-Clusters.

b) Récupérer les données ERP de la base de données DBOM via l'agent de données, en vue d'être segmentées.

c) Envoyer les centres des clusters (dans la première itération, les coordonnées de k clusters initiaux représentent les centres des clusters) ainsi que les données ERP faites l'objet de clustering à l'agent Calcul-Distance.

d) Recevoir les distances calculées de l'agent Calcul-Distance.

e) Choisir la distance minimale à partir des distances reçues.

f) Affecter chaque enregistrement de données ERP au plus proche cluster en se basant sur la distance minimale choisie).

g) Envoyer les nouveaux clusters à l'agent Calcul-Centroïde pour recalculer leurs nouveaux centres de clusters.

h) Recevoir les nouveaux centroïdes envoyés par l'agent Calcul-Centroïde.

i) Relancer ce processus à nouveau par rapport aux nouveaux centres reçus (Revenir à l'étape 'c' et répéter toutes les étapes).

j) Répéter toutes les étapes précédentes jusqu'à ce que les distances calculées ne changent pas ou le nombre d'itérations est achevé.

IV.2.13. Agent Calcul-Centroide

Il travaille selon les besoins de l'agent Affecte-Clusters dans le but de calculer les centres des clusters de données ERP extraites. L'architecture interne de cet agent comporte deux modules fondamentaux pour calculer les centroïdes des clusters tels que le module de communication et le module de Calcul des Centroïdes.(Voir Figure IV 10)

- Le module de communication est responsable sur la l'interaction avec l'agent Affecte-Clusters. Il reçoit en entrée les clusters crées récemment et il envoie les Centroïdes calculés vers l'agent Affecte-Clusters.
- Un module de calcul des Centroïdes : il prend en charge le calcul des centres des clusters envoyés par l'agent Affecte-Cluster et reçus par le module de communication. Ce module utilise la moyenne pondérée pour calculer les centroïdes des clusters.

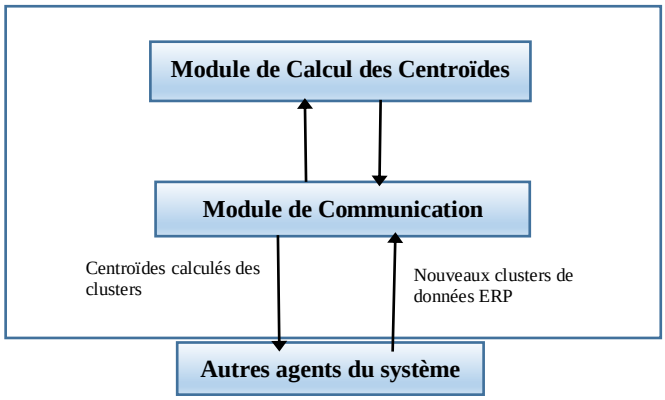


Figure IV-10 : Architecture interne de l'agent Calcul- Centroïdes

Les étapes suivantes montrent le fonctionnement de l'agent Calcul-Centroïde pour calculer les centres des clusters :

a) Recevoir la requête de l'agent Affecte-Clusters contenant les coordonnées des nouveaux clusters constituées.

b) Calculer le centroïdes de chaque cluster reçu en utilisant la moyenne pondérée.

c) Envoyer les centroïdes calculés des clusters à l'agent Affecte-Clusters.

IV.2.14. Agent Calcul-Distance

Il se déclenche par l'Agent Affect-Cluster selon ses besoins. Il travaille en collaboration avec ce dernier dans le but de calculer les distances entre les enregistrements de données ERP et les centres des clusters qui sont déjà calculés et fournis par l'agent Calcul-Centroid. Selon la demande de l'agent Affecte-cluster, K instances de l'agent Calcul-Distance sont lancés en parallèle pour calculer les distances entre chaque enregistrement reçu de données ERP et les K Centroides de clusters (C1, C2 …Ck) calculés par l'agent Calc-Centroid. L'architecture interne de l'agent Calcule-Distance est composée de deux modules principaux citons le module de communication et le module de calcul de distance : (Voir Figure IV 11)

- Le module de communication prend en charge l'interaction avec l'agent Affecte-Clusters. Son rôle est de recevoir des messages de l'agent affecte-clusters contiennent les K Centroide de clusters et les données ERP concernées par l'opération du clustering. Aussi, il envoie les résultats des calculs de distances à l'agent affecte-clusters pour accomplir ses tâches.
- le module de calcul de distance se charge de calculer les distances entre chaque enregistrement de données ERP et les centoïdes de tous les clusters.

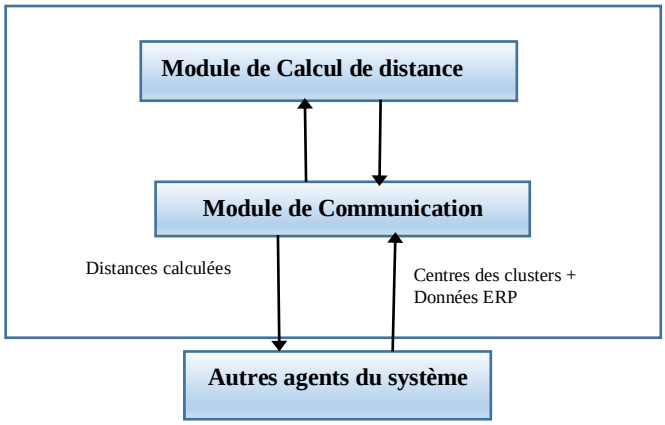


Figure IV-11 : Architecture interne de l'agent Calcul-Distance

Les étapes suivantes présentent le fonctionnement de l'agent Calcul-Distance :

d) Recevoir la requête de l'agent Affect-Cluster contenant les coordonnées des centres des clusters et les données ERP à segmenter.

e) Calculer la distance de chaque enregistrement de données ERP par rapport à tous les centres des clusters.

f) Envoyer le résultat du calcul de distance à l'agent Affect-Cluster.

IV.6. Mécanisme de coopération dans l'approche « CMAAC-ERP »

Dans cette section nous rappelons les différentes méthodes utilisées pour la coopération entre les agents (Voir la section III.3.4) : le regroupement et la multiplication, la communication, la spécialisation, la collaboration par partage de tâches et de ressources, la coordination d'action, la résolution des conflits par arbitrage et négociation. La méthode de coopération entre les agents est considérée comme une caractéristique primordiale dans un système multi-agents. En effet, une résolution distribuée d'un problème est le résultat de l'interaction coopérative entre les agents.

Dans notre approche « CMAAC-ERP » nous avons adopté pour « la communication par l'envoi des messages » comme une méthode de coopération entre les agents afin de réaliser les tâches du clusering de données ERP. Chaque agent décide son comportement pour satisfaire le but global du système afin de regrouper efficacement les données ERP métiers dans des clusters de données homogènes.

IV.7. Communication des agents dans « CMAAC-ERP »

Puisque les agents de notre approche « CMAAC-ERP » sont de type cognitif, la communication entre les agents de notre système s'effectuent par l'envoi de messages. Elle permet aux agents de partager la charge d'un problème de clustering de données ERP en le subdivisant en sous problèmes.

Les SMA étant basés sur la communication par l'envoi des messages, se distinguent par le fait que chaque agent contient une représentation propre et locale de l'environnement SMA. Chaque agent va alors interroger les autres agents sur cet environnement ou leur envoyer des informations sur sa propre perception des choses. [199].

Le langage de communication que nous avons adopté pour l'envoi de messages entre les agents de l'approche « CMAAC-ERP » est le langage ACL (Agent Communication Language) dont la spécification de FIPA (Foundation for Intelligent Physical Agents) permet une interopérabilité maximale entre les agents. D'ailleurs, la communication par l'envoi de messages se fait en deux modes : soit en mode point à point, ou en mode diffusion [179]. Dans notre proposition nous utilisons le mode « point à point » pour la transmission des messages entre les agents car le type d'agent utilisé dans « CMAAC-ERP » est de type cognitif, l'agent émetteur de message doit connaître avec précision l'adresse de(s) agent(s) destinataire(s). La syntaxe d'un message FIPA-ACL est illustrée comme suit :

```
(request
:sender Agent_A
:receiver Agent_B
:content
(….)
:in-reply-to action
:replay-with
reponse
:language FIPA-SL0)
```

Figure IV-12 Syntaxe d'un message FIPA-ACL

IV.8. Scenario de fonctionnement des agents du système

Le diagramme de séquence suivant (IV-10) décrit le processus de fonctionnement des agents de l'approche proposée « CMAAC-ERP » pour le clustering de données ERP basé sur K-means.

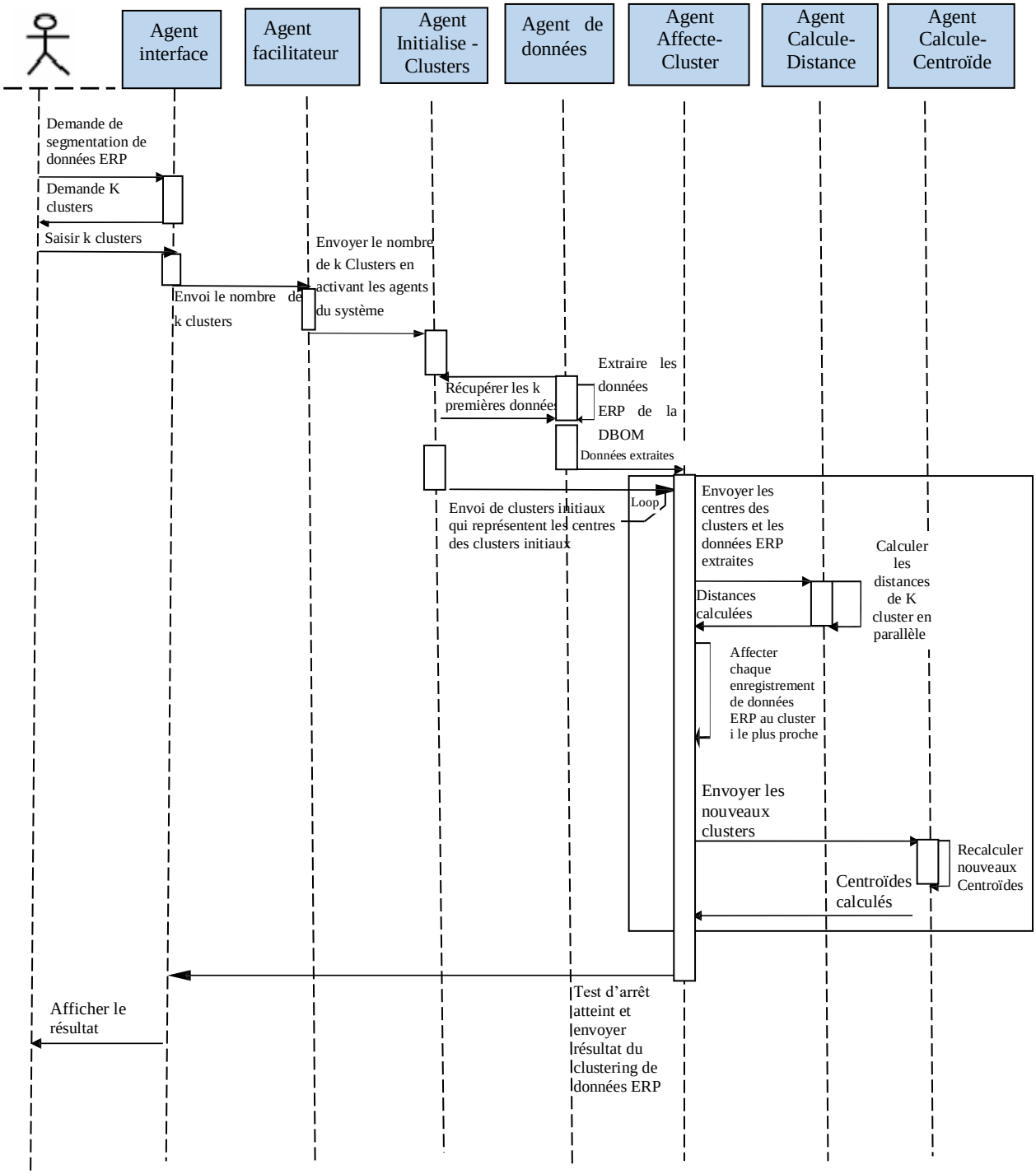


Figure IV-13 : Scenario de fonctionnement des agents du système

L'utilisation du SMA dans l'approche proposée s'avère idéale pour effectuer la tâche du clustering de données ERP d'une manière parallèle et distribuée dans le but de réduire le

temps nécessaire à la segmentation de données ERP. En premier lieux, la tâche de l'extraction des données ERP de la base de données DBOM se fait parallèlement avec la tâche de la construction des groupes initiaux pour gagner un peu de temps lors de cette première étape. En outre, le déroulement de la tâche de l'affectation des données ERP dans les clusters adéquats est effectué en parallèle avec la tâche de calcul de Centroid et celle du calcul des distances. En effet, pour chaque itération de l'algorithme K-Means à base d'agents, les centroids calculés (C1, C2, …, CK) sont reçus par l'agent affect-cluster, celui-ci lance plusieurs agents de calcul des distances (de 1...K Agents) d'une façon parallèle. Ensuite, il envoie chaque centre du clsuter Ci à l'agent de calcul des distances correspondant (de 1...k) et les données ERP à segmenter. Cette architecture à base d'agents supporte l'exécution des tâches en parallèle, notamment pour le calcul des distances nécessaire à la tâche du clustering de données ERP.

IV.9. Analyse comparative sur les approches relatives au clustering basées agents

Dans cette section, il est utile de comparer notre approche proposée « *CMAAC-ERP* » avec les travaux présentés dans le chapitre III pour le clustering de données à base d'agents. Notre comparaison repose sur un ensemble de critères à savoir :

1. Type d'architecture de système (centralisée, distribuée)
2. Tâches Data Mining effectuées par le système
3. Communication entre les agents et le processus Data Mining (avant, pendant, après l'extraction des connaissances)
4. Réutilisation des algorithmes Data Mining existants sans modification (Oui, Non)
5. Le choix de la technique et l'algorithme data mining
6. Interactivité du système avec l'utilisateur
7. Nature des sources de données
8. Type du parallélisme dans le système (parallélisme de données ou de tâches)
9. Le type de synergie entre les agents et le Data mining
10. Le type de synchronisation des agents

Tableau IV-1 : Table comparative des travaux présentés du clustering à base d'agents

	PADMA [139]	**PAPYRUS** [140]	**Approche de Chaimontree** [110]	**MAD-IDS**[142]	**Notre approche CMAAC-ERP** [202] et [221]
Type d'architecture ?	Centralisée	Distribuée	Distribuée	Distribuée	Distribuée
Tâches effectuées par le système ?	Clustering	Clustering	Clustering	Clustering Règles -assoc.	Clustering
Communication des agents (avant, pendant, après) de le processus Data Mining) ?	Au début, lors et à la fin de processus DM	Au début, lors de et à la fin de l'extraction	Lors de processus de Data Mining	Au Début et fin Du processus	Lors de processus de Data Mining
Réutilisation des	Oui	Oui	Oui	Non	Non

algorithmes DM ?					
Choix de la technique et l'algorithme DM ?	Prédéterminés	Adaptatifs	Adaptatifs	Prédéterminés	Prédéterminés
Interactivité avec l'utilisateur DM ?	Oui	Oui	Oui	Oui	Oui
Nature des sources de données ?	Distribuée	Distribuée	Distribuée	Distribuée	**Centralisée de l'ERP**
Type du parallélisme ?	Données	Données	Données	Tâches	**Tâches**
Synergie des agents avec le Data mining ?	Coopération	Coopération	Coopération	Coopération	Coopération
Nombre d'agents Synchronisés ?	Plusieurs agents	Un seul agent	Plusieurs Agents	Plusieurs agents mobiles	Plusieurs Agents

Par l'analyse verticale de la table comparative : nous constatons que le système PAPYRUS présente une architecture intéressante pour le Data Mining basé sur le clustering distribué, mais avec une mauvaise synchronisation entre les agents, chose qui dégrade la performance du système. Aussi, le système PADMA repose sur une architecture standard Data Mining distribué basé agents permettant une interactivité forte avec les utilisateurs, une réutilisabilité des algorithmes DM avec une bonne prise en charge des problèmes de synchronisation. Par contre, l'approche IDS (MAD-IDS) est fondée sur les tâches du clustering, les règles d'association, plusieurs agents mobiles et un parallélisme de tâches. Ce système permet l'interactivité avec l'utilisateur DM, mais son architecture manque de standardisation dans son domaine d'application qui le rend non réutilisable. L'approche de Chaimontree possède aussi une architecture fortement distribuée basée agents. Elle peut s'adapter exceptionnellement à la tâche de classification ou aux règles d'association, avec une coopération parfaite entre les agents et la technologie de Data Mining.

Par l'analyse horizontale de la table comparative : nous observons que la nature des sources de données utilisées dans les différentes approches présentées est de type distribuée. Dans notre proposition pour le clustering à base d'agents, nous utilisons une source de données de type centralisée, car notre approche proposée « CMAAC-ERP » est axée sur le système ERP qui permet une gestion homogène et cohérente du SI d'une entreprise, autour d'une grande base de données centrale et partagée. La centralisation de données ERP sur une seule base de données est parmi les caractéristiques primordiales du progiciel ERP à prendre en considération dans notre proposition. D'un autre côté, la plupart des travaux présentés utilisent le parallélisme de données dans leurs architectures puisque les données sont distribuées. Or, l'approche « CMAAC-ERP » utilise le parallélisme de tâches pour distribuer la complexité du clustering sur plusieurs agents cognitifs sachant que chacun d'eux modélise une partie du système du clustering de données à base K-Means.

Pour récapituler, la plupart des approches présentées dans la section III.5.1 utilisent des sources de données de nature distribuées. Contrairement à notre approche « CMAAC-ERP » qui est fondée sur une source de données de type centralisée, car elle est basée principalement sur le progiciel ERP où la centralisation de données est nécessaire pour une gestion homogène

et cohérente du système d'information de l'entreprise. En outre, la plupart des travaux présentés dans section III.5.1 utilisent le parallélisme de données dans leurs architectures dues à la distribution de données sur plusieurs sites distribués. Par contre, notre approche proposée utilise le parallélisme des tâches pour distribuer la complexité du clustering de données ERP entre plusieurs agents coopératifs et autonomes, modélisant chacun une partie du système de clustering de données à base K-Means. Cela vise à accélérer la tâche de clustering de données ERP afin d'aider les managers pour prendre des bonnes décisions en temps opportun.

IV.10. Conclusion et perspectives

Dans le cadre de ce chapitre, nous avons apporté une solution au problème de complexité d'extraction de connaissances au niveau du système ERP. Pour y parvenir, nous nous sommes basés sur le couplage technologique entre le Data Mining et le paradigme du système multi-agents. Nous avons montré que ces deux domaines sont complémentaires et peuvent évoluer dans le cadre d'un processus unique. Leur association est capable de faciliter le processus d'extraction de connaissance à partir d'une grande base de données ERP.

En effet, nous avons proposé une approche multicouche « CMAAC-ERP » pour la segmentation d'un grand ensemble de données ERP, tout en se basant sur la méthode du clustreing « k_Means » et les systèmes multi-agents. Ceux-ci offrent des caractéristiques intrinsèques telles que : l'autonomie, la modularité, la distribution et l'intelligence pour minimiser le temps immense de calcul nécessaire à la segmentation d'une grande quantité de données métiers ERP.

Enfin, et pour implémenter un système multi-agent il est nécessaire d'utiliser une plateforme multi-agent. Dans le cadre de ce travail, la plate-forme JADE basée sur Java semble être la plus appropriée pour développer l'approche proposée car elle repose sur des agents cognitifs. Cette plateforme offre un ensemble complet de services et d'agents répondant aux spécifications du FIPA.

Chapitre V : Une approche à base d'agents pour la distribution des règles d'association par métier à partir d'un ERP

(**AADARB-ERP):** A new **A**pproach **A**gent-based for **D**istributed of **A**ssociation **R**ules by **B**usiness to improve the decision process in **ERP** systems).

V.1. Introduction

Le système de gestion intégré (ERP) gère plusieurs domaines fonctionnels ou opérationnels d'une entreprise, autour d'une base de données importante et centrale. Jour après jour et avec l'usage quotidien, la base de données ERP rassemble une quantité énorme de données qui cachent des connaissances pertinentes sur les activités internes ou externes de l'entreprise, mais elles restent moins exploitées. Pour répondre à ce besoin, les entreprises utilisent de nouvelles techniques de Data Mining, qui proposent l'utilisation d'un ensemble d'algorithmes pour analyser et extraire les connaissances cachées dans les grandes bases de données. Ces technologies peuvent être intégrées aux systèmes ERP pour enrichir la fonction décisionnelle (Business intelligence : BI) via l'extraction de nouvelles connaissances pour assister les managers à prendre les bonnes décisions en temps opportun.

Plusieurs approches et plateformes ont été proposées pour atteindre ces objectifs de décision qui sont relativement efficaces selon le domaine d'application. Dans cette vision, et pour découvrir des connaissances pertinentes dans un système ERP, nous avons appliqué l'algorithme de l'extraction des règles d'association sur une base de données ERP. Néanmoins, cette base de données ERP augmente au fil du temps en raison des utilisations quotidiennes, ce qui rend l'extraction des règles d'association sur celle-ci moins efficaces et nécessite une charge de travail et un temps d'exécution considérable. Dans ce chapitre, nous nous intéresserons au technique de l'extraction des règles d'association, qui se déroule en fonction de deux phases importantes : (1) la découverte d'ensembles d'items fréquents, et (2) la génération des règles d'association. La première phase est la plus coûteuse en termes de temps d'exécution car elle nécessite plusieurs parcours dans la base de données pour calculer le support des items et la confiance. La solution de ce problème consiste à essayer de minimiser le temps d'exécution consommé par ces algorithmes. Les chercheurs ont utilisé plusieurs paradigmes pour rendre ces algorithmes de Data Mining plus efficaces. Aujourd'hui, l'intégration du paradigme du système multi-agents dans le processus d'exploration de données est considérée comme une technologie puissante et très efficace dans la mesure où la complexité du traitement des données sera répartie sur un ensemble d'entités communicantes, autonomes et réactives avec des compétences appelées agents.

Dans ce contexte, plusieurs approches basées agents pour l'extraction des règles d'association ont été développées pour alléger l'exécution de ces algorithmes (Voir la section III.5.2.1). Cependant, la plupart de ces approches sont destinées aux architectures de type distribué, ce qui suppose que les données traitées sont réparties sur plusieurs bases de

données, où les données doivent être traitées sur leurs sites, en raison de plusieurs contraintes telles que : le coût de stockage, la communication, et la sécurité. Par contre, notre approche proposée « AADARB-ERP » utilise des données centralisées due à la nature de la base de données ERP. Elle propose une architecture à base d'agents pour la distribution des règles d'association par métier à partir d'une base de données ERP.

Ce chapitre commence par une introduction générale en montrant la motivation et les objectifs de l'approche proposée. Ensuite nous présentons l'architecture générale du système développée et l'architecture du système multi agents proposée pour l'extraction des règles d'association à partir du système ERP. Nous décrivons par la suite, l'architecture fonctionnelle des agents du système en expliquant leurs interactions. Après, nous exposons une analyse comparative entre notre approche proposée « AADARB-ERP » et les travaux existants selon un certain nombre de critères. Puis, nous faisons une brève présentation de l'environnement de développement et les interfaces de l'application développée. Enfin, nous clôturons par une expérimentation et une conclusion.

V.2. Objectifs de l'approche proposée «AADARB-ERP »

Les objectifs de notre approche «AADARB-ERP » s'articule autour des points suivants :

1. Réduction du temps global consommé par la tâche de l'extraction des règles d'association métiers à partir du système ERP.
2. Amélioration de la qualité de présentation de connaissances vis-à-vis de l'utilisateur Data Miner, par l'usage des règles plus claires offertes par la technique de règles d'association.
3. Limitation de l'espace de recherche des règles d'association via le partitionnement de données métiers ERP (qui concerne l'opération de Data Mining) sur des parties de données moins volumineuses afin de paralléliser l'extraction des règles d'association métiers.
4. Distribution de ces parties de données sur plusieurs machines distantes pour rendre le traitement de données ERP en parallèle distribué, ce qui réduit le temps global de l'exécution des règles d'association.
5. Partitionnement horizontal de données ERP par métier, permet de produire des règles d'associations par métier plus significatives et plus cohérentes.

V.3. Présentation de l'approche « AADARB-ERP »

Dans ce qui suit, nous présentons l'architecture générale de l'approche proposée «AADARB-ERP » et son architecture multi-agents :

V.3.1. Architecture générale de « AADARB-ERP »

Une des caractéristiques du système ERP est la centralisation de données sur une grande base données qui permet une gestion homogène et cohérente du système d'information de l'entreprise. Cependant, cette caractéristique est indésirable lorsqu'on cherche à mettre en place un système de Data Mining plus efficace pour l'extraction des règles d'association afin

de répondre aux objectifs décisionnels de l'entreprise dans les meilleurs délais. Dans ce cadre, le paradigme multi-agents propose des concepts particulièrement intéressants pour le développement des plateformes Data Mining tels que : Organisation dynamique, autonomie de contrôle, décentralisation, collaboration, réutilisation, …etc. Les systèmes multi-agents (SMA) sont actuellement très largement utilisés, notamment pour la modélisation et l'implémentation des systèmes complexes qui nécessite l'interaction entre plusieurs entités. En outre, l'approche multi-agent a l'avantage de garder les informations de chaque agent entièrement privé jusqu'à ce qu'il y ait une demande précise pour une partie des informations ou des connaissances. Dans ce contexte, nous proposons notre approche « AADARB-ERP » pour l'extraction des règles d'association par métier via une grande base de données ERP.

Notre approche proposée « AADARB-ERP » est basée sur une architecture distribuée, qui est capable également de rendre l'exécution du processus d'extraction des règles d'association en parallèle et distribuée à travers d'une grande base de données ERP. Elle permet de partitionner les données ERP par métier sur des partitions de données moins volumineuses. Après, les données de chaque métier seront traitées séparément par un agent de règles d'association et sur une machine différente. L'architecture proposée (voir Figure V-1) est capable de résoudre le problème de la complexité de traitement de l'algorithme de règle d'association sur l'énorme volume de données ERP par la distribution de traitement sur plusieurs agents autonomes et coopératifs. Chaque agent de règle d'association est chargé de traiter les données d'un métier spécifique sur une machine différente. Notre architecture permet ainsi une manipulation parallèle et flexible des données ERP à travers l'aspect décentralisé du système multi-agents dont le but est de diminuer le temps d'exécution global d'extraction des règles d'association métiers.

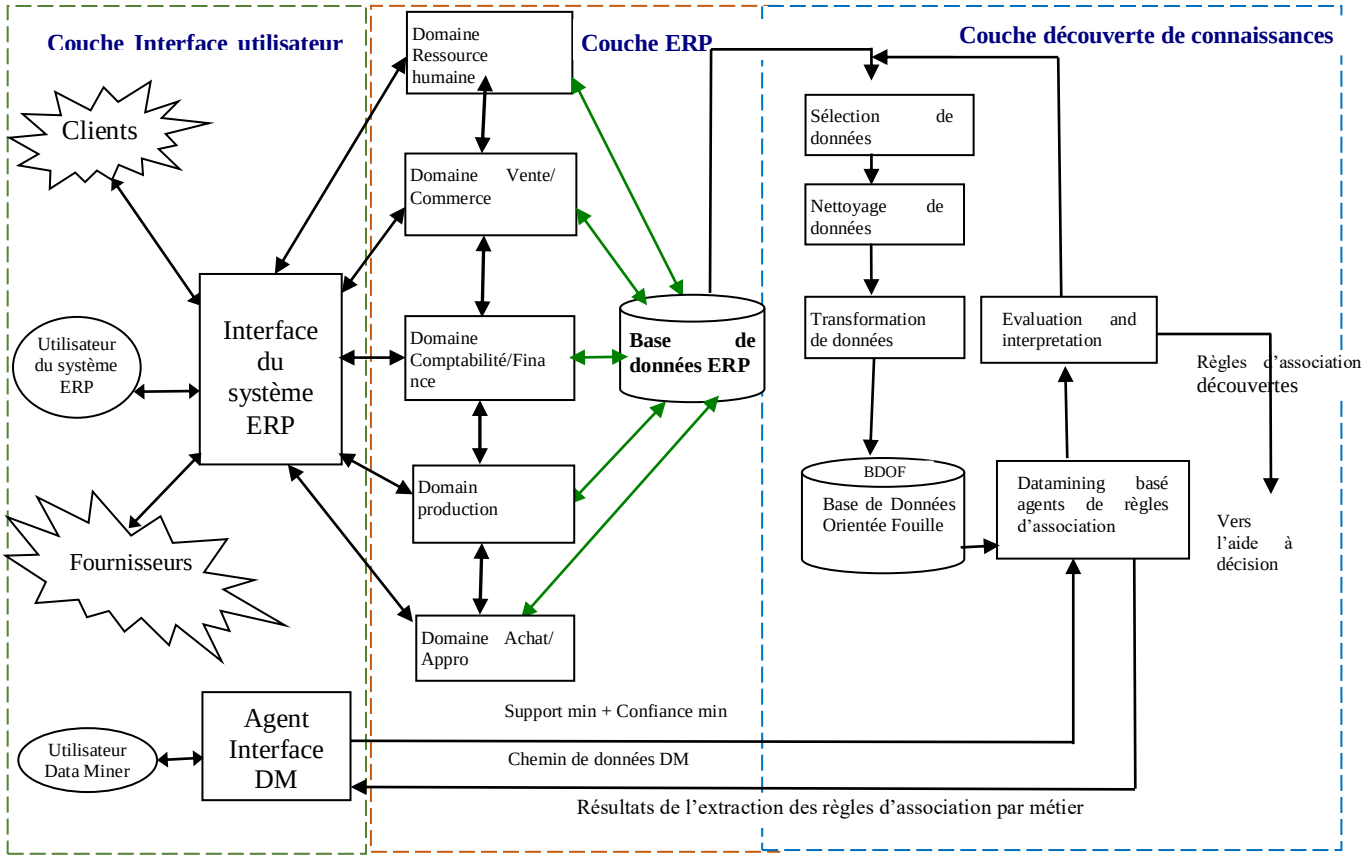


Figure V-1 : Architecture Data Mining basée agent de règles d'association pour l'ERP

L'architecture générale proposée dans la Figure V 1 est multicouches, dont chaque couche possède ses propres fonctionnalités avec des responsabilités précises. Voici une description plus précise des différentes couches et leurs relations.

A. Couche Interface Utilisateur

Cette couche représente l'interface du système avec les utilisateurs ERP, l'utilisateur Data Miner et les organismes externes (fournisseurs, clients, banques …etc.). Principalement, elle permet de capturer le but de l'utilisateur afin de répondre le mieux possible à son besoin. L'interface ERP interagit avec les utilisateurs de l'ERP pour les aider à effectuer une tâche bien précise dans leurs domaines tels que : le domaine ressource humaine, domaine vente/ commerce, domaine comptabilité/finance, domaine achat/ appro et domaine production. L'agent d'interface DM interagit avec l'utilisateur Data minier pour l'assister lors de la formulation de la requêtes de Data Mining telles que la saisie du support minimum et la confiance minimale ainsi que la précision du chemin de données ERP qui font l'objet du Data Mining. Cette couche représente aussi l'interface directe avec la couche « découverte des connaissances », qui envoie les paramètres nécessaires pour le lancement du processus de Data Mining afin d'extraire les résultats demandées sous forme des règles d'association métiers. Un autre rôle très important de cette couche est la présentation des résultats d'extraction des règles d'association en face de l'utilisateur Data Miner pour l'aide à décision.

B. Couche ERP :

C'est une couche intermédiaire, composée d'un ensemble de métiers et une base de données ERP unique et centrale, ayant une variété des données collectées à partir de n'importe quel domaine de l'ERP. Elle assure la communication et la collaboration entre les différents domaines tels que le domaine Achat/Appro, le domaine Vente/Commercial, le domaine Comptabilité/Finance…etc. Cette couche joue aussi un rôle fondamental pour la communication avec la couche interface utilisateur afin de répondre aux requêtes des utilisateurs ERP et la couche de découverte de connaissances à partir de la base de données ERP.

C. Couche de découverte de connaissances :

Cette couche modélise le processus complet de l'extraction de connaissances, qui passe par un ensemble des étapes successives, à savoir : Sélection de données, Nettoyage de données, transformation de données, Data Mining basé agents de règles d'association, évaluation et interprétation des résultats. Le but principal de cette couche est de découvrir des nouvelles connaissances utiles et inconnues, cachées dans la base de données d'ERP. Ces nouvelles connaissances sont représentées sous forme de règles d'association par métier pour une utilisation ultérieure, dont le but est d'assister les décideurs à prendre les bonnes décisions au bon moment. Un autre rôle très important dans cette couche est la modélisation de l'étape de découverte proprement dite, Data Mining basée sur les règles d'association par le Système Multi-Agents qui sera détaillé dans la prochaine section.

V.3.2. Architecture du système multi-agents

Après avoir préparé et transformé les données ERP qui font l'objet de Data Mining, dans la Base de Données Orientée Fouille (BDOF), nous modélisons l'étape « Data Mining basée règles d'association » en utilisant le paradigme du système multi-agents (SMA). Ce dernier fournit des caractéristiques importantes pour distribuer l'expertise sur un ensemble d'agents cognitifs modélisant chacun une tâche du système de l'extraction des connaissances. Cette architecture est illustrée dans la Figure V-2 où chaque agent possède une compétence particulière utilisée pour résoudre une partie du problème global de l'extraction des règles d'association à partir d'une grande base de données ERP.

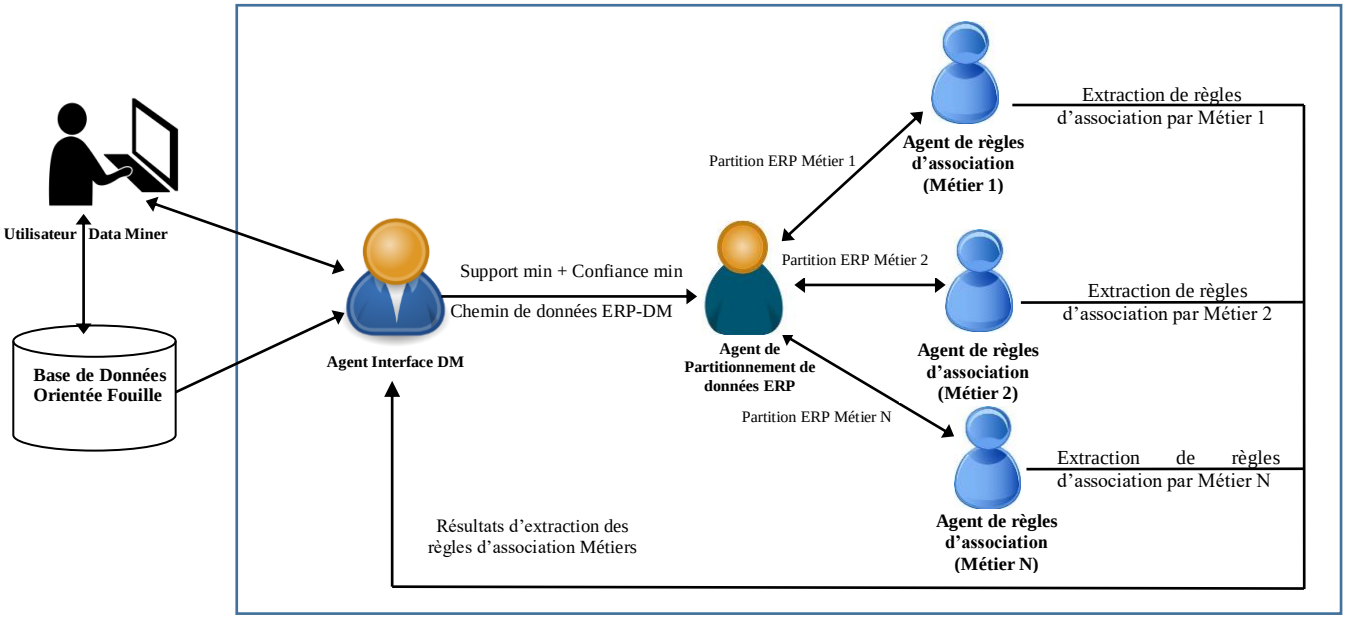


Figure V-2 : Architecture du Système Multi-Agents proposée

L'avantage principal de l'architecture SMA proposée réside dans la capacité d'optimiser l'exécution de l'algorithme de règles d'association autour d'une grande base de données ERP, tout en se basant sur la distribution, la flexibilité, la réutilisation, l'évolutivité et l'efficacité de systèmes multi-agents. En effet, l'agent interface DM fournit les paramètres nécessaires au lancement de l'architecture SMA. Par la suite, l'agent de partitionnement de données divise horizontalement la quantité de données ERP par métier, sur des sous-ensembles de données (partitions) moins volumineuses selon un critère introduit par l'utilisateur Data Minier. Ce partitionnement est important dans le but est de distribuer la charge du traitement de l'algorithme de règles d'association à plusieurs sous-traitements distribués. Ceux-ci fonctionnent en parallèle sur ces partitions de données ERP et sur différentes machines distantes. Ensuite, l'agent de partitionnement de données crée en parallèle un ensemble des agents de règles d'association selon le nombre des partitions de données ERP. Puis, les agents des règles d'association seront exécutés en parallèle pour la génération des règles d'association par métier. Par la suite, Ils envoient les résultats de l'extraction des règles d'association par métier à l'agent Interface Data Miner pour l'affichage final.

Le principal but de l'architecture SMA de notre approche «AADARB-ERP» est le lancement de plusieurs processus d'extraction des règles d'association par métier

simultanément afin de réduire de temps global consommé par la génération des règles d'association métiers sur la base de données ERP.

Dans le système SMA proposé, nous avons utilisé des agents cognitifs parce qu'ils permettent d'effectuer les tâches d'extraction des règles d'association sur plusieurs machines distantes, d'une manière parallèle distribuée. Ils sont capables de prévoir et d'anticiper les actions de l'environnement de l'extraction de connaissances. Leurs caractéristiques clés sont :

i. L'autonomie permet à l'agent le contrôle de ses actions et ses états internes,
ii. La flexibilité permet à l'agent de percevoir et de prendre les initiatives appropriées aux changements de l'environnement, pour cela il doit être :
    a. Proactif : il présente un comportement proactif et opportuniste afin de prendre des initiatives ;
    b. Sociable : capable d'interagir avec les autres agents quand la situation le demande afin de réaliser leurs tâches ou d'aider les autres agents à achever leurs buts (coopération).

En plus, l'agent cognitif doit choisir les actions les plus satisfaisantes grâce à sa capacité de perception de son environnement et sa puissance de raisonnement, afin d'accomplir les traitements nécessaires pour prendre des bonnes décisions. Il peut fournir ses compétences nécessaires pour la coopération ou bien de demander des services des autres agents afin de réaliser des tâches spécifiques.

De ce fait, nos agents cognitifs ont pour but de partitionner horizontalement les données ERP qui font l'objet de l'extraction de connaissances sur des petites partitions de données moins volumineuses, l'intégration et l'affichage des résultats de l'extraction de règles d'association. Ainsi, le rôle de l'agent de règles d'association est l'exécution parallèle et distribuée de plusieurs instances de l'algorithme règles d'association sur ces données partitionnées avec efficacité et évolutivité. Ceci conduit à la réduction de temps global du traitement pour la génération des règles d'association sur l'énorme volume de données ERP.

Pour récapituler, notre architecture SMA est entièrement distribuée dans le sens où chaque agent communique avec n'importe quel agent de système sans passer par un intermédiaire. En effet, Le choix de cette architecture d'agent se fait principalement en fonction d'un besoin réel pour améliorer le temps d'exécution consommé par l'algorithme de règles d'association, lors du traitement de l'énorme quantité de données ERP métiers. L'architecture proposée est adaptée au contexte et aux spécificités de notre système d'extraction des règles d'association métiers qui permet :

(1) Un partitionnement adaptatif en utilisant un découpage horizontal de données ERP par métier,

(2) Une efficacité par le lancement parallèle distribué de plusieurs instances de l'algorithme de règles d'association, et

(3) Une fiabilité par la présentation des résultats significatifs et clairs, en forme de règles d'association par métier vis-à-vis l'utilisateur data minier.

Dans ce qui suit, nous présentons l'architecture fonctionnelle de chaque agent cognitif utilisé dans notre architecture SMA.

V.4. Architecture fonctionnelle des agents du système

V.4.1. Agent interface DM

L'agent interface DM est le seul agent qui interagit avec l'utilisateur Data Miner et le système DM en même temps. Il fournit une interface graphique pour interagir avec le système DM, qui permet à l'utilisateur Data Miner de formuler ses requêtes Data Mining, de saisir le minimum support et la confiance minimale ainsi que le chemin de données qui font l'objet de l'extraction de connaissances. Ensuite cet agent transmet les paramètres saisis à l'agent partitionnement de données pour lancer l'opération de découpage de données ERP par métier. Enfin, cet agent reçoit les résultats de l'extraction des règles d'association métiers pour les présenter à l'utilisateur Data Miner pour l'aide à décision.

A. Architecture interne de l'agent interface DM

Dans la Figure V-3, l'architecture interne de l'agent interface DM est comme suit :

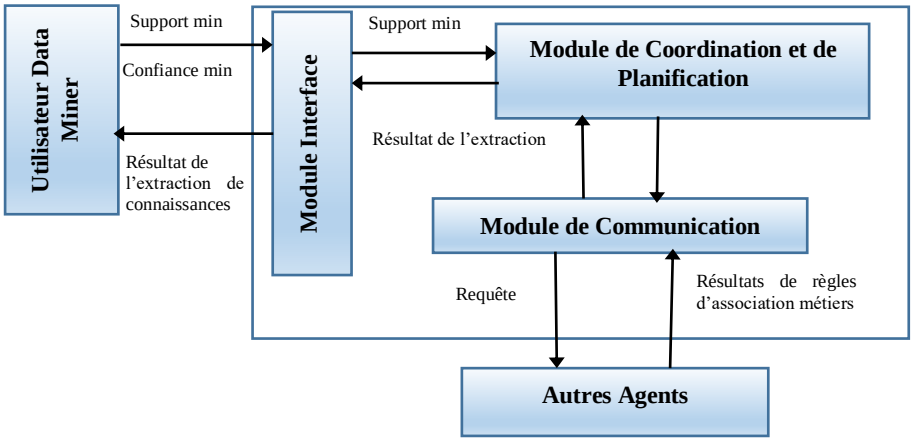


Figure V-3 : Architecture interne de l'agent interface DM

L'agent interface DM comporte trois modules différents, qui assurent son fonctionnement. Ces modules sont :

i. ***Module d'interface*** : il est responsable de l'interaction entre l'agent humain Data Mineur et le système. Il collecte les données d'entrées nécessaires au déclenchement du système. Il assure une interface d'assistance pour l'agent Data Miner, que ce soit pour la formulation de requête de Data Mining, ou bien pour la réception des résultats sous format des règles d'association métier.

ii. ***Module de Coordination et de Planification*** : Ce module prend en charge l'ensemble du processus global de la coordination du système. Il prend en considération un ensemble de paramètres pour répondre à son but (les résultats d'interprétation des

messages sous format des règles d'association métiers et le module de communication) et produit en sortie, un plan qui prend en charges ses buts. Ce module se compose d'un sous module de coordination et un sous module de planification pour l'orientation et l'ordonnancement des tâches internes.

iii. ***Module de communication*** : il gère l'échange d'information de communication avec les autres agents à travers un langage de communication FIPA-ACL. Il contient un processus de prise en charge des messages, à savoir : la réception et le filtrage des messages entrants ainsi que la formulation et l'envoi des messages sortants.

B. Fonctionnement de l'agent interface utilisateur

Cet agent passe par les étapes suivantes pour atteindre ses buts :

1. La capture du minimum support et la confiance minimale introduits par l'utilisateur Data Miner, nécessaire au lancement de l'opération de Data Mining.
2. L'Introduction du chemin de données font l'objet de Data Mining saisi par l'utilisateur Data Miner.
3. La création et l'initialisation de l'agent partitionnement de données.
4. L'envoie des paramètres introduits par l'utilisateur (minimum support, la confiance minimale et le chemin de données DM) à l'agent partitionnement de données pour commencer son travail.
5. La réception des messages envoyés par les agents de règles d'association, qui représentent les résultats des règles d'association métier pour l'interprétation et la présentation de connaissances vis-à-vis de l'utilisateur Data Miner.

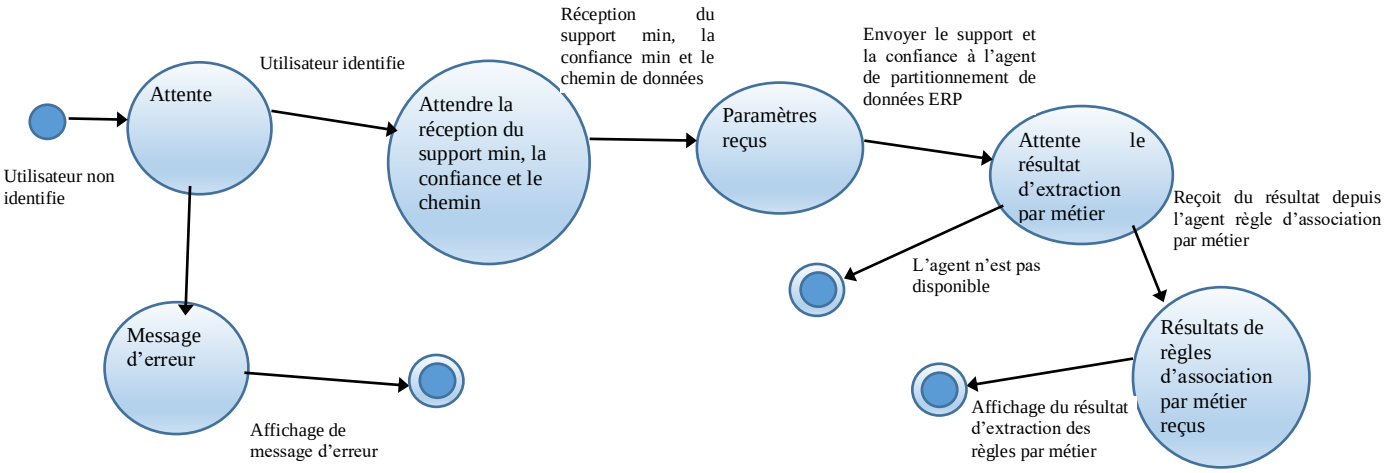


Figure V-4 : Diagramme d'état de l'agent Interface Data Mining

Entrée : *Support min, confiance min, chemin de la BDOF ;*
Sortie : *Résultat de l'extraction de règles d'association par métier ;*
Début
Wait all;*//* Attend la réception du support minimum et la confiance minimale provenant de l'utilisateur Data Miner ainsi que le chemin de la BDOF; */*
*/*Dans la première exécution de l'agent interface DM*/*
Créer *l'agent partitionnement de données ;*
Compte *; /*Utiliser pour compter le nombre des résultats de règles d'association par métier provenant des agents de règles d'association*/*
*/*Num_Partition : C'est le nombre de partitions métiers */*
*/*Si l'agent règle d'association termine son travail, il envoie le résultat de l'extraction par métier à l'agent interface DM ; ce dernier va compter le nombre des résultats reçus.*
Si *Compte = Num_Partitions_métiers* ***Alors*** *il afficher les résultats d'extraction des règles d'association par métier ;*
    Si *Compte == Num_Partition* ***Alors***
    Afficher les résultats d'extraction des règles d'association par métier ;
    Fin SI

Fin

Figure V-5 : Pseudo code de l'agent Interface Data Mining

V.4.2. Agent partitionnement de données ERP

C'est un agent cognitif dont le rôle est le partitionnement de données ERP qui font l'objet de l'opération de l'extraction des règles d'association. Il décompose horizontalement l'énorme masse de données ERP par métier sur plusieurs partitions de données moins volumineuses. Après, il lance en parallèle, plusieurs instances de l'agent de règles d'association par métier afin d'accélérer le processus d'extraction des règles d'association.

A. Architecture interne de l'agent de partitionnement de données ERP

L'architecture interne de l'agent de partitionnement de données ERP présentée dans la Figure V-5, elle comporte trois modules distincts et une Base de Données ERP Orientée Fouille (BDOF-ERP).

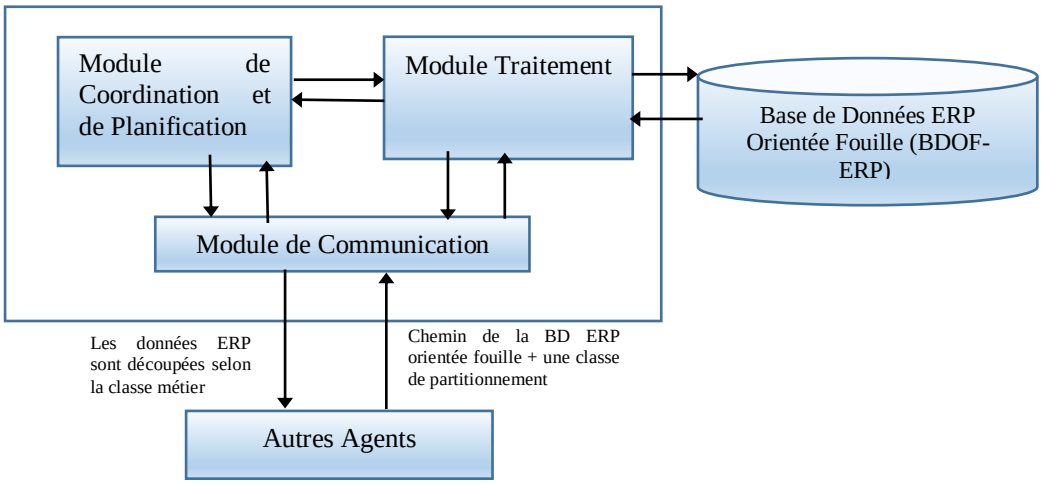


Figure V-6 : Architecture de l'agent partitionnement Data ERP

i. ***Module de communication :*** Ce module gère l'interaction avec l'agent interface utilisateur DM et l'agent de règles d'association. Il est responsable de toutes les fonctionnalités d'expédition et de réception des messages par l'intermédiaire d'un langage de communication entre les agents**.**

ii. ***Module de Coordination et de Planification*** : Ce module prend en charge l'ensemble du processus global de la coordination du système. Il prend en considération un ensemble de paramètres d'entrées afin de satisfaire son but de partitionnement de données ERP du module de traitement et le module de communication. Il coordonne les tâches effectuées par l'agent partitionnement de données ERP sur un plan d'action pour accomplir ses buts.

iii. ***Module de traitement*** : il contient toutes les méthodes et les procédures pour le partitionnement horizontal de données ERP sur des petites partitions de données par métier. Ces partitions sont créées d'une manière dynamique.

iv. ***Base de Données ERP Orientée Fouille (BDOF)*** : Contient les données de l'ERP qui font l'objet de l'extraction de connaissances. Ces données sont déjà sélectionnées, nettoyées et transformées par l'intervention de l'utilisateur Data Miner.

B. Fonctionnement de l'agent de partitionnement de données ERP

Cet agent suit les étapes suivantes pour accomplir ses tâches :

1. Réception du message envoyé par l'agent interface DM qui contient le chemin de données ERP.
2. Démarrage de l'opération de partitionnement de données sur plusieurs partitions moins volumineuses par métier.
3. Création de plusieurs agents de règles d'association, en parallèle, selon le nombre des partitions de données métiers.
4. Notification de l'agent interface DM pour que l'opération du partitionnement de données ERP soit terminée.
5. Lancement des agents de règles d'association afin de compléter leurs tâches.

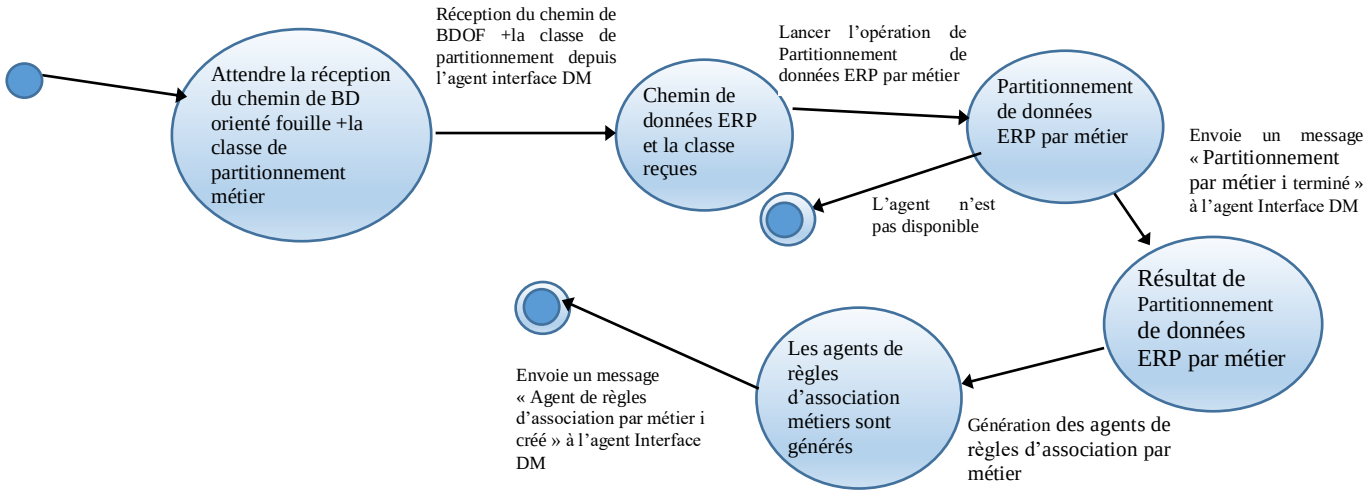


Figure V-7 : Diagramme d'état de l'agent de partitionnement de données ERP

```
Entrée : Base de Données Orientée Fouille (BDOF);
Sortie : Plusieurs partitions de données partitionnées par métier ;
Début
Wait all ; /*Attend la réception du chemin de la BDOF proviens de l'agent interface DM. */
 i= 1 ; // i Utiliser pour parcourir la Partition métier choisie (1 >= i <= Num_Partition)
/*Num_Partition: C'est le nombre de partitions métier*/
/* X : c'est un linge de la base de données BDOF
Pour chaque X ϵ BDOF faire
        Pour chaque i <= Num_Partition faire
                Si (Xi ϵ Partition_i) Alors
                                        Affecter Xi dans la partition métier_i ;
                Fin si
        Fin pour
Fin pour
Envoyer la fin de l'opération de partitionnement de données à l'agent interface DM;
i= 1;
Pour chaque i <= Num_Partition faire
Créer un agent de règle d'association i;
Envoyer la partition_i à l'agent de règle d'association i;
Envoyer un message « l'agent de règle d'association i » est créé et initialisé ;
Fin pour
Fin
```

Figure V-8 : Pseudo code de l'agent de partitionnement de données ERP

V.4.3. Agents de règles d'association

Plusieurs instances de l'agent de règles d'association sont créées d'une manière parallèle et dynamique par l'agent de partitionnement de données selon le nombre de partitions de données métiers. Son rôle est l'exécution de l'algorithme d'extraction de règles d'association après la réception du support minimum et la confiance minimale. Aussi, un autre rôle de cet agent est l'envoi des résultats de l'extraction de règles d'association par métier à l'agent interface utilisateur DM pour l'affichage finale. L'architecture interne de cet agent est illustrée dans la figure V 6.

A. Architecture interne de l'agent de règles d'association

L'architecture interne de cet agent comporte trois modules assurant son fonctionnement. Ces modules sont présentés comme suit :

i. ***Module de communication :*** ce module gère l'échange de messages avec l'agent interface DM et l'agent partitionnement de données via un langage de communication entre les agents. Il contient un processus de prise en charge des messages, à savoir : la réception et le filtrage des messages entrants ainsi que la formulation et l'envoi des messages sortants.

ii. ***Module de traitement :*** Il contient toutes les méthodes et les procédures pour créer et lancer l'algorithme de règles d'association ainsi que leur paramètres nécessaires telles que : le minimum support, la confiance minimale, le nombre des règles générés…etc.

iii. ***Module de Coordination :*** Il coordonne son but local avec le but global du système en se basant sur un ensemble de paramètres telles que : l'exécution d'algorithme de règles d'association sur chaque partition de données métiers et le module de communication.

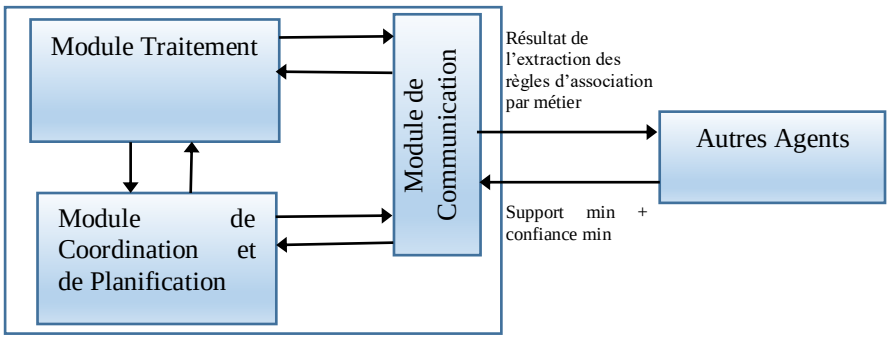


Figure V-9 : Architecture interne de l'agent de règles d'association

B. Fonctionnement d'agent de règles d'association

Les étapes passées par l'agent de règles d'association pendant son fonctionnement sont les suivantes :

1. Perception des messages envoyées de l'agent interface DM tels que : le minimum support et la confiance minimale, nécessaire au démarrage de l'algorithme de règles d'association.
2. Réception de sous-ensembles de données métiers parvient de l'agent partitionnement de données ERP.
3. Exécution parallèle et distribuée de plusieurs instances de l'algorithme de règles d'association sur les partitions de données métiers.
4. Envoi des résultats de l'extraction de règles d'association par métier à l'agent interface DM pour la présentation finale.

La figure V-10 montre le diagramme d'état de l'agent de règles d'association :

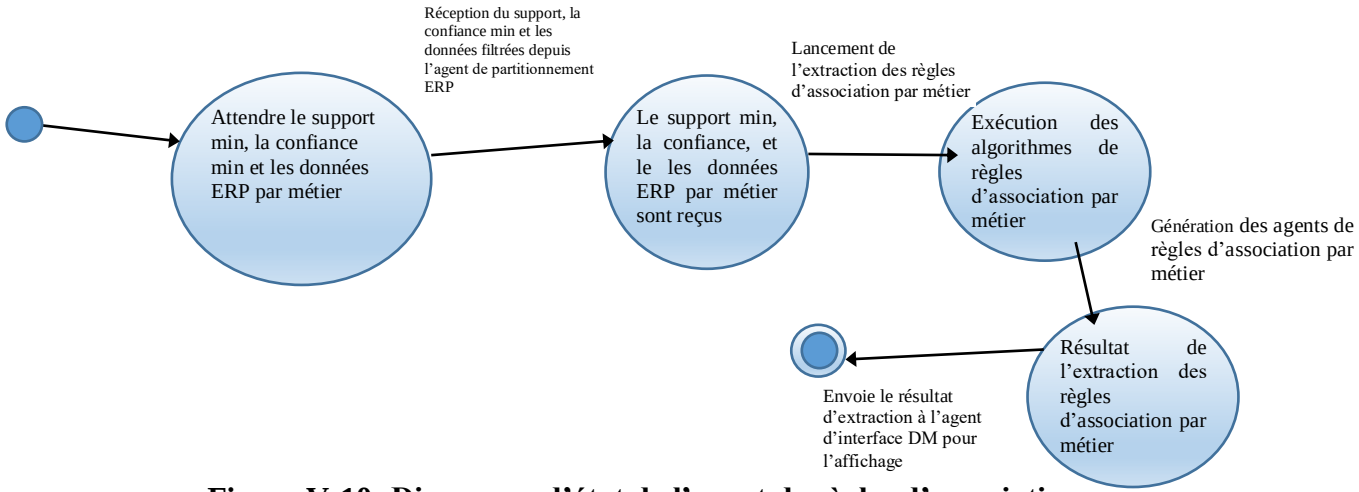


Figure V-10: Digramme d'état de l'agent de règles d'association

***Entrée** : Support min, confiance min, Données partitionnées par métier ;*
***Sortie** : Règles d'association par métier ;*
Début
***Wait all** ; //Attend la réception du support minimum et la confiance minimale provenant de l'agent interface DM ainsi que la réception de données partitionnées par métier provenant de l'agent de partitionnement de données.*
i= 1 ; // i Utiliser pour parcourir la classe choisi (1 >= i <= Num_Partition)
*/*Num_Partition : C'est le nombre de partitions métier*/*
Réception du Support min et confiance min provenant de l'agent interface DM ;
Réception de Partition i de données partitionnées par métier provenant de l'agent de partitionnement de données ;
Lancement de l'agent de règles d'association i ;
Envoi de règles d'association par métier à l'agent interface DM pour l'affichage de résultats ;
Notification de fin de l'opération d'extraction de règles d'association i à l'agent interface DM ;

FIN

Figure V-11 : Pseudo code de l'agent d'extraction de règles d'association

V.5. Communications des agents dans l'approche «AADARB-ERP »

La réussite de notre approche «AADARB-ERP » nécessite également une communication efficace entre les différents agents tout au long de son déroulement. Elle permet de bénéficier des services et du savoir-faire des autres agents, ce qui permet d'augmenter leurs capacités perceptives. La communication des agents constitue l'un des moyens principaux assurant la répartition des tâches et la coordination des actions [113] et [198].

Comme nous avons vu dans le chapitre III (Section III.3.5.) deux principaux modes de communication sont possibles : par envoi de messages ou bien par l'utilisation d'un blackboard. Le mode de communication que nous avons adopté pour notre approche est celui par envoi de messages. En effet, les agents utilisés dans « AADARB-ERP » sont de type cognitif où l'agent émetteur du message doit connaître avec précision l'adresse de l'agent destinataire. Par conséquent, le choix de communication par l'envoi de message est primordial pour faciliter la communication et l'interopérabilité entre nos agents durant les tâches d'extraction des règles d'association métiers.

Dans ce contexte, plusieurs langages de communication ont été développés pour faire interopérer les agents. Dans notre proposition, nous avons utilisé le langage FIPA-ACL pour formuler les messages échangés entre les différents agents de « AADARB-ERP ». En outre, ce langage supporte le concept de conversation pour indiquer une instance particulière d'un protocole, ce qui permet d'éviter de produire des comportements indésirables pendant l'extraction de règles d'association métiers. Pour pouvoir permettre aux agents de «AADARB-ERP » de distribuer la charge du problème d'extraction des règles d'association métiers d'une manière parallèle, la conversation elle-même doit être identifiée explicitement dans les messages, de façon à éliminer les ambiguïtés, en rattachant les messages aux contextes qui les concernent. Le langage FIPA-ACL [119] défini cinq paramètres liés à la gestion de la conversation tels que : protocol, conversation-id, reply-with, in-reply-to, et reply-by. En pratique, reply-with et in-reply-to sont peu utilisés, au profit de conversation-id qui identifie globalement la conversation.

V.6. Scénarios de fonctionnement de l'architecture SMA proposée

Les agents utilisés dans notre architecture SMA sont les suivants : Agent interface DM, Agent de partitionnement de données, Agent d'extraction de règles d'association. Tous les agents ont été créés dans l'environnement de développement JADE et enregistrés auprès de l'agent d'enregistrement (RA). Le processus DARM (Distributed Associate Rule Mining) est démarré par l'agent interface DM qui fournit l'interface entre le système et l'utilisateur Data Miner pour répondre aux demandes de ce dernier. Cet agent va réceptionner les trois paramètres nécessaires au démarrage de processus DARM qui sont le minimum support, la confiance min et le chemin où se trouve les données ERP pour les envoyer à l'agent partitionnement de données ERP. Ce dernier à son tour divise les données ERP horizontalement par métier sur plusieurs partitions moins volumineuses. Après, il lance un ensemble d'agents d'extraction de règles d'association parallèlement sur les partitions de données ERP, en fonction du nombre de métiers. Chaque agent d'extraction des règles d'association est chargé de l'exécution de la tâche d'extraction des règles d'association selon le métier qui lui est attribué. Une fois l'une des opérations d'extraction des règles d'association par métier est achevée par l'un des agents d'extraction de règles d'association, le résultat est envoyé directement à l'agent interface DM pour l'affichage et la présentation finale. Cette description est illustrée dans la Figure V-13.

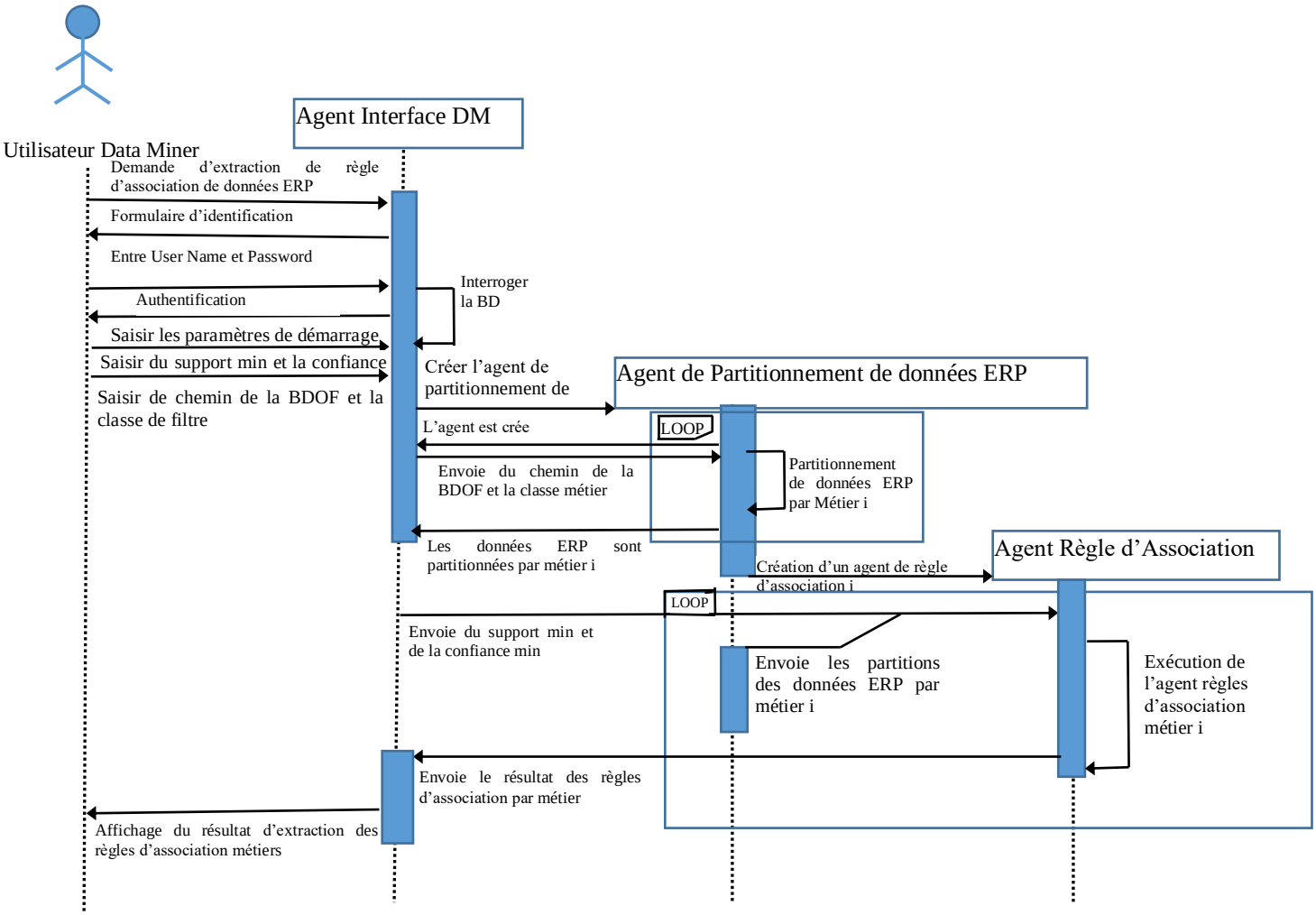


Figure V-12 : Diagramme de séquence AUML

La plupart des systèmes de DARM à base d'agents sont caractérisés par un nombre élevé de messages échangés au cours de l'extraction des règles d'association qui influence sur l'efficacité de ces systèmes. Il est donc important de minimiser le nombre et la taille des messages échangés dans notre travail. Dans les travaux précédents déjà décrits dans la section III.5.2., l'échange des informations entre les agents nécessite beaucoup d'envoi de messages entre eux afin d'extraire et fusionner les règles d'association. Dans notre architecture, les échanges de messages et de données sont optimisés, puisque les agents de notre système n'ont pas besoin d'échanger des données qu'avant l'étape de lancement d'agents de l'extraction de règles d'association et avant l'étape d'intégration des résultats. En plus, la phase d'agrégation des règles d'association n'existe plus puisque les règles produites sont affichées par métier. L'agent interface DM n'a pas à attendre que toutes les opérations d'extraction des règles métiers se terminent sur tous les sites afin d'afficher les résultats de l'extraction de règles d'association par métier, contrairement aux travaux précédents où les agents devaient attendre avant de pouvoir lire leurs messages et exécuter leurs tâches.

V.7. Analyse comparative sur les approches relatives à DARM à base d'agents

Une comparaison qualitative de notre approche proposée « AADARB-ERP » est fournie dans le Tableau V-1, par rapport aux autres travaux de DARM basés agents qui sont déjà présentés dans la section III.5.2. Elle est basée sur un certain nombre de critères afin d'évaluer ladite approche qui prend en considération l'extraction des règles d'association métiers dans l'environnement de l'ERP. Ci-après les critères de comparaison utilisés et, ensuite, la table comparative :

1. Itinéraire d'exécution des agents (en série ou en parallèle) ?
2. Utilisation d'un environnement de développement des agents ?
3. Implémentation du système (réelle ou juste un prototype) ?
4. Algorithme de règles d'association utilisé dans l'étude ?
5. Utilisation d'un modèle de coût pour mesurer et prévoir le temps de réponse dans le système ?
6. Utilisation d'une interface graphique pour l'utilisateur (GUI) dans le système.
7. Type de données utilisées dans la validation expérimentale (synthétique ou réelle) ?
8. Nature de sources de données (Centralisée ou Distribuée) ?
9. Cadre d'utilisation du système (Application pratique, projet de développement, Etudes de cas) ?
10. Existence d'un environnement d'exécution des sites distribués.
11. Existence d'un site central pour le lancement des agents.

Tableau V-1 : Tableau comparative des approches des règles d'association

	MADARM [167]-(2011)	**ABDDPARM** [155]-(2014)	**PEMA** [166]-(2015)	**AeMGSAR** [171]-(2015)	**Notre Approche «AADARB-ERP »** [220]-(2018)
Itinéraire d'exécution des agents	Série et Parallèle	Parallèle	Série et Parallèle	Série et Parallèle	Parallèle
Environnement de développement des agents	NON	OUI	OUI	OUI	OUI
Implémentation du système	Prototype	Réelle	Réelle	Réelle	Réelle
Algorithme ARM utilisé	Apriori et FP-growth	Apriori-T	PEMA	Apriori	Apriori
Utilisation de modèle de coût	Oui	NON	NON	OUI	OUI
Utilisation de l'interface GUI	NON	NON	NON	OUI	OUI
Données de validation expérimentale	NON	Ensemble de données Synthétique	Ensemble de données Réelle	Ensemble de données Synthétique	**Base de données Réelle de l'ERP**
Nature de sources de Données	Distribuée	Distribuée	Distribuée	Distribuée	**Centralisée**
Cadre de l'utilisation du système	NON	NON	NON	Une étude de cas en bio-Informatique	**Une étude de cas sur la prestation de services aux puits dans l'ENSP**
Environnement d'exécution des sites distribués	NON	OUI	Oui, Virtual	OUI	Oui, Virtual
Site central de lancement des agents	NON	OUI	OUI	OUI	OUI

Cette analyse comparative montre que les approches MADARM [67]-(2011), PEMA [66]-(2015 et AeMGSAR[71]-(2015) sont basées sur des Itinéraires parallèles et en séries, tandis que ABDDPARM [55]-(2014 et notre approche « AADARB-ERP » sont fondées sur un itinéraire d'exécution parallèle. L'approche MADARM [67]-(2011) n'est qu'un prototype, sans environnement de développement et sans implémentation. Dans la plupart des approches présentées, l'algorithme Apriori est principalement utilisé pour l'extraction des règles d'association à l'exception de l'approche PEMA [66]-(2015) qui développe un algorithme spécifique pour l'extraction des règles d'association. Les MADARM [67]-(2011), AeMGSAR[71]-(2015) et notre approche « AADARB-ERP » utilisent un modèle de coût pour mesurer le temps de réponse lors de l'exécution des tâches DARM. Par contre, les deux

autres approches ne l'ont pas. Notre approche « AADARB-ERP » et l'approche de AeMGSAR[71]-(2015) ont été expérimentées sur des études de cas réelles tandis que les autres ne les disposent pas. L'approche « AADARB-ERP » et PEMA [66]-(2015) utilisent des bases de données réelles pour la validation du système développé, où ABDDPARM [55]-(2014) et AeMGSAR [71]-(2015) appliquent leurs expérimentations sur un ensemble de données synthétique. Seulement notre approche « AADARB-ERP » et l'approche de AeMGSAR[71]-(2015) utilisent une interface utilisateur graphique dont leurs plateformes de DARM sont basées agents. La plupart des approches présentées disposent d'un site central de lancement des agents et un environnement pour l'exécution des sites distribués sauf l'approche MADARM [67]-(2011). Tous les travaux examinent des sources de données de nature distribuée alors que notre approche « AADARB-ERP » utilise une source de données de nature centralisée, puisque cela est relative à l'architecture de l'ERP, qui est fondée principalement sur une base de données centralisée. En plus, notre approche est distinguée par rapport aux autres approches présentées, en ce qui concerne l'extraction des règles d'association métiers. Celle-ci permet de trouver des associations cachées et intéressantes entre les items de services aux puits à partir d'une grande base de données ERP de l'entreprise ENSP.

Cette analyse comparative nous permet d'apprécier notre approche proposée « AADARB-ERP » [220] pour l'extraction des règles d'association métiers à partir de la base de données du système ERP. Elle est basée particulièrement sur le partitionnement horizontal de données ERP et les agents coopératifs. L'utilisation du système multi-agents est primordiale dans cette approche, puisqu'il permet de créer des modèles de connaissances plus clairs et plus robustes par l'intégration des agents coopératifs au sein du processus de l'extraction des règles d'association, tout en se penchant sur la distribution, la collaboration, la flexibilité, la réutilisation, l'évolutivité et l'efficacité des systèmes multi-agents. De même, notre approche proposée est fondée principalement sur une base de données ERP de type centralisé, mais aussi elle s'appuie sur une architecture totalement distribuée. D'une manière parallèle et distribuée, elle est également capable d'exécuter de multiples processus d'extraction des règles d'association par métier à partir d'une grande masse de données ERP. Cela se fait à l'aide de la technologie d'agent et d'un partitionnement horizontal de données métiers. Ce qui est attendu, c'est que notre approche réduira le temps de traitement global quant à l'extraction des règles d'association métiers ainsi qu'elle améliorera la qualité des règles métiers extraites pour aider les managers dans la prise de décisions stratégiques.

V.8. Implémentation et expérimentation de l'approche proposée

Cette section est consacrée à la présentation des différents outils utilisés pour le développement de notre approche proposée «AADARB-ERP » ainsi que la description du domaine d'application. Ensuite, nous décrivons la phase de préparation des données ERP et l'implémentation des sites distribués. Nous présentons par la suite les interfaces du système développé et les résultats probants obtenus dans le cadre des expérimentations.

V.8.1. Outils d'implémentation

Plusieurs outils sont utilisés pour développer l'approche proposée. « Java Eclipse IDE for Java Developers » a été utilisé comme un environnement de développement. « JADE 3.8 » a été utilisé comme une plate-forme middleware de développement du système multi-agent. L'outil « WEKA » est une boite à outils open source d'apprentissage automatique écrite en java, a été utilisée comme un environnement de Data Mining. La machine virtuelle « VMware Workstation 12 Pro » a été utilisée pour simuler la distribution de données ERP dans un environnement distribué. La base de données Postgres de l'ERP Odoo, de l'entreprise **ENSP** (**E**ntreprise **N**ationale de **S**ervices aux **P**uits), a été utilisée comme une base d'expérimentation pour valider le système développé.

A. La plateforme JADE

Afin d'implémenter les agents de notre système, la plateforme JADE (Java Agent Development Framework) fondé sur le langage JAVA semble la plus adéquate car elle répond mieux à nos besoins pour le développement des agents cognitifs et fournit en même temps, un ensemble complet des services et d'agents conformes aux spécifications FIPA. La plateforme Jade est largement répandue pour le développement orienté agent, et elle a été utilisée avec succès dans les différents milieux.

JADE est un environnement de développement d'agents implémenté totalement en JAVA. Il facilite la mise en place d'un système multi-agent (SMA) et essaie d'optimiser les performances du système d'agents distribué, tout en répondant aux spécifications FIPA (Foundation for Intelligent Physical Agents) à travers un ensemble de composants [39]. Ces composants sont le langage de communication entre agents (ACL : Agent Communication Language), le système de gestion des agents (AMS : Agent Management System) et le facilitateur d'annuaire (DF). Ces trois agents sont automatiquement créés et activés lorsque la plate-forme Jade est activée [38]. Toute communication entre agents est effectuée par des messages FIPA-ACL.

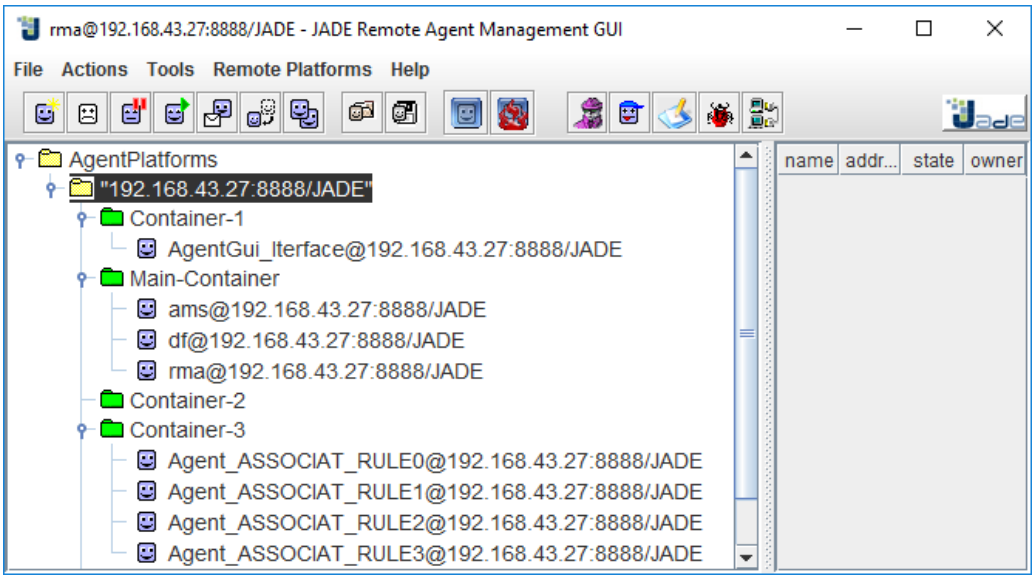


Figure V-13 : Agents de notre système développé sous Jade

B. L'outil WEKA

Dans notre travail, nous avons utilisé la plateforme WEKA comme un outil de prétraitement de données ERP et comme librairie des algorithmes de règles d'association afin de les intégrer avec les agents Jade développés.

WEKA (Waikato Environment for Knowledge Analysis) est une boite à outils open source d'extraction des connaissances, écrite en java, développée à l'université Waikato, Nouvelle-Zélande. Il est désormais utilisé dans beaucoup de domaines différents, en particulier la recherche et l'éducation. Les principaux points forts de WEKA sont : [37]

- librement disponible (en particulier gratuitement) sous la licence publique générale GNU,
- portable car elle est entièrement implémentée en Java et peut, en conséquence, fonctionner sur la quasi-totalité des plateformes modernes,
- contient une collection complète de préprocesseurs de données et de techniques de modélisation, et
- facile à utiliser par un novice en raison de l'interface graphique qu'il contient et facile aussi à intégrer dans les applications JAVA.

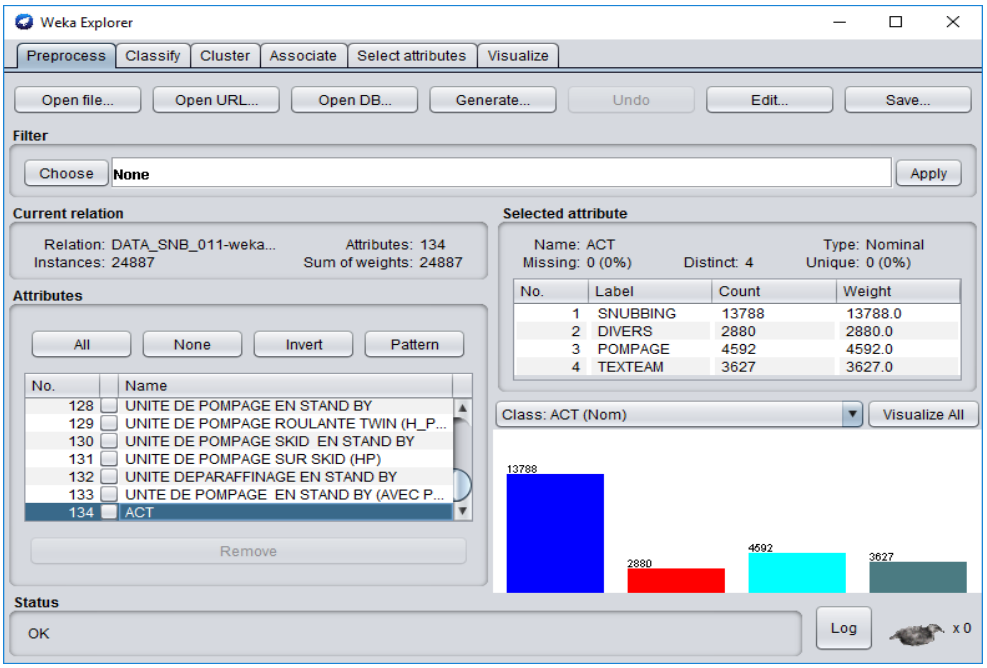


Figure V-14 : Services métiers de direction ENSP-Snubbing sous Weka 3.8

C. Interfaces de l'ERP Odoo et sa base de données Postgres

Dans cette section, nous présentons quelques interfaces du progiciel « ERP-Odoo » et sa base de données « Postgres » propres à l'entreprise ENSP. La base de données « Postgres » de l'ERP-Odoo est employée comme une source de données métiers pour expérimenter notre approche proposée.

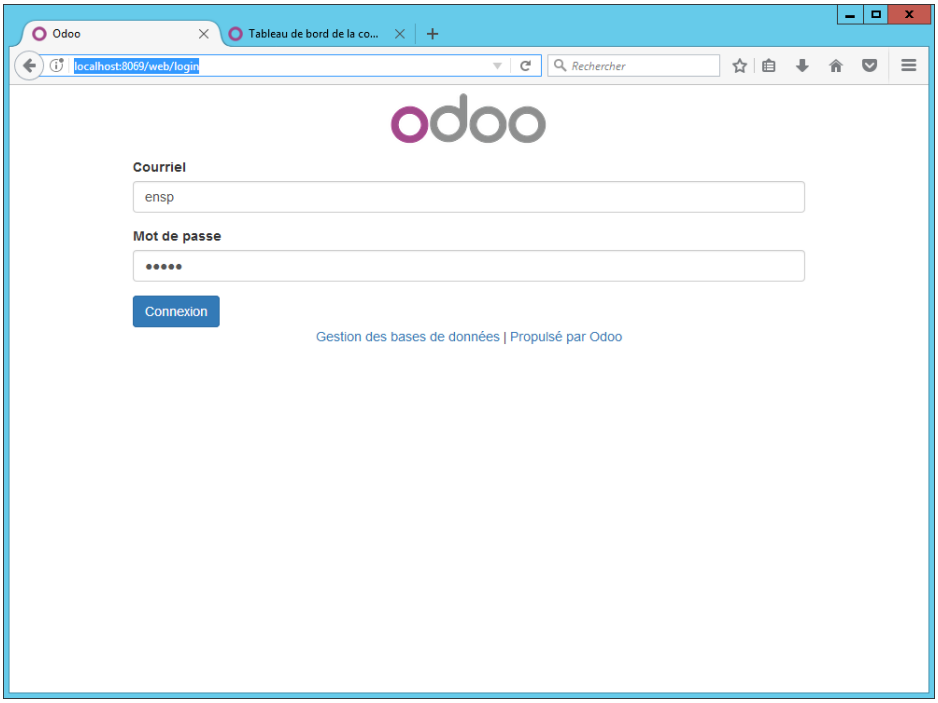


Figure V-15 : Authentification du progiciel ERP-Odoo de l'ENSP

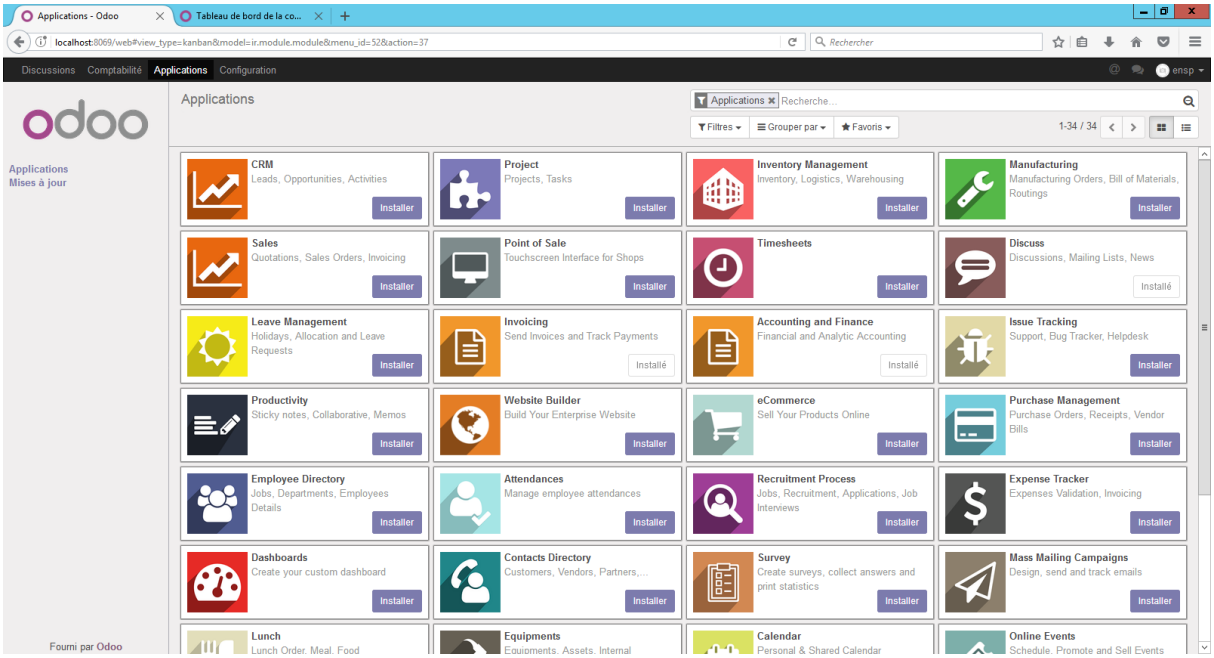


Figure V-16 : Interface principale de l'ERP Odoo de l'ENSP

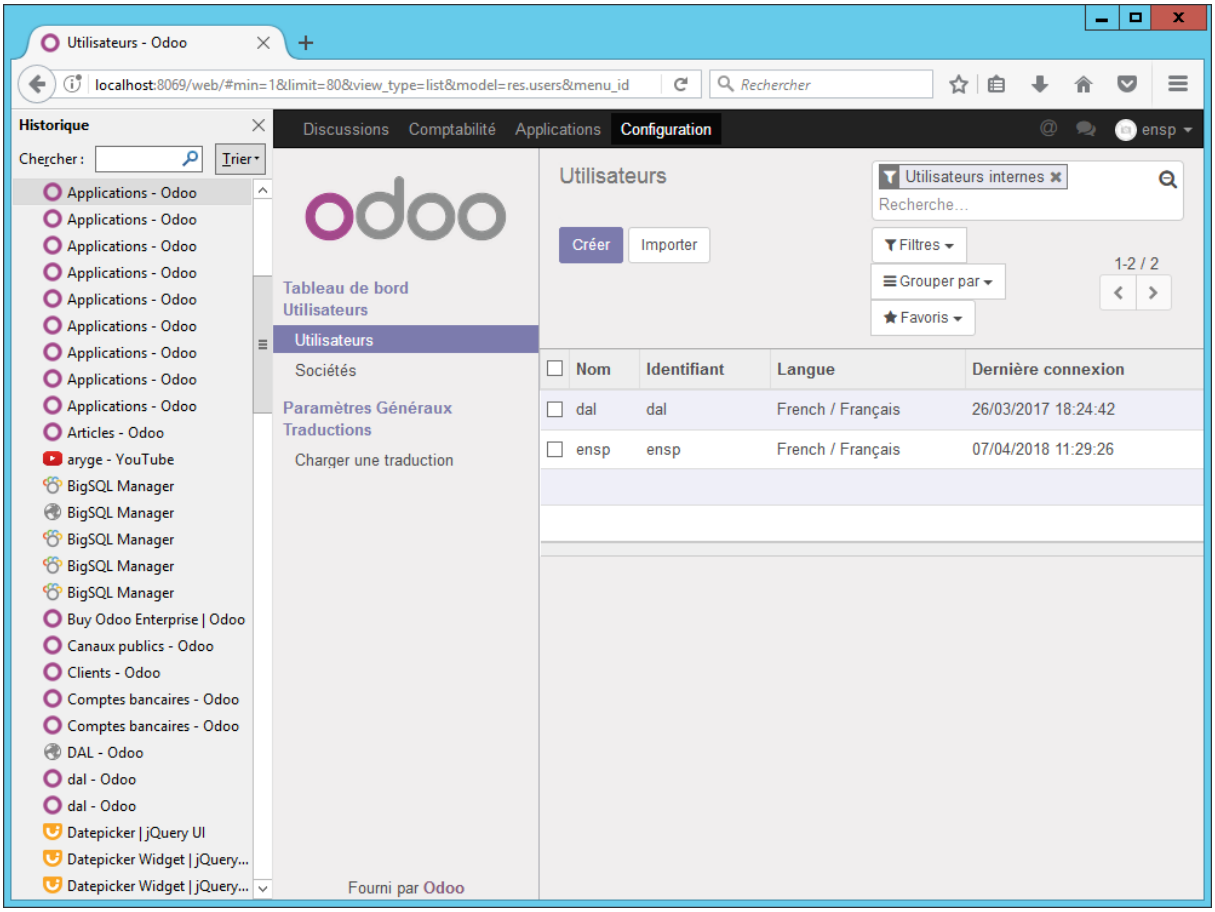


Figure V-17 : Utilisateurs du système ERP-Odoo

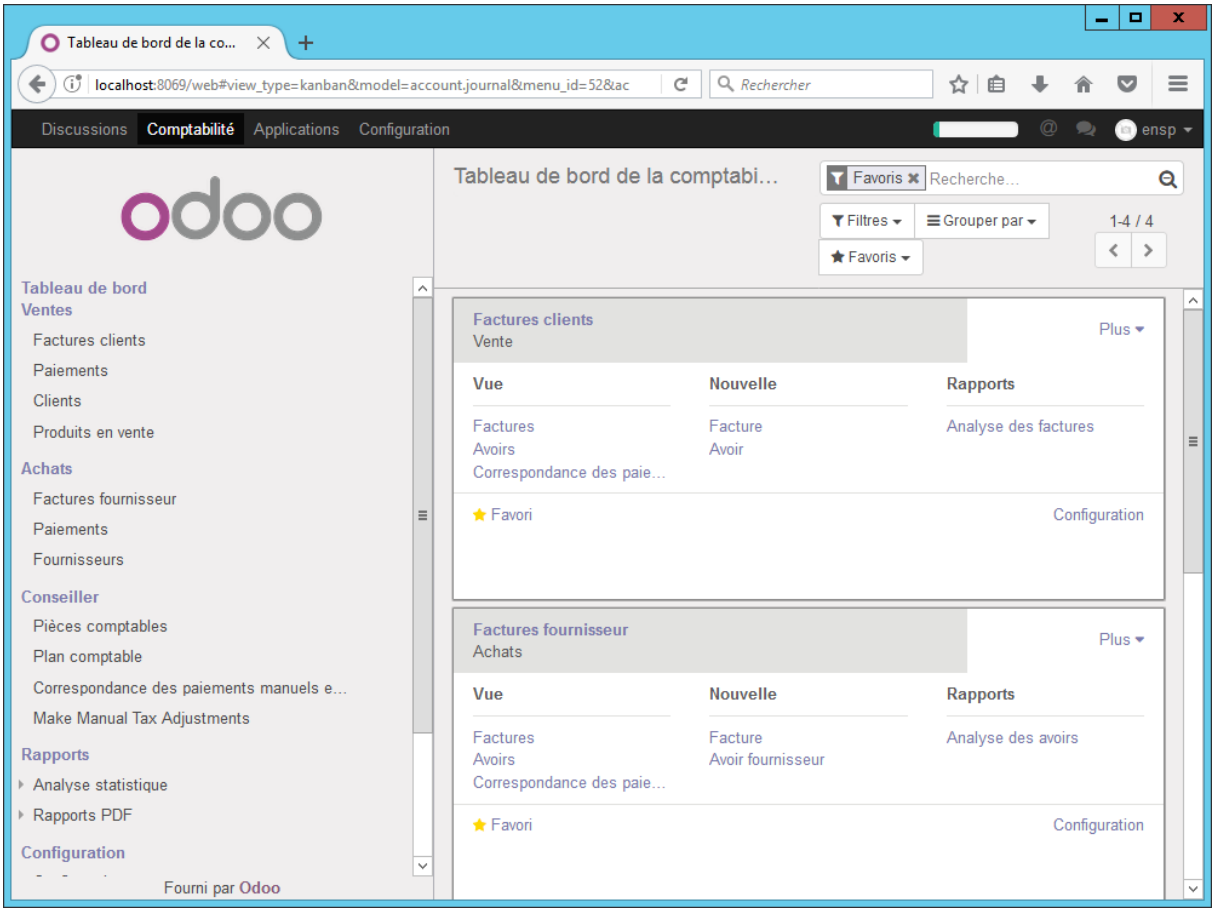


Figure V-18 : Module comptabilité et facturation de l'ERP Odoo

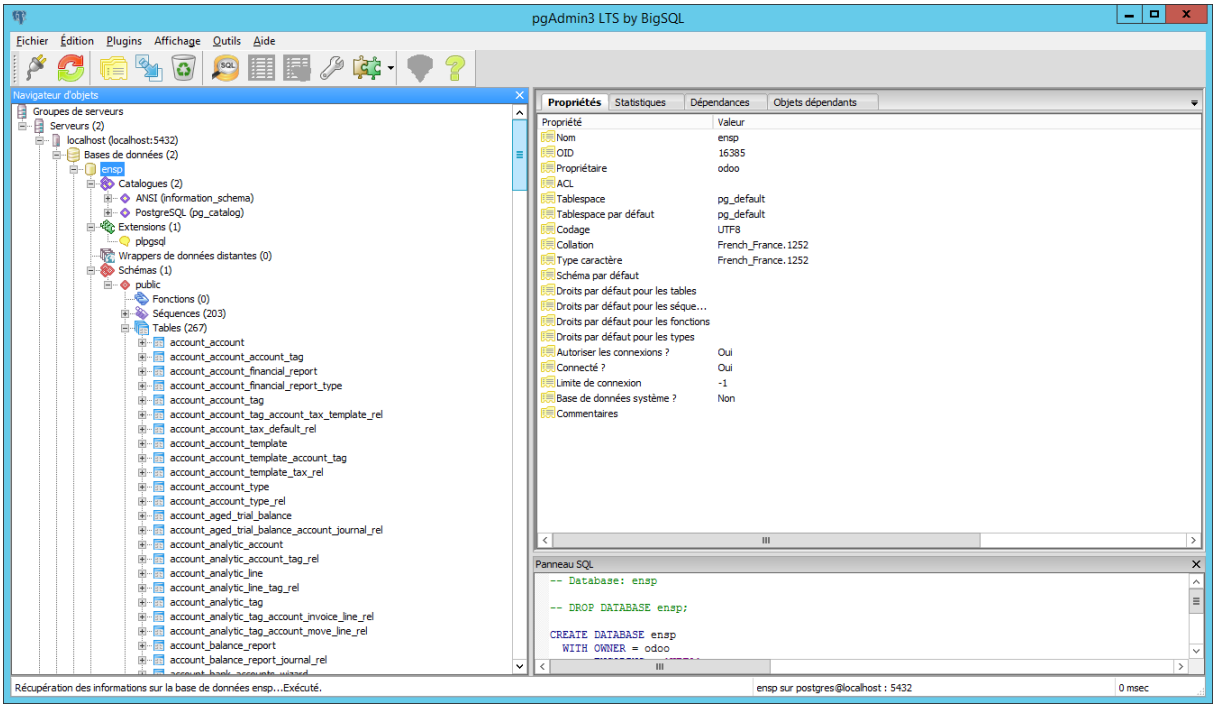


Figure V-19 : Base de données Postgred de l'ERP Odoo

V.8.2. Description du domaine d'application

Dans notre expérimentation, nous tenons comme étude de cas réel, la base de données ERP de services aux puits de l'entreprise nationale de services aux puits (ENSP). Cette entreprise, filiale de SONATRACH 100%, est un groupe représentant un important capital de savoir-faire et d'expérience accumulée depuis plus de 30 années dans les services techniques relatifs à l'exploration et l'exploitation pétrolière. Elle intervient pour tester et entretenir les puits de pétrole, ainsi que le snubbing, pompage, texteam …etc.

La base de données ERP accumule une quantité énorme de données de la société ENSP à cause des usages quotidiens et continus. D'autant plus qu'elle cache des connaissances décisionnelles sur le marché et la concurrence, mais elles restent peu exploitées. Pour répondre à ce besoin, nous avons utilisé la technique d'extraction des règles d'association pour chercher et analyser les métiers opérationnels à réaliser ensemble. Ces règles d'association permettent d'analyser le comportement des clients et de comprendre leurs habitudes en termes de prestation de services afin d'établir des décisions efficaces pour l'entreprise ENSP, telles que :

1. Préparation et organisation des équipements nécessaires à la réalisation d'un service aux puits ;
2. Connaissance préalable des services métiers qui sont fournis ensemble ;
3. Sélection et optimisation du personnel nécessaire à la réalisation des services métiers (Pompage, Texteam, Snubbing …etc);
4. Optimisation de la gestion des stocks et l'agencement des rayons du magasin.
5. Prédiction de futurs besoins en prestations des clients par métier pour anticiper les actions nécessaires.

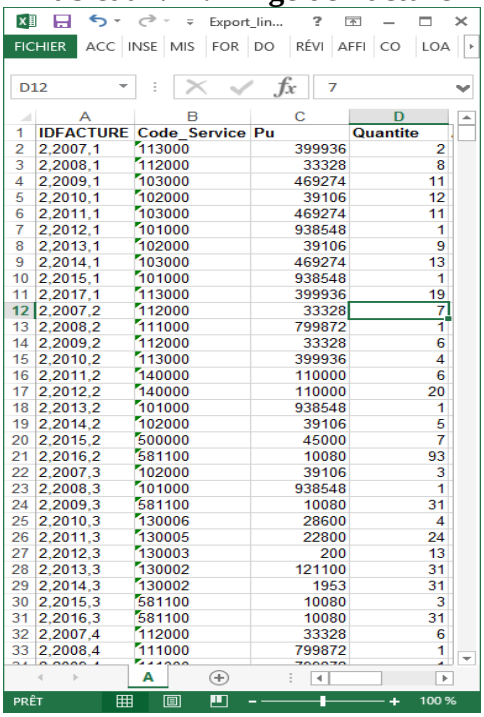
V.8.3. Préparation des données

Selon [36], la phase de la préparation de données peut prendre plus de temps que l'exploration de données et peut présenter des défis égaux avec la phase de l'exploration de données elle-même. Cette phase consiste à obtenir des données en accord avec les objectifs de Data Mining visés. Ces données proviennent généralement des bases de production et sont structurées en champs typés (dans un domaine d'application). L'obtention des données est souvent réalisée à l'aide d'outils d'extraction de données par requête (OLAP, SQL, etc.).

Les données utilisées dans notre expérimentation sont exportées depuis la base de données ERP-PostgreSQL relative aux prestations de services ENSP. Ces données représentent une importante masse des données en matière de services aux puits.

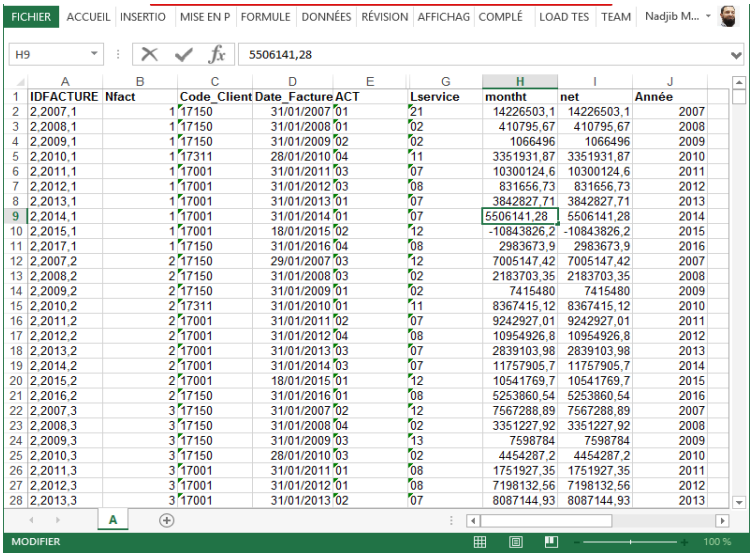
Dans cette phase, les données ERP qui font l'objet de Data Mining sont obtenues à partir d'une exportation Excel, d'un ensemble de tables en relation avec les services aux puits, et ce à partir de la base de données ERP Postgres de l'ENSP. Ces tables sont : Facture, Lignes de factures, Lignes de services et Métier (Voir les Tableaux V-1, V-2, V-3 et V-4).

Tableau V-2: Linge de Facture



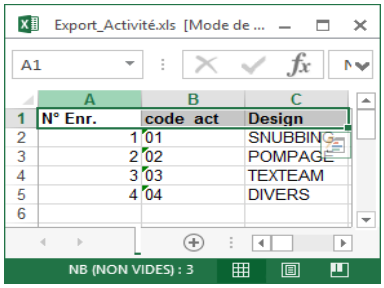
	A	B	C	D
1	IDFACTURE	Code_Service	Pu	Quantite
2	2,2007,1	113000	399936	2
3	2,2008,1	112000	33328	8
4	2,2009,1	103000	469274	11
5	2,2010,1	102000	39106	12
6	2,2011,1	103000	469274	11
7	2,2012,1	101000	938548	1
8	2,2013,1	102000	39106	9
9	2,2014,1	103000	469274	13
10	2,2015,1	101000	938548	1
11	2,2017,1	113000	399936	19
12	2,2007,2	112000	33328	7
13	2,2008,2	111000	799872	1
14	2,2009,2	112000	33328	6
15	2,2010,2	113000	399936	4
16	2,2011,2	140000	110000	6
17	2,2012,2	140000	110000	20
18	2,2013,2	101000	938548	1
19	2,2014,2	102000	39106	5
20	2,2015,2	500000	45000	7
21	2,2016,2	581100	10080	93
22	2,2007,3	102000	39106	3
23	2,2008,3	101000	938548	1
24	2,2009,3	581100	10080	31
25	2,2010,3	130006	28600	4
26	2,2011,3	130005	22800	24
27	2,2012,3	130003	200	13
28	2,2013,3	130002	121100	31
29	2,2014,3	130002	1953	31
30	2,2015,3	581100	10080	3
31	2,2016,3	581100	10080	31
32	2,2007,4	112000	33328	6
33	2,2008,4	111000	799872	1

Tableau V-3 : Facture



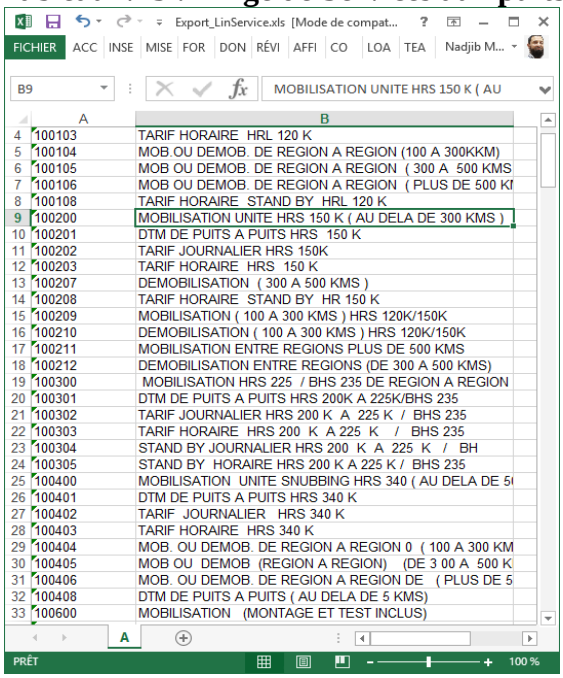
	A	B	C	D	E	G	H	I	J
1	IDFACTURE	Nfact	Code_Client	Date_Facture	ACT	Lservice	montht	net	Année
2	2,2007,1	1	17150	31/01/2007	01	21	14226503,1	14226503,1	2007
3	2,2008,1	1	17150	31/01/2008	01	02	410795,67	410795,67	2008
4	2,2009,1	1	17150	31/01/2009	02	02	1066496	1066496	2009
5	2,2010,1	1	17311	28/01/2010	04	11	3351931,87	3351931,87	2010
6	2,2011,1	1	17001	31/01/2011	03	07	10300124,6	10300124,6	2011
7	2,2012,1	1	17001	31/01/2012	03	08	831656,73	831656,73	2012
8	2,2013,1	1	17001	31/01/2013	01	07	3842827,71	3842827,71	2013
9	2,2014,1	1	17001	31/01/2014	01	07	5506141,28	5506141,28	2014
10	2,2015,1	1	17001	18/01/2015	02	12	-10843826,2	-10843826,2	2015
11	2,2017,1	1	17150	31/01/2016	04	08	2983673,9	2983673,9	2016
12	2,2007,2	2	17150	29/01/2007	03	12	7005147,42	7005147,42	2007
13	2,2008,2	2	17150	31/01/2008	03	02	2183703,35	2183703,35	2008
14	2,2009,2	2	17150	31/01/2009	01	02	7415480	7415480	2009
15	2,2010,2	2	17311	31/01/2010	01	11	8367415,12	8367415,12	2010
16	2,2011,2	2	17001	31/01/2011	02	07	9242927,01	9242927,01	2011
17	2,2012,2	2	17001	31/01/2012	04	08	10954926,8	10954926,8	2012
18	2,2013,2	2	17001	31/01/2013	03	07	2839103,98	2839103,98	2013
19	2,2014,2	2	17001	31/01/2014	03	07	11757905,7	11757905,7	2014
20	2,2015,2	2	17001	18/01/2015	01	12	10541769,7	10541769,7	2015
21	2,2016,2	2	17150	31/01/2016	01	08	5253860,54	5253860,54	2016
22	2,2007,3	3	17150	31/01/2007	02	12	7567288,89	7567288,89	2007
23	2,2008,3	3	17150	31/01/2008	04	02	3351227,92	3351227,92	2008
24	2,2009,3	3	17150	31/01/2009	03	13	7598784	7598784	2009
25	2,2010,3	3	17150	28/01/2010	03	02	4454287,2	4454287,2	2010
26	2,2011,3	3	17001	31/01/2011	01	08	1751927,35	1751927,35	2011
27	2,2012,3	3	17001	31/01/2012	01	08	7198132,56	7198132,56	2012
28	2,2013,3	3	17001	31/01/2013	02	07	8087144,93	8087144,93	2013

Tableau V-4 : Métier



	A	B	C
1	N° Enr.	code_act	Design
2	1	01	SNUBBING
3	2	02	POMPAGE
4	3	03	TEXTEAM
5	4	04	DIVERS
6			

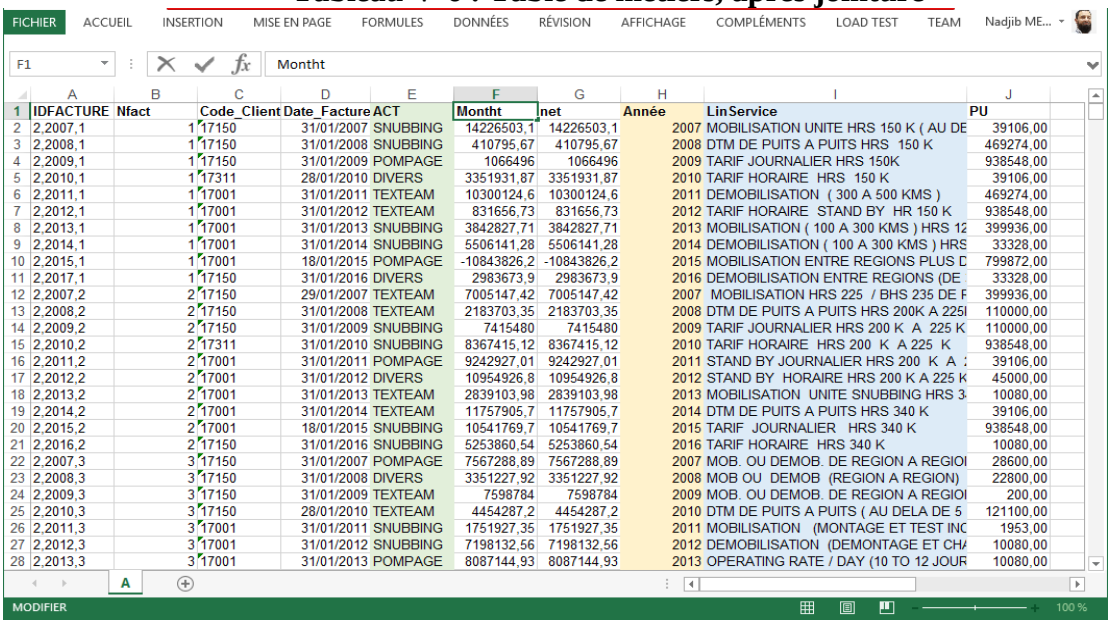
Tableau V-5 : Linge de Services aux puits



	A	B
4	100103	TARIF HORAIRE HRL 120 K
5	100104	MOB.OU DEMOB. DE REGION A REGION (100 A 300KKM)
6	100105	MOB OU DEMOB. DE REGION A REGION (300 A 500 KMS
7	100106	MOB OU DEMOB. DE REGION A REGION (PLUS DE 500 KI
8	100108	TARIF HORAIRE STAND BY HRL 120 K
9	100200	MOBILISATION UNITE HRS 150 K (AU DELA DE 300 KMS)
10	100201	DTM DE PUITS A PUITS HRS 150 K
11	100202	TARIF JOURNALIER HRS 150K
12	100203	TARIF HORAIRE HRS 150 K
13	100207	DEMOBILISATION (300 A 500 KMS)
14	100208	TARIF HORAIRE STAND BY HR 150 K
15	100209	MOBILISATION (100 A 300 KMS) HRS 120K/150K
16	100210	DEMOBILISATION (100 A 300 KMS) HRS 120K/150K
17	100211	MOBILISATION ENTRE REGIONS PLUS DE 500 KMS
18	100212	DEMOBILISATION ENTRE REGIONS (DE 300 A 500 KMS)
19	100300	MOBILISATION HRS 225 / BHS 235 DE REGION A REGION
20	100301	DTM DE PUITS A PUITS HRS 200K A 225K/BHS 235
21	100302	TARIF JOURNALIER HRS 200 K A 225 K / BHS 235
22	100303	TARIF HORAIRE HRS 200 K A 225 K / BHS 235
23	100304	STAND BY JOURNALIER HRS 200 K A 225 K / BH
24	100305	STAND BY HORAIRE HRS 200 K A 225 K / BHS 235
25	100400	MOBILISATION UNITE SNUBBING HRS 340 (AU DELA DE 5
26	100401	DTM DE PUITS A PUITS HRS 340 K
27	100402	TARIF JOURNALIER HRS 340 K
28	100403	TARIF HORAIRE HRS 340 K
29	100404	MOB. OU DEMOB. DE REGION A REGION 0 (100 A 300 KM
30	100405	MOB OU DEMOB (REGION A REGION) (DE 3 00 A 500 K
31	100406	MOB. OU DEMOB. DE REGION A REGION DE (PLUS DE 5
32	100408	DTM DE PUITS A PUITS (AU DELA DE 5 KMS)
33	100600	MOBILISATION (MONTAGE ET TEST INCLUS)

Après, nous procédons à la jointure de ces tables pour produire une table globale de métiers qui sera utilisée comme base pour notre expérimentation.

Tableau V-6 : Table de métiers, après jointure



	A	B	C	D	E	F	G	H	I	J
1	IDFACTURE	Nfact	Code_Client	Date_Facture	ACT	Montht	net	Année	LinService	PU
2	2,2007,1		1	17150	31/01/2007	SNUBBING	14226503,1	14226503,1	2007	MOBILISATION UNITE HRS 150 K (AU DE	39106,00
3	2,2008,1		1	17150	31/01/2008	SNUBBING	410795,67	410795,67	2008	DTM DE PUITS A PUITS HRS 150 K	469274,00
4	2,2009,1		1	17150	31/01/2009	POMPAGE	1066496	1066496	2009	TARIF JOURNALIER HRS 150K	938548,00
5	2,2010,1		1	17311	28/01/2010	DIVERS	3351931,87	3351931,87	2010	TARIF HORAIRE HRS 150 K	39106,00
6	2,2011,1		1	17001	31/01/2011	TEXTEAM	10300124,6	10300124,6	2011	DEMOBILISATION (300 A 500 KMS)	469274,00
7	2,2012,1		1	17001	31/01/2012	TEXTEAM	831656,73	831656,73	2012	TARIF HORAIRE STAND BY HR 150 K	938548,00
8	2,2013,1		1	17001	31/01/2013	SNUBBING	3842827,71	3842827,71	2013	MOBILISATION (100 A 300 KMS) HRS 12	399936,00
9	2,2014,1		1	17001	31/01/2014	SNUBBING	5506141,28	5506141,28	2014	DEMOBILISATION (100 A 300 KMS) HRS	33328,00
10	2,2015,1		1	17001	18/01/2015	POMPAGE	-10843826,2	-10843826,2	2015	MOBILISATION ENTRE REGIONS PLUS D	799872,00
11	2,2017,1		1	17150	31/01/2016	DIVERS	2983673,9	2983673,9	2016	DEMOBILISATION ENTRE REGIONS (DE	33328,00
12	2,2007,2		2	17150	29/01/2007	TEXTEAM	7005147,42	7005147,42	2007	MOBILISATION HRS 225 / BHS 235 DE F	399936,00
13	2,2008,2		2	17150	31/01/2008	TEXTEAM	2183703,35	2183703,35	2008	DTM DE PUITS A PUITS HRS 200K A 225I	110000,00
14	2,2009,2		2	17150	31/01/2009	SNUBBING	7415480	7415480	2009	TARIF JOURNALIER HRS 200 K A 225 K	110000,00
15	2,2010,2		2	17311	31/01/2010	SNUBBING	8367415,12	8367415,12	2010	TARIF HORAIRE HRS 200 K A 225 K	938548,00
16	2,2011,2		2	17001	31/01/2011	POMPAGE	9242927,01	9242927,01	2011	STAND BY JOURNALIER HRS 200 K A	39106,00
17	2,2012,2		2	17001	31/01/2012	DIVERS	10954926,8	10954926,8	2012	STAND BY HORAIRE HRS 200 K A 225 K	45000,00
18	2,2013,2		2	17001	31/01/2013	TEXTEAM	2839103,98	2839103,98	2013	MOBILISATION UNITE SNUBBING HRS 3	10080,00
19	2,2014,2		2	17001	31/01/2014	TEXTEAM	11757905,7	11757905,7	2014	DTM DE PUITS A PUITS HRS 340 K	39106,00
20	2,2015,2		2	17001	18/01/2015	SNUBBING	10541769,7	10541769,7	2015	TARIF JOURNALIER HRS 340 K	938548,00
21	2,2016,2		2	17150	31/01/2016	SNUBBING	5253860,54	5253860,54	2016	TARIF HORAIRE HRS 340 K	10080,00
22	2,2007,3		3	17150	31/01/2007	POMPAGE	7567288,89	7567288,89	2007	MOB. OU DEMOB. DE REGION A REGIOI	28600,00
23	2,2008,3		3	17150	31/01/2008	DIVERS	3351227,92	3351227,92	2008	MOB OU DEMOB (REGION A REGION)	22800,00
24	2,2009,3		3	17150	31/01/2009	TEXTEAM	7598784	7598784	2009	MOB. OU DEMOB. DE REGION A REGIOI	200,00
25	2,2010,3		3	17150	28/01/2010	TEXTEAM	4454287,2	4454287,2	2010	DTM DE PUITS A PUITS (AU DELA DE 5	121100,00
26	2,2011,3		3	17001	31/01/2011	SNUBBING	1751927,35	1751927,35	2011	MOBILISATION (MONTAGE ET TEST INC	1953,00
27	2,2012,3		3	17001	31/01/2012	SNUBBING	7198132,56	7198132,56	2012	DEMOBILISATION (DEMONTAGE ET CHA	10080,00
28	2,2013,3		3	17001	31/01/2013	POMPAGE	8087144,93	8087144,93	2013	OPERATING RATE / DAY (10 TO 12 JOUR	10080,00

Les données de la table de métiers peuvent être incomplètes, bruyantes et incohérentes, ce qui peut influencer sur la qualité et l'efficacité des règles d'association. De ce fait, cette table doit subir à un ensemble des opérations de nettoyages et d'épurations de données telles que :

A. Des opérations pour améliorer l'efficacité de l'exploration de données :

1. Suppression des attributs moins utiles.
2. Suppression des anomalies ou élimination des enregistrements en double.

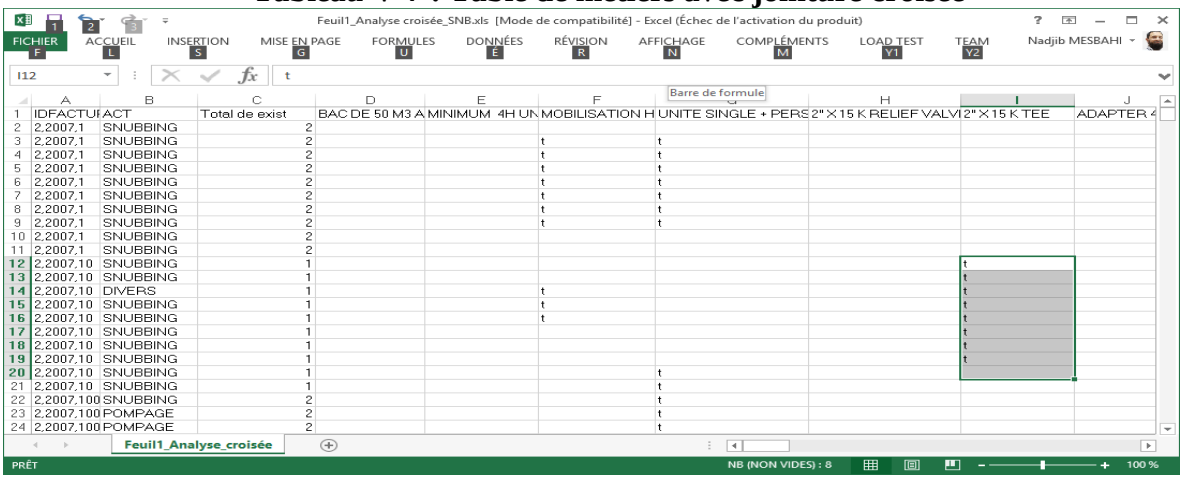
3. Réduction des données pour avoir moins de valeurs possibles.

B. Des opérations pour améliorer la qualité de données

1. Récupérer des données incomplètes : remplir les valeurs manquantes, ou réduire l'ambiguïté.
2. Purifier les données : corriger les erreurs ou supprimer les valeurs aberrantes (valeurs inhabituelles ou exceptionnelles).
3. Résoudre les conflits de données : recours aux connaissances d'un expert métier dans le cas de divergences.

Une fois les données de la table de métiers sont nettoyées, il est nécessaire de la transformer en format utilisable pour l'algorithme de l'extraction des règles d'association « Apriori » en faisant une jointure croisée.

Tableau V-7 : Table de métiers avec jointure croisée



	A	B	C	D	E	F	G	H	I	J
1	IDFACTUR	ACT	Total de exist	BAC DE 50 M3 A	MINIMUM 4H UN	MOBILISATION H	UNITE SINGLE + PERS	2" X 15 K RELIEF VALVI	2" X 15 K TEE	ADAPTER 4
2	2,2007,1	SNUBBING	2							
3	2,2007,1	SNUBBING	2			t	t			
4	2,2007,1	SNUBBING	2			t	t			
5	2,2007,1	SNUBBING	2			t	t			
6	2,2007,1	SNUBBING	2			t	t			
7	2,2007,1	SNUBBING	2			t	t			
8	2,2007,1	SNUBBING	2			t	t			
9	2,2007,1	SNUBBING	2			t	t			
10	2,2007,1	SNUBBING	2							
11	2,2007,1	SNUBBING	2							
12	2,2007,10	SNUBBING	1						t	
13	2,2007,10	SNUBBING	1						t	
14	2,2007,10	DIVERS	1			t			t	
15	2,2007,10	SNUBBING	1			t			t	
16	2,2007,10	SNUBBING	1			t			t	
17	2,2007,10	SNUBBING	1						t	
18	2,2007,10	SNUBBING	1						t	
19	2,2007,10	SNUBBING	1						t	
20	2,2007,10	SNUBBING	1				t			
21	2,2007,10	SNUBBING	1				t			
22	2,2007,100	SNUBBING	2				t			
23	2,2007,100	POMPAGE	2				t			
24	2,2007,100	POMPAGE	2				t			

Par la suite, cette table doit être convertie en format de fichier plat « .Arff » utilisable par l'outil Weka 3.8.

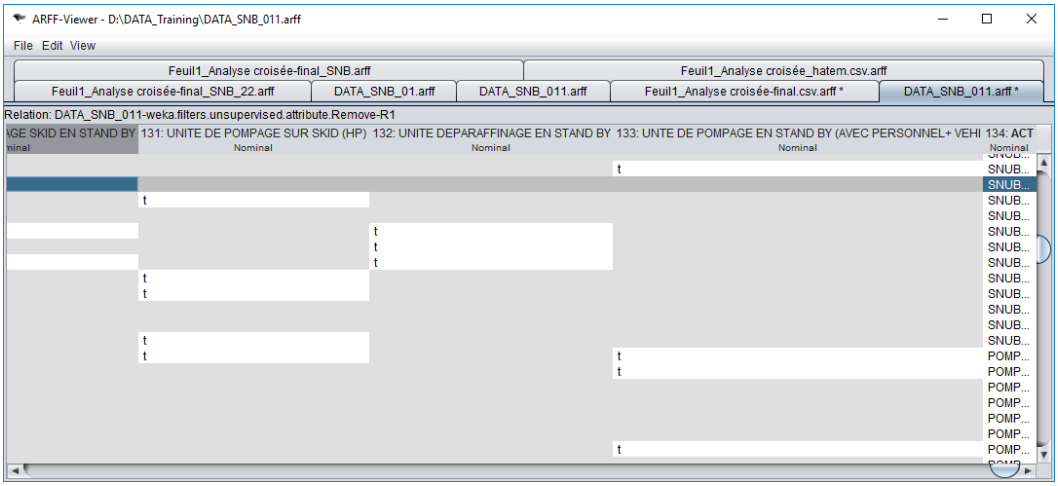


Figure V-20 : Table de métiers sous Weka (En format .Arff)

La table métiers en format « .Arff » comporte plus de 135 attributs comme des colonnes et plus de centaine de milliers d'enregistrements comme des lignes. Dans le but d'expérimenter l'approche proposée, quatre échantillons de la table métiers ont été préparées,

afin de comparer la performance du système développé avec un algorithme classique « Apriori » qui est très connu pour l'extraction des règles d'association. Les paramètres pris pour différencier entre les quatre échantillons sont les suivants : l'ancienneté des données, le nombre d'enregistrements, le nombre d'attributs, la taille de la table. Les caractéristiques de quatre échantillons de données ERP métiers sont présentées dans les tableaux de 8 à 11.

Tableau V-8 : Table métiers-Echantillon 01

Caractéristiques de la table 8 de métiers	
Ancienneté des données	2 ans
Nombre d'attributs	135 attributs
Nombre d'enregistrements	24 887 lignes
Taille de la table	2 MB
Nom de la table	Table8Métiers

Tableau V-9 : Table métiers-Echantillon 02

Caractéristiques de la table 9 de métiers	
Ancienneté des données	4 ans
Nombre d'attributs	135 attributs
Nombre d'enregistrements	49 774 lignes
Taille de la table	3.5 MB
Nom de la table	Table9Métiers

Tableau V-10 : Table métiers-Echantillon 03

Caractéristiques de la table 10 de métiers	
Ancienneté des données	6 ans
Nombre d'attributs	135 attributs
Nombre d'enregistrements	74 661 lignes
Taille de la table	5.5 MB
Nom de la table	Table10Métiers

Tableau V-11 : Table métiers-Echantillon 04

Caractéristiques de la table 11 de métiers	
Ancienneté des données	8 ans
Nombre d'attributs	135 attributs
Nombre d'enregistrements	99 548 lignes
Taille de la table	8.5 MB
Nom de la table	Table11Métiers

V.8.4. Implémentation des sites distribués de données ERP

L'outil de virtualisation « VMware® Workstation 12 Pro » a été utilisé pour créer l'environnement distribué, dans lequel les données ERP de la table métiers sont partitionnées et distribuées sur les différents sites, afin de faire une extraction efficace de règles d'association par métier au profit de l'entreprise ENSP. Au total, trois sites de données sur trois machines virtuelles VMware et un site principal de données sur le système hôte, sont créés dans l'environnement VMware® Workstation. La configuration des sites de données est présentée dans les tableaux de V-11 à V-14.

Tableau V-1 : Description de site de données 1 (système hôte)

Description de site de données 1 (Système hôte).	
Disque dur	500 Go
Mémoire	16 GB
Processeurs	Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz
Type du système	64 bits
Système d'exploitation invité	Windows 10 Professionnel

Tableau V-2 : Description de site de données 2

Description de site de données 2 (VMware)	
Disque dur	100 Go
Mémoire	4 GB
Processeurs	Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz
Type du système	64 bits
Système d'exploitation invité	Windows 8 Ultimate Edition

Tableau V-3 : Description de site de données 3

Description de site de données 3 (VMware)	
Disque dur	100 Go
Mémoire	4 GB
Processeurs	Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz
Type du système	64 bits
Système d'exploitation invité	Windows 8 Ultimate Edition

Tableau V-4 : Description de site de données 4

Description de site de données 4 (VMware)	
Disque dur	100 Go
Mémoire	4 GB
Processeurs	Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz
Type du système	64 bits
Système d'exploitation invité	Windows 8 Ultimate Edition

V.8.5. Interfaces du système développé

La figure suivante montre la première fenêtre qui apparait lors du lancement de notre système. Elle représente la fenêtre de l'agent Interface DM qui indique son démarrage. Dans cette fenêtre, l'agent interface DM invite l'utilisateur Data Miner à entrer le support minimum, la confiance minimale et la source de données ERP afin de lancer l'opération de l'extraction des règles d'association par métier.

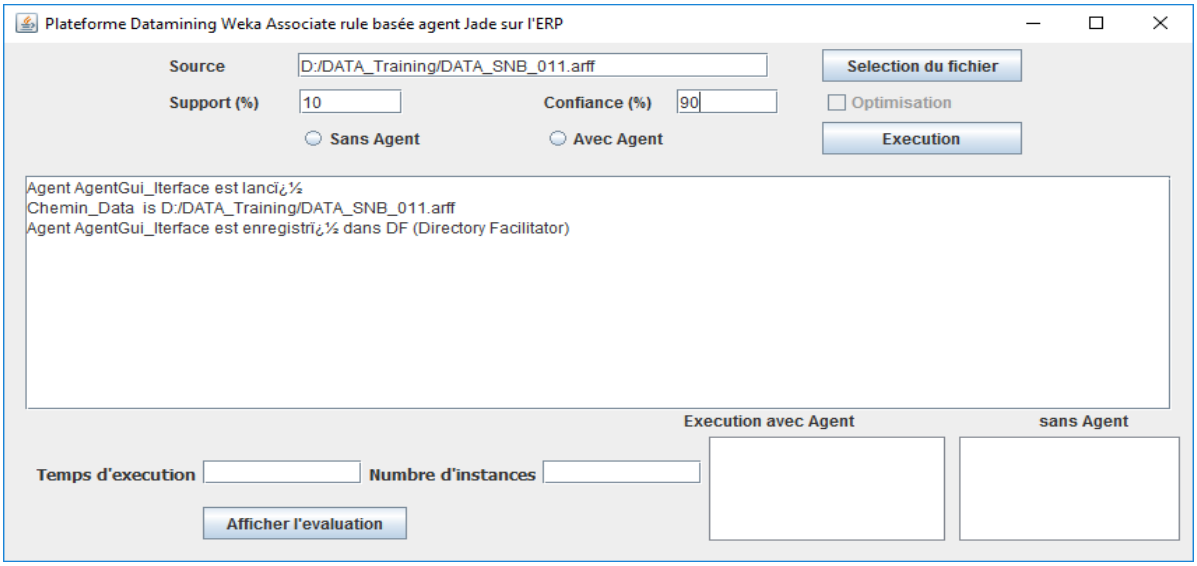


Figure V-21 : L'interface principale du système développé

Une fois les paramètres sont introduits, l'agent utilisateur DM envoi ces paramètres à l'agent de partitionnement de données ERP pour entamer le travail de partitionnement de données ERP. Lorsque cet agent termine son travail, il crée et lance, d'une façon parallèle et distribuée, plusieurs agents de règles d'association métiers. Chaque agent de règles d'association métiers commence l'extraction des règles d'association, en exécutant l'algorithme « Apriori » sur une des partitions de données ERP métiers. Les agents de règles d'association et l'agent de l'interface DM se communiquent et coopèrent pendant tout le processus de l'extraction de règles d'association. À la fin de l'opération, l'agent interface DM affiche le résultat de l'extraction de règles d'association par métier.

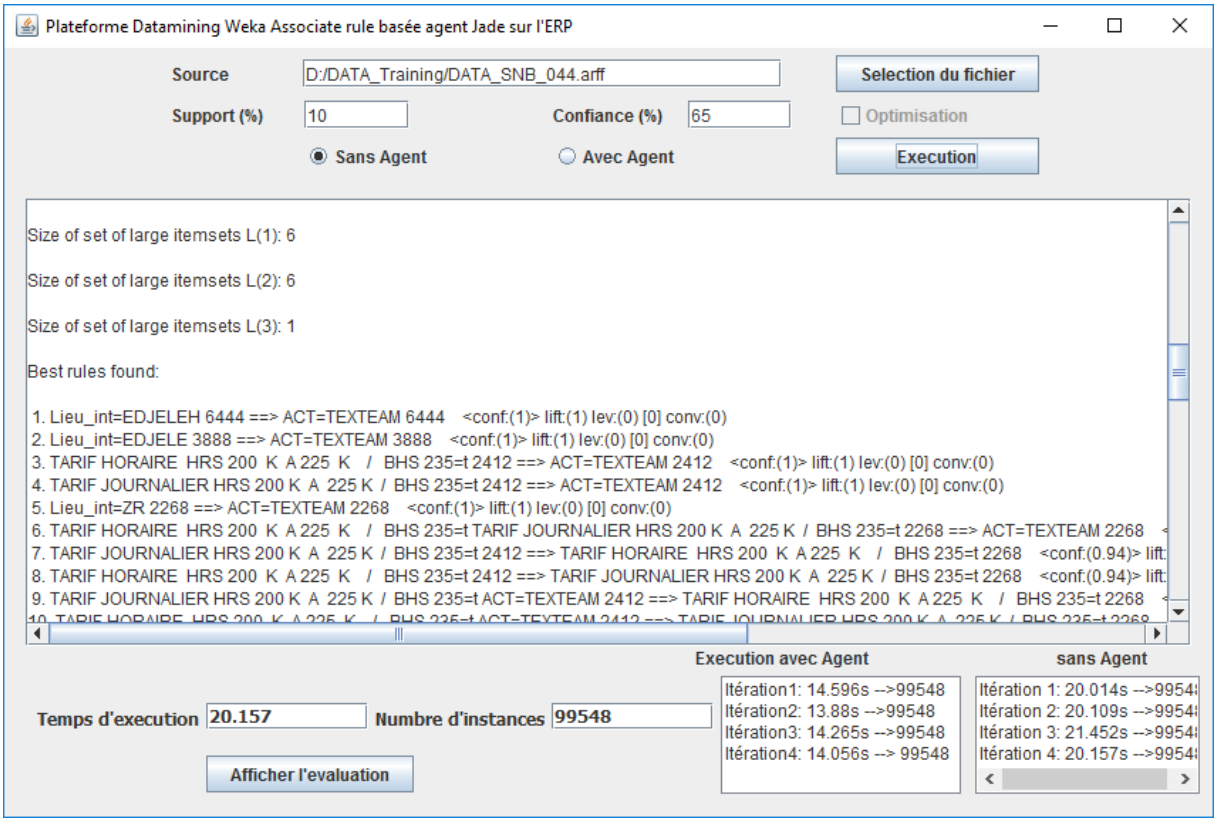


Figure V-22 : Fenêtre du résultat de l'extraction de règles d'association par métier

V.8.6. Expérimentations

Nous avons effectué plusieurs sortes d'expérimentations, afin d'obtenir une riche et fiable évaluation expérimentale. Ces expérimentations sont réalisées sur les quatre échantillons de données ERP de la table métiers de l'ENSP. Les paramètres pris en compte dans cette analyse sont les suivants : la taille des échantillons de données ERP en termes de nombre d'enregistrements, du nombre d'attributs et l'ancienneté des données dans la table métier. Toutes les expérimentations sont accomplies sur une machine hôte et trois machines virtuelles VMware, fonctionnant sur Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz, Processeur 64 bits (s) et sur 16Go de RAM pour le système hôte et 4Go de RAM pour les machines virtuelles, avec des systèmes d'exploitation Microsoft Windows 10 et 8 (Voir les tableaux de 12 à 15). Tous les échantillons de données ERP de la table métiers sont distribués entre le système hôte et les trois machines virtuelles créées. Les éléments suivants ont été mesurés dans nos expériences pour montrer les performances du système développé : (1) le temps de réponse en secondes, (2) le support minimum (3) la confiance minimum et (4) le nombre d'enregistrements des échantillons de données ERP utilisés.

V.8.7. Résultats expérimentaux obtenus

Dans cette section, nous mettons l'accent sur la performance de notre système en fonction des résultats conclus à travers de diverses expériences effectuées sur les échantillons de données ERP de la table métiers. Nous avons abouti à des résultats fiables après avoir modifié les valeurs des paramètres suivants : le support minimum, la confiance minimale et le nombre d'enregistrements des quatre échantillons de la table métiers (Voir les tableaux métiers de V-7 à V-10).

La première expérimentation montre les résultats obtenus en comparant la performance de notre système développé avec l'approche classique qui utilise l'algorithme « Apriori » pour l'extraction des règles d'association. Cette comparaison s'articule autour de la variation du seuil minimum du support de 10% à 25% avec la fixation de la confiance à 90% (Voir la figure V-23 et V-24). Deux échantillons de données ERP ont été employés dans cette expérimentation, à savoir : Tableau V-7 (Table métiers-Echantillon 01) et Tableau V-8 (Table métiers-Echantillon 02).

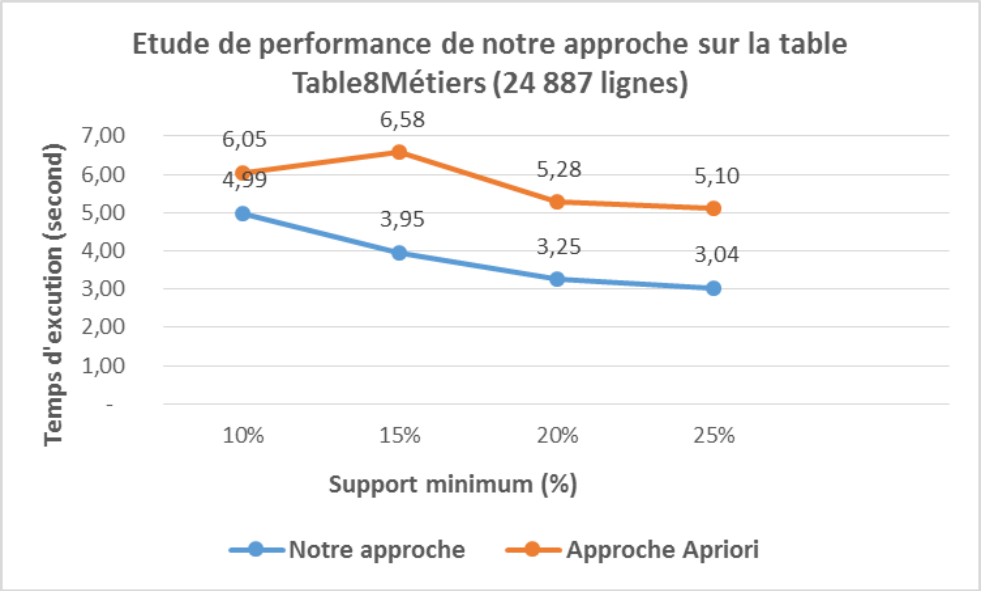


Figure V-23 : Etude de performance par variation du support sur la Table 8

Nous constatons selon le graphe ci-dessus que notre approche est plus performante que celle dite Apriori. Sur la base de la valeur de 10% du support minimum, notre approche « AADARB-ERP » a fait ressortir un gain de temps précieux, puisque le temps d'exécution obtenu est de 4.99 S contre 6.05 S résultant de l'approche « Apriori ». Après avoir effectué une valeur de 25 % du support minimum, notre approche a enregistré un gain de temps de 3.04 S, alors que l'algorithme Apriori a donné un temps de 5,10 S.

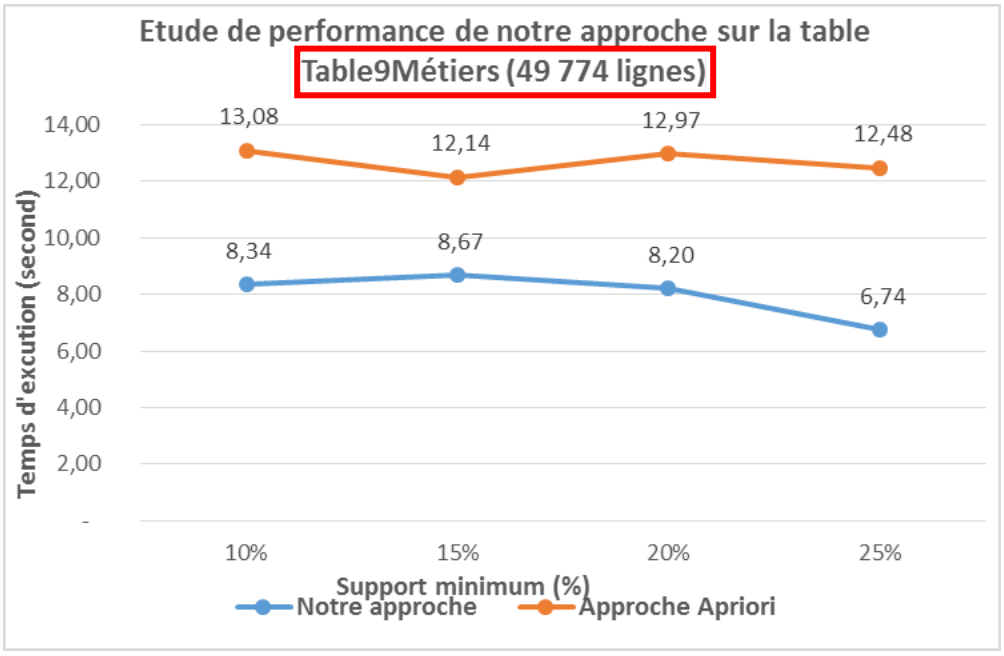


Figure V-24 : Etude de performances par variation du support sur la Table 9

Le graphe ci-dessus indique que notre approche « AADARB-ERP » a permis d'atteindre 8.34 S, où celle dite « Apriori » s'est fixée à un résultat de 13.08 S, sachant que la valeur du support minimum appliquée est de 10%. En outre, le test d'une valeur de 25 % du support

minimum, le temps de réponse achevé par notre approche est de 6.74 S, du moment que l'algorithme Apriori s'est établi à un temps de réponse de 12,48 S.

Grosso modo et à la lumière des deux graphes précédemment illustrés, il nous a été donné de remarquer que notre approche enregistre toujours une meilleure performance en matière de temps de réponse quelles que soient les valeurs introduites.

En ce qui concerne la deuxième expérimentation, nous avons utilisé également les deux échantillons de données ERP à savoir, la Tableau V-9 (Table métiers-Echantillon 03) et Tableau V-10 (Table métiers-Echantillon 04). Elle est consacrée à étudier la performance du système proposé en comparaison avec l'algorithme « Apriori » par la variation de confiance minimale de 50% à 80% et la fixation du support minimum à 10% (Voir la figure 25 et 26).

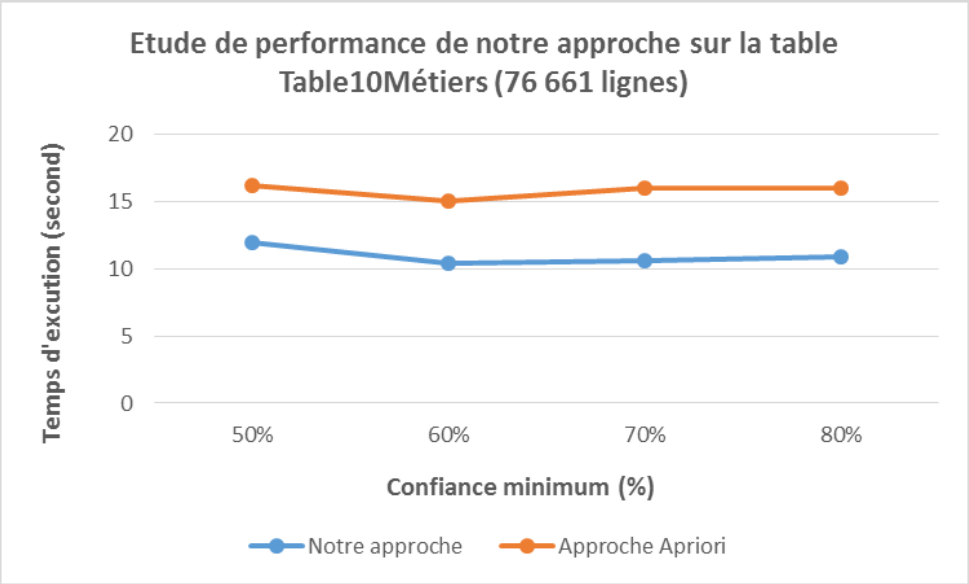


Figure V-25 : Etude de performance par variation de confiance sur la Table 10

Le graphe V-25 ci-dessus montre qu'au pourcentage de 50% de Confiance minimale, notre approche a donné 12.00 S tandis que celle d'Apriori a atteint 16.00 S comme temps de réponse. Lorsque le paramètre de la confiance minimale est fixé à 80 %, notre approche a amélioré le temps de réponse en le régressant à 10.5 S contrairement à l'algorithme Apriori qui s'est maintenu à 16.00 S.

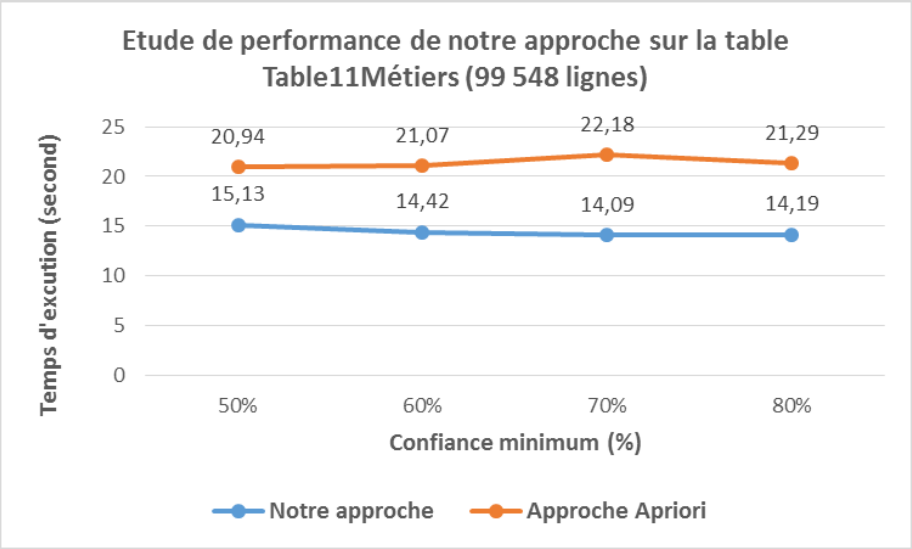


Figure V-26 : Etude de performance par variation de confiance sur la Table 11

Nous constatons d'après le graphe V-26 ci-dessus que notre approche a enregistré 15.13 S du temps d'exécution contre 20.94 S achevé par celle d'Apriori, et ce quand le pourcentage de la confiance minimale est fixé à 50%. Au seuil de 80 % de confiance minimale, le temps de réponse a diminué à 14.19 S grâce à notre approche qui demeure plus efficace par rapport à l'algorithme Apriori qui s'est établi à 21.29 S.

La figure V-27 montre les résultats obtenus dans la troisième expérience en évaluant la performance de notre système comparativement à celle de l'algorithme Apriori. Cette expérimentation repose sur le changement du nombre d'enregistrements des quatre échantillons des données ERP, et la fixation de la confiance minimale à 90% ainsi que le support minimum à 10%.

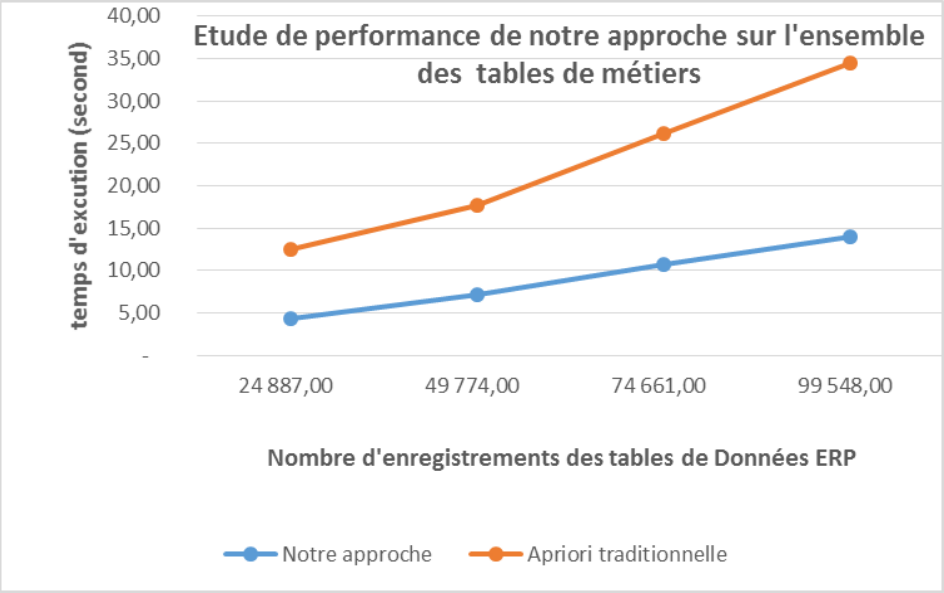


Figure V-27 : Etude de performance de notre approche sur l'ensemble des échantillons des données ERP

D'après la figure V-27 ci-dessus, notre approche a donné un temps d'exécution de 4.5 S et l'algorithme Apriori a atteint 12.5 S, où le nombre d'enregistrements de la base de données

ERP était égal à 24 887.00. Quand ce dernier était l'équivalent de 99 548.00 enregistrements, le temps de réponse de notre approche a atteint approximativement 15 S alors que 35.00 S était enregistré par l'algorithme Apriori.

Dans le tableau ci-dessous «Tableau V 15», des résultats concluants ont été obtenus à travers des autres expérimentations effectuées sur les échantillons des données ERP. Celles-ci se reposent sur le changement des valeurs des paramètres du Support minimum et de la Confiance minimale sur les quatre échantillons des tables métiers.

Tableau V-5 Performance de « AADARB-ERP » par la variation du support et de la Confiance

		Table8Métiers		**Table9Métiers**		**Table11Métiers**		**Table12Métiers**	
		(24 887 Lignes)		(49 774 Lignes)		(74 661 Lignes)		(99 548 Lignes)	
Support Min	**Confiance Min**	**AADARB-ERP**	**Apriori**	**AADARB-ERP**	**Apriori**	**AADARB-ERP**	**Apriori**	**AADARB-ERP**	**Apriori**
5%	90%	4,210 S	5,199 S	7,746 S	10,657 S	11,123 S	15,471 S	15,271 S	21,785 S
10%	90%	3,954 S	5,255 S	7,247 S	10,352 S	10,863 S	15,201 S	14,481 S	20,753 S
15%	90%	3,647 S	5,244 S	7,108 S	10,584 S	10,193 S	15,492 S	13,557 S	20,731 S
20%	90%	3,702 S	5,205 S	6,433 S	10,739 S	9,321 S	15,303 S	12,926 S	21,103 S
10%	95%	4,148 S	5,053 S	7,831 S	10,365 S	11,93 S	16,203 S	14,596 S	20,014 S
10%	85%	3,906 S	5,252 S	7,446 S	10,135 S	10,619 S	15,503 S	13,881 S	20,109 S
10%	75%	3,914 S	5,250 S	7,325 S	10,065 S	10,805 S	15,494 S	14,265 S	21,452 S
10%	65%	3,923 S	4,984 S	7,299 S	10,048 S	10,778 S	15,338 S	14,056 S	20,157 S

Les résultats obtenus dans les deux premières expérimentations montrent que le temps de réponse de notre système d'extraction des règles d'association métiers est plus performant que l'algorithme classique « Apriori ». En outre, Les résultats des dernières expérimentations (Voir Figure V 27 et Tableau V 15) ont affirmé que notre système développé maintient son efficacité, même si le nombre d'enregistrements de données ERP augmente au fil de temps. En revanche, l'approche classique de « Apriori » perd son efficacité en raison de l'augmentation de la base de données ERP lors de l'extraction des règles d'association métiers. En conclusion, les résultats des trois expérimentations effectuées sur les échantillons de données ERP prouvent l'efficacité de notre approche destinée à l'optimisation de l'extraction des règles d'association métiers à partir d'une base de données ERP volumineuse. Cela revient à l'utilisation du système multi-agents qui assure efficacement le partitionnement et la distribution des données ERP sur des machines distinctes.

V.9. Conclusion

Dans ce chapitre, nous avons présenté une nouvelle approche de règles d'association distribuée par métier basée agents pour améliorer le processus de décision dans les systèmes ERP. Cette approche prend en considération la complexité d'extraction des règles d'association métiers dans une base de données volumineuse et centralisée du système ERP.

L'utilisation du paradigme agent dans notre approche donne plus de flexibilité, d'évolutivité, de parallélisme et d'interactions intelligentes entre les différents composants du système. Avec l'augmentation du volume de données ERP, l'extraction des règles d'association métiers devient très difficile et non fiable. C'est pourquoi une distribution après une division des données ERP en partitions s'impose. Dans ce contexte de complexité, l'architecture du système multi-agents proposée est l'un des moyens les plus efficaces pour remédier à cette problématique grâce aux avantages des SMAs. Une application a été développée avec une expérimentation pour valider le système proposé, tout en utilisant plusieurs outils tels que Java, Weka, Jade, et la base de données Postgres de l'ERP Odoo. Les données ERP utilisées dans notre expérimentation appartiennent à l'Entreprise Nationale de Services aux Puits (ENSP) sise au sud Algérien. Les résultats de nos expérimentations ont démontré que l'approche développée « AADARB-ERP » est plus performante que celle classique de l'algorithme « Apriori » en termes de vitesse d'exécution, d'adaptabilité, de flexibilité et de la qualité de présentation des règles d'association par métier. La mise en pratique de notre système améliore le management de l'ENSP notamment lors de la préparation et la prestation des services métiers. Elle peut servir comme un système d'aide à la décision pour les managers et les ingénieurs de cette entreprise.

Conclusion générale et perspectives

Dans le cadre de cette thèse, nous avons essayé d'apporter des solutions pour adapter certaines tâches de Data Mining non supervisées grâce à l'utilisation des systèmes multi-agents, tout en respectant les spécificités du système ERP. Ce dernier est principalement fondé sur une grande base de données ERP (en terme de volume) centralisée où le recours aux traitements parallèles distribués est indispensable afin d'extraire des modèles de connaissances efficaces et performants. Reposant sur ces hypothèses et face à la diversité des méthodes existantes dans le domaine d'extraction de connaissances, nous avons fait un tour d'horizon des approches les plus connues à base d'agents pour les tâches de clustering et de règles d'association, en comparant leurs points forts et leurs points faibles.

Néanmoins, la plupart des approches proposées dans ce contexte utilisent des données de nature distribuée, où les données doivent être traitées sur leurs sites, en raison de plusieurs contraintes, à savoir : le coût de stockage, la communication et la sécurité. Par contre, les approches proposées dans le cadre de cette thèse traitent des données centralisées vu la nature de la base de données ERP. Elles proposent deux architectures à base d'agents pour l'extraction des connaissances décisionnelles à partir de données métiers ERP.

L'utilisation du système Multi agents est primordiale dans le cadre de notre travail de recherche, puisqu'il permet de distribuer la complexité des tâches de Data Mining sur plusieurs entités autonomes et coopératives. Les caractéristiques intrinsèques des agents telles que : la distribution, la collaboration, la flexibilité, l'évolutivité et l'efficacité rendent leur utilisation très appropriée dans des systèmes hétérogènes, complexes et distribués. En effet, l'intégration des agents coopératifs au sein du processus de Data Mining a permis de produire des modèles de connaissances plus performants. Mais aussi, elle a rendu l'exécution des algorithmes du Data Mining d'une manière parallèle et distribuée pour une extraction efficace de connaissances décisionnelles dont la base de données ERP est volumineuse et centralisée. Tout au long de notre étude nous avons abordé deux problématiques liées au Data Mining non supervisé : le clustering et les règles d'association autour d'un progiciel de gestion intégré.

Dans la première problématique de clustering de données ERP, nous avons proposé une approche multicouche à base d'agents, qui est fondé sur le parallélisme et la distribution des tâches. Cette approche a permis la distribution de la complexité de l'algorithme k-means sur plusieurs agents coopératifs dans le principal but est de regrouper les enregistrements de la base de données ERP en groupes similaires (clusters).

Concernant la deuxième problématique de règles d'association, nous avons proposé une nouvelle approche à base d'agents de règles d'association distribuée par métier dans le but d'améliorer le processus de décision dans les systèmes ERP. L'approche a obtenue également des meilleurs résultats grâce au partitionnement intelligent et à la distribution de données métiers ERP sur des machines distinctes. Elle est également capable d'exécuter efficacement de multiples processus d'extraction des règles d'association métiers, de façon parallèle et distribuée, à partir de grande base de données ERP.

Une application a été développée et expérimentée afin de valider l'approche proposée, en utilisant plusieurs outils logiciels tels que Java, Weka, Jade, ERP Odoo et la base de données ERP Postgres. Notre expérimentation est réalisée sur des données ERP réelle de l'Entreprise Nationale de Services aux Puits (ENSP). En fait, les résultats de l'expérimentation ont montré que le système développé est meilleur que le système classique qui utilise l'algorithme « Apriori ». Ainsi, la mise en pratique de notre système sur le terrain peut réduire la charge pendant la préparation de services aux puits en termes de temps et de précision. Cette approche peut être intégrée dans la fonction décisionnelle (BI) pour faciliter la prise de décision au profit des managers de l'entreprise ENSP.

Nos approches proposées sont ouvertes du fait que ce travail ne s'arrête pas à ce niveau et que des perspectives d'évolution s'offrent à nous :

1. Implémentation et expérimentation de la première approche de clustering de données ERP «CMAAC-ERP » sur une étude de cas réel pour valider notre modèle élaboré.
2. Extension de notre travail en intégrant des ontologies tout au long du processus Data Mining à base d'agents. Objectif, enrichir les résultats de l'extraction de connaissances par l'expérience d'un expert du métier.
3. Adoption des autres techniques des règles d'association dans notre approche «AADARB-ERP » pour généraliser notre modèle actuel.
4. Remplacement du partitionnement « Agents par métier » de la deuxième approche proposée «AADARB-ERP » par le clustering de données à base d'agents «CMAAC-ERP ». Pour mesurer l'efficacité de cette démarche, il sera nécessaire d'évaluer les performances en termes de la qualité de présentation des règles et de vitesse d'exécution.
5. Enrichissement de la deuxième approche de l'extraction des règles d'association métiers «AADARB-ERP » par l'intégration de clustering de données au niveau de chaque sous-ensemble de données métiers. Cela vise à améliorer la qualité des règles métiers extraites et accélérer le traitement des données via l'exécution parallèle de plusieurs agents de règles d'association sur chaque sous-ensemble de données.

Bibliographie

[1] A. Hammami, P. Burlat, J. Pierre, Contribution à la conception et au pilotage d'une entreprise réseau, 3e Conférence Francophone de MOdélisation et SIMulation "Conception, Analyse et Gestion des Systèmes Industriels" MOSIM'01, Apr 2001, 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] 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, 2004.

[4] D. Bentley, "Independent vs. dependent demand, MRP inputs and processing, Closed-loop MRP, MRP II, ERP", San José State University, Washington, 2006.

[5] J-L. Lequeux, Manager avec les ERP , Editions d'organisation, 2008.

[6] M. Abbas, ERP Systems in HEI Context from a Multiple Perspective View: A Case Study, 2011.

[7] F-A. Blain, Présentation générale des ERP et leur architecture modulaire, http://fablain.developpez.com/tutoriel/presenterp/, 20/03/2006.

[8] M. E. Shacklett, L'essentiel sur SAP Business Suite L'essentiel sur SAP Business Suite», http://www.lemagit.fr/conseil/Lessentiel-sur-SAP-Business-Suite, consulter le 21-02-2016.

[9] S. Baudry, Développement sur l'ERP OfbizNéogia, Rapport de stage, Polytechnique de l'Université de Tours, 2005.

[10] K. ishak, Architecture distribuée interopérable pour la gestion des projets multi-sites. Application à la planification des activités de production », Thèse Présentée en vue de l'obtention du grade de docteur, université de Toulouse, 2010.

[11] J. François, Planification des chaînes logistiques : Modélisation du système décisionnel et performance, Thèse de Doctorat à l'Université Bordeaux I, Volume 17 décembre 2007.

[12] MARKESS International, Attentes des entreprises pour les solutions de gestion intégrée ERP/PGI face aux nouveaux enjeux, Référentiel de pratiques (2011-2013), http://www.markess.fr, 2013.

[13] C. Lassiette, « Les progiciels de gestion intégrée,Article, Centre de Recherche en Economie et Gestion (CREG) Paris», http://www.creg.ac-versailles.fr/article.php3?id_article=209, consulter le 23\02\2016.

[14] PGPP, Progiciel de Gestion pour la PME Principaux ERP propriétaires et Open Source, version 1.0, 2006.

[15] N. Kale, Chapter 8 : erp system architecture», chapter of book ERP System and Enterprise Architecture, University of Southern California, 2012.

[16] C. Lassiette, Les progiciels de gestion intégrée,Article, Centre de Recherche en Economie et Gestion (CREG) Paris», http://www.creg.ac-

versailles.fr/article.php3?id_article=209, Consulté le 27\09\2007.

[17] A. larionova, gestion des processus de la création d'une maison erp, Université nationale d'économie de Kharkiv Simon Kuznets, Ukraine, Université Lumière Lyon 2, France, 2014.

[18] C.LEROY, EDITO - La fin de l'ERP généraliste "tout en un" ?, Chief Editor Groupe CXP, Conseil et analyse en solutions logicielles pour l'entreprise et ses métiers, France, http://www.cxp.fr/content/news/edito-la-fin-de-lerp-generaliste-tout-en-un, Consulté le 05/11/2014.

[19] P.Rahali, Senior Analyst ,Groupe CXP,EDITO – L'ERP métier, la solution pour un ROI rapide ?, http://www.cxp.fr/content/news/edito-lerp-metier-la-solution-pour-un-roi-rapide, Consulté le 16/10/2015.

[20] R. Beretz, « Les ERP verticalités, gages d'efficacité ? », http://www.erp-infos.com/info_article/m/2031/les-erp-verticalises-gages-defficacite%20.html, publié le 03/06/2013

[21] La couverture fonctionnelle des ERP pour les PME, http://www.voxime.com/quel-erp-pour-quel-metier.html?PHPSESSID=7eb43c5814e02d4e2aeecd566fb80f17, Consulté le 01/02/2015.

[22] Inconvénients ERP, http://www.guidepme.com/article514/Inconvenients-ERP, Consulté le 14/06/2014.

[23] LES PROGICIELS DE GESTION INTEGREE, http://www.creg.ac-versailles.fr/IMG/pdf/progiciels_gestion_integree.pdf

[24] I. Ehie, et M. Madsen, Identifying critical issues in enterprise resource planning (ERP) implementation, Computers in Industry, Volume 56, issue 6, p. 545 557, 2005.

[25] A. Chaabouni, Implantation d'un ERP (Enterprise Resource Planning) : antécédents et conséquences, AIMS, XVème Conférence Internationale de Management Stratégique, Annecy / Genève 13-16 Juin 2006.

[26] S. M. Glover, D. F. Prawitt, et M. B. Romney, Implementing ERP: Internal auditing can help eliminate mistakes that commonly derail organizations' ERP initiatives. Internal Auditor, Volume 56, p. 40-47, 1999.

[27] C. Ash, et J. Burn,. A strategic framework for the management of ERP enabled e-business change. European Journal of Operational Research, Volume 146 Issue 2, p. 374-387, 2003

[28] P.,Bingi, M. K. Sharma, et Godla, Critical issues affecting an ERP implementation. Information Systems Management, Volume 16 Issue 3, p. 7-14. 1999.

[29] E. Shehab, M. Sharp, L. Supramaniam, et T. A. Spedding, Enterprise resource planning: An integrative review. Business Process Management Journal, Volume 10 Issue 4, p. 359-386, 2004.

[30] G. Shanks, A. Parr, B. Hu, B. Corbitt, T. Thanasankit, et P. Seddon, Differences in critical success factors in ERP systems implementation in australia and china: A cultural analysis. Proceedings of the 8th European Conference on Information Systems, p. 537-544, 2000.

[31] D. S. Bajwa, J. E. Garcia, et T. Mooney, An integrative framework for the assimilation of enterprise resource planning systems: Phases, antecedents, and

outcomes. Journal of Computer Information Systems, Volume 44 issue 3, 81-90, 2004.

[32] S. Shang, P. Seddon, Assessing and managing the benefits of enterprise systems: the business manager's perspective, Journal of Information Systems, Volume 12. 2002.

[33] P. Grabski, Financial impact of entreprise ressource planning implementations, international journal of accounting information systems Volume 2, p. 271-294, 2001.

[34] M-L. B. Gomez A. Frot, Duwer Quels effets organisationnels pour les ERP, Actes de conférence AIMS, p. 1-24, 2002.

[35] F. FOURATI, Veille stratégique : de l'évaluation de l'utilisation des agents intelligents à la prise de décision , Thèse de doctorat, Université Paris Dauphine, 2006.

[36] X. Yan, C. Zhang, and S. Zhang, Towards data bases mining: Pre-processing collected data. Applied Artificial Intelligence Volume 17 Issue 5–6 p. 545–561, 2003.

[37] https://fr.wikipedia.org/wiki/Weka_(apprentissage_automatique), date consultation 22/11/2015.

[38] C. Paraschiv, : Les agents intelligents pour un nouveau commerce électronique, InterEditions, Paris, 2004.

[39] T. YUAN, Software Agents, Introduction to JADE, 2008.

[40] M. S. Chen, J. Han, et P. S. Yu, Data Mining: An Overview from a Database Perspective. IEEE Transactions on Knowledge and Data Engineering, p. 866-883, Volume 8, issue 6,1996.

[41] The Gartner Group, www.gartner.com.

[42] P. f, P. Hadjinian, R. Stadler, J. Verhees et A. Zanasi, Discovering Data Mining: From Concept to Implementation, Prentice Hall, Upper Saddle River, NJ, 1998.

[43] E-G. TALBI, Fouille de données (Data Mining) : Un tour d'horizon, Laboratoire d'Informatique Fondamentale de Lille, 2012.

[44] S. Tufféry, Data Mining et statistique décisionnelle, Editions Technip, 2007.

[45] U. Fayyad, S. Piatetsky, Gregory et S. Padhraic: The KDD process for extracting useful knowledge from volumes of data. ACM New York, NY, USA. Volume Volume 39, 11. 1996.

[46] H. A. Edelstein, Introduction to Data Mining and Knowledge Discovery (3rd ed). Potomac, MD: Two Crows Corp. 1999.

[47] N. Pasquier, Data Mining : Algorithmes d'extraction et de réduction des régles d'association dans les bases de données, Thèse de doctorat, université Clermont-Ferrand 2, 2000.

[48] D. A. Zighed, Y. Kodratoff, and A Napoli, "Extraction de connaissance à partir d'une base de donnée," Bulletin AFIA'01, 2001

[49] J. Wijsen: Data Mining et Data Warehousing, 2001.

[50] U. Fayyad: Data Mining and Knowledge Discovery, Kluwer Academic Publishers., Volume 2 , Issue 1, p. 5 – 7, 1998.

[51] H. J. Miller, et H. Jiawei, Geographic Data Mining and knowledge discovery: an

overview. New York : Taylor & Francis, 2001.

[52] R. Osmar, S. Zaïane, Introduction to Data Mining, Chapter I, University of Alberta, Department of Computing Science, p. 1-15, 1999.

[53] C. Kenneth, B. Carey, E. Grusy, C. Marjaniemi, et D. Sautter, A Perspective on Data Mining. Northern Arizona University, 1998.

[54] M. J. Berry, G. S. Linoff, Data Mining Techniques For Marketing, Sales, and Customer Relationship, Management, Second Edition, 2004

[55] M. J. Berry, G. S., Linoff, Mastering Data Mining: The Art and Science of Customer Relationship Management, 2000.

[56] B. Jouve, « fouille de données » éléments de cours master 2 université de Lyon, France. 2012.

[57] D.T. Larose, Des données à la connaissance, une introduction au datamining, Editions Vuibert Informatique, Paris, 2005.

[58] V. Fiolet, algorithmes distribués d'extraction des connaissances, Université des sciences et technologie de Lille, Thèse de doctorat, France, 2006.

[59] D. DeWitt et J. Gray, Parallel database systems: the future of high performance database systems, Communications of the ACM, Volume 35, Issue 6, p. 85–98, 1992.

[60] P. Valduriez, Parallel database systems: Open problems and new issues , Distributed and parallel Databases, Volume 1, Issue 2, p. 137–165, 1993.

[61] M. Tommasi et R. Gilleron, « Découverte de connaissances à partir de données, cours des outils, des techniques liées à l'informatique décisionnelle, 2000.

[62] T. Zhang, R. Ramakrishnan, et M. Livny, BIRCH: an efficient data clustering method for very large databases », SIGMOD international conference on Management of data p. 103-114, Montréal, Canada, June 04-06, 1996.

[63] S. Guha, R. Rastogi, et K. Shim, CURE : an efficient clustering algorithm for large databases, ACM SIGMOD international conference Management of Data, p. 73-84, 1998.

[64] G. Karypis, E.-H. Han, et V. Kumar,Chameleon: Hierarchical clustering using dynamic modeling, IEEE Computer, Volume 32, Issue 8, p. 68–75, 1999.

[65] E. W. Forgy, Cluster analysis of multivariate data: efficiency versus interpretability of classifications, Biometrics, Volume 21, p. 768–769, 1965.

[66] R. T. Ng et J. Han, CLARANS: A method for clustering objects for spatial data mining, Knowledge and Data Engineering, IEEE Transactions on, Volume 14, Issue 5, p. 1003–1016, 2002.

[67] L. Breiman, J. H. Friedman, R. A. Olshen, et C. J. Stone, Classification and Regression Trees, Wadsworth, Belmont, California, 1984.

[68] J. R. Quinlan, C4. 5: programs for machine learning, Morgan Kaufmann, Volume 1, San Francisco, CA, 1993.

[69] M. Mehta, R. Agrawal, et J. Rissanen, « SLIQ: A fast scalable classifier for Data Mining, Advances in Database Technology—EDBT'96, Springer, p. 18–32, 1996.

[70] J. Shafer, R. Agrawal, et M. Mehta, SPRINT : A scalable parallel classi er for data mining, The 22th International Conference on Very Large Data Bases, p. 544-555,

September 03-06, 1996.

[71] V. Ganti, J. Gehrke, R. Ramakrishnan, et W.-Y. Loh, A framework for measuring differences in data characteristics, Journal of Computer and System Sciences, Volume 64, Issue 3, p. 542–578, 2002.

[72] R. Agrawal, R. Srikant, et others, Fast algorithms for mining association rules, 20th International Conference on Very Large Data Bases, p.487-499, September 12-15, 1994.

[73] A. Savasere, E. R. Omiecinski, et S. B. Navathe, An efficient algorithm for mining association rules in large databases The 21th International Conference on Very Large Data Bases, p. 432-444, September 11-15, 1995.

[74] J. M. Delorme, « L'apport de la fouille de données dans l'analyse de texte », Conservatoire National des Arts et Métiers, Centre régional de Montpellier, Avril 2002.

[75] R. Agrawal, T. Imieliński, et A. Swami, Mining association rules between sets of items in large databases. International Conference on Management of Data , Volume 22 Issue 2, p. 207-216. 1993.

[76] C. T. Diop, M. Lo, et al, Intégration de règles d'association pour améliorer la recherche d'informations XML. Quatrième conférence francophone en Recherche d'Information et Applications. École Nationale Supérieure des Mines de Saint Étienne, 2007.

[77] M. Abdelali, et O. Hicham Création de règles d'association. Caen, Ensicaen. 2003.

[78] R. GILLERON, M. TOMMASI, Découverte de connaissances à partir de données, 2000.

[79] K. E. Belbachir, Data Mining distribué sur les grilles de données : application de règles d'association, Université des sciences et technologie d'Oran, Thèse de doctorat, Algerié, 2016.

[80] S. Ben Yahia, and E. Mephu Nguifo Approches d'extraction de règles d'association basées sur la correspondance de Galois. Lens, Centre de Recherche en Informatique de Lens, 2004.

[81] C. Marinica, F. Guillet, et al. Vers la fouille de règles d'association guidée par des ontologies et des schémas de règles. QDC 2008. École polytechnique de l'université de Nante. 2008.

[82] H. Cherfi, and Y. Toussaint Adéquation d'indices statistiques à l'interprétation de règles d'association. Actes des 6es Journées internationales d'Analyse statistique des Données Textuelles. Saint-Malo.Volume l, 2002.

[83] K. Aby, and A. EL Kourri, Post traitement de règles d'association. Caen, ISMRA ENSI Caen, 2003.

[84] J. Han et Y. Fu, Discovery of multiple-level association rules from large databases, 21th Int'l Conf. Very Large Data Bases, p. 420-431, Sept 1995.

[85] J. S. Park, M.-S. Chen, et P. S. Yu, An effective hash-based algorithm for mining association rules, ACM, Volume 24. 1995.

[86] J. Han, J. Pei, Y. Yin, et R. Mao, Mining frequent patterns without candidate generation: A frequent-pattern tree approach, Data Mining and knowledge discovery,

Volume 8, Issue 1, p.53–87, 2004.

[87] M. Tommasi et R. Gilleron, « Découverte de connaissances à partir de données », cours des outils, des techniques liées à l'informatique décisionnelle, 2000.

[88] M. Zaki, S. Parthasarathy, M. Ogihara, et Wang., L. New algorithms for fast discovery of association rules. 3rd Intl. Conf. on Knowledge Discovery and Data Mining, p. 283-296, 1997.

[89] H. Toivonen et others, Sampling large databases for association rules », 22th International Conference on Very Large Data Bases, p.134-145, , 1996.

[90] S. Brin, R. Motwani, J. D. Ullman, et S. Tsur, Dynamic itemset counting and implication rules for market basket data , ACM SIGMOD international conference on Management of data, p.255-264, May 11-15, 1997, Tucson, Arizona, USA, 1997.

[91] M. Plasse, N. Niang-Keita, et G. Saporta, Utilisation conjointe des règles d'association et de la classification de variables. Rapport de recherche, 2006.

[92] M.J. Zaki. Parallel and distributed association mining : A survey. IEEE Concurrency, 7(4) :14-25, /1999.

[93] R. Agrawal et J. C. Shafer, Parallel mining of association rules, IEEE Transactions on knowledge and Data Engineering, Volume 8, Issue 6, p. 962–969, 1996.

[94] D. W. Cheung, J. Han, V. T. Ng, A. W. Fu, et Y. Fu, A fast distributed algorithm for mining association rules, Fourth international conference Parallel and Distributed Information Systems, p. 31–42, December 1996.

[95] D. W. Cheung, V. T. Ng, A. W. Fu, et Y. Fu, Efficient mining of association rules in distributed databases, Knowledge and Data Engineering, IEEE Transactions on, Volume 8, Issue 6, p. 911–922, 1996.

[96] M. Z. Ashrafi, D. Taniar, et K. Smith, ODAM: An optimized distributed association rule mining algorithm, IEEE distributed systems online, Volume 5, Issue 3, p. 1–18, 2004.

[97] E.-H. Han, G. Karypis, et V. Kumar, Scalable parallel Data Mining for association rules , Knowledge and Data Engineering, IEEE Transactions on, Volume 12, Issue 3, p. 337–352, 2000.

[98] M. V. Joshi, E.-H. S. Han, G. Karypis, et V. Kumar, Efficient parallel algorithms for mining associations. Springer, 2000.

[99] T. Shintani et M. Kitsuregawa, Hash based parallel algorithms for mining association rules , 4th international conference Parallel and Distributed Info. Systems, p. 19–30, December 1996.

[100] M. J. Zaki, S. Parthasarathy, M. Ogihara, et W. Li, Parallel algorithms for discovery of association rules , Data Mining and Knowledge Discovery, Volume 1, Issue 4, p. 343–373, 1997.

[101] M. Halkidi, Y. Batistakis, et M. Vazirgiannis, « On clustering validation techniques », Journal of Intelligent Information Systems, Volume 17, Issue 2-3, p. 107–145, 2001.

[102] S. Bourakache, Un environnement sémantique à base d'agent pour la formation à distance (E-Learning), Thèse de doctorat, Université Biskra, 2014.

[103] longbing cae, Data Mining and multi agents integration,2008

[104] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, From Data Mining to Knowledge Discovery in Databases, American Association for Artificial Intelligence, 0738-4602, 1996.

[105] J. Ferber, Multi-agent Systems — An Introduction to Distributed Artificial Intelligence,Addison-Wesley, London, 1999.

[106] A. Symeonidis, C. Chatzidimitriou, I. Athanasiadis, M. Pericles, Data mining for agent reasoning: A synergy for training intelligent agents, Engineering Applications of Artificial Intelligence, Elsevier, Volume 20, Issue. 8, p. 1097- 1111, 2007.

[107] Z. Zhang, C. Zhang, S. Zhang, An agent-based hybrid framework for database mining, Journal of Applied Artificial Intelligence, Volume 17, Isssue. 5-6, p. 383-398, 2003.

[108] M.,Mohammadiam, Intelligent Agents for Data Mining and Information Retrieval, Idea Group Inc, 2004.

[109] N. P. Trilok, P. Niranjan and K. S. Pravat , Improving performance of distributed Data Mining with multi-agent system, International Journal of Computer Science, Volume 9, Issue 3, p. 74-82, 2012.

[110] S. Chaimontree, Multi-Agent Data Mining with Negotiation: A Study in Multi-Agent Based Clustering , These de doctorat ,University of Liverpool, 2012.

[111] L. Lao, Data Mining and Multi-agent Integration, Faculty of Engeneering and Information, Sydney university, Springer, 2007.

[112] A. Pradesh, Multi agent-based distributed data mining: an over view, Research Scholar, CSIT Department, JNT University, Hyderabad., INDIA , Volume 3 Issue 11, 2010.

[113] J. Ferber, “Les systèmes multi-Agents – vers une intelligence collective”, inter éditions 1995.

[114] W.M. Wooldridge, An Introduction to multi-agent systems, University of Liverpool UK. 2002.

[115] H. S. Nwana, Software Agents: An Overview, Knowledge Engineering Review, Volume 11, Issue 3, p. 205-244, 1996.

[116] M. Grabner, F. Gruber, L. Klug, W. Stockner ,EvalAgents, D2.1 - Agent technology: State of the Art,2000.

[117] Y. Demazeau, « From Interactions to Collective Behaviour in Agent-Based Systems », Proceeding of the First European Conference on Cognitive Science, Saint Malo, p. 117-132, 1995.

[118] S. J Russel, P. Norvig, Artificial intelligence : A modern Approach, 1995.

[119] S. Benharzallah, Thèse de Doctorat, Coopération multi agents pour le traitement des requêtes sur des sources de données hétérogènes et distribuées, Université Mentouri de Constantine, 2005.

[120] B.chaib-draa, Agents et systèmes multiagents (IFT 64881A), Note de cours, département d’informatique faculté des sciences et de génie, université LAVAL QUEBEC, 1999.

[121] I. Jarras et B. Chaib-draa, Aperçu sur les systèmes multiagents, Série scientifique, Centre interuniversitaire de recherche en analyse des organisations (CIRNO), Université de Montréal, 2002.

[122] G. Weiss, "Multi-agent systems: A modern Approach to distributed artificial intelligence", MIT Press Cambridge UK, 1999.

[123] H. Mazyad, Une Approche Multi-agents à Architecture P2P pour l'Apprentissage Collaboratif, Thèse Pour l'obtention du titre de Docteur de l'Université du Littoral Côte d'Opale, 2013.

[124] J. Denis, Thèse de Doctorat : « Délégation de Rôle et Architectures Dynamiques de Systèmes Multi-Agents Conversationnels» , Université Lyon I – Claude Bernard, 2003.

[125] D. Nessah., « un modèle de raisonnement pour un système de recherche sémantique d'informations sur le web basé agents », Thèse de Doctorat, l'Université de biskra, 2014.

[126] FIPA Communicative Act Library Specification, 2000.

[127] J. G. Carbonell and J. Siekmann, Lecture Notes in Artificial Intelligence, Agent-Oriented Information Systems III 7th International Bi-ConferenceWorkshop, AOIS 2005 Utrecht, ISBN10 3-540-48291-1 Springer Berlin Heidelberg NewYork, 2006.

[128] S. Chaimontree, K. Atkinson, and F. Coenen, Multi-agent based clustering: Towards generic multi-agent data mining. In Advances in Data Mining. Applications and Theoretical Aspects,p. 115-127, Springer Berlin Heidelberg, 2010.

[129] F. Coenen, P. Leng, S. Ahmed, T-trees, vertical partitioning and distributed association rule mining. In: Proc. 3rd IEEE Int. Conf. on Data Mining. ICDM '03, IEEE Computer Society, Washington, DC, USA, p. 513– 516, 2003.

[130] J. Dasilva, C. Giannella, R. Bhargava, H. Kargupta,M. Klusch, Distributed Data Mining and agents. Engineering Applications of Artificial Intelligence Volume 18, Issue 7, p.79-–807, 2005.

[131] G. Forman, B. Zhang, Distributed data clustering can be efficient and exact. ACM SIGKDD Explorations Newsletter 2, 34–38 (2000).

[132] B.H. Park, H. Kargupta, Distributed data mining: Algorithms, Systems, and Applications. In: Data Mining Handbook. p. 341–358. IEA, 2002.

[133] F. Provost, Distributed data mining: Scaling up and beyond. In: In Advances in Distributed and Parallel Knowledge Discovery. p. 3–27. MIT Press,1999.

[134] Younis, O., Fahmy, S.: Distributed clustering in ad-hoc sensor networks: a hybrid, energy-efficient approach. In: INFOCOM 2004. 23rd Annual Joint Conf. of the IEEE Computer and Communications Societies. Volume 1, p. 629–640, 2004.

[135] H. Kargupta, P. Chan, (eds.): Advances in Distributed and Parallel Knowledge Discovery. MIT Press, Cambridge, MA, USA, 2000.

[136] M. Zaki, Parallel and distributed association mining: an introduction. In: Proceedings of the large-scale parallel data mining, lecture notes in artificial intelligence 1759. Berlin, Germany: Springer-Verlag; p. 1–23. 2000.

[137] M.J. Zaki, Y. Pan, Introduction: Recent developments in parallel and distributed data

mining. Distributed Parallel Databases Volume 11, p.123–127, 2002

[138] H. Kargupta, B. Stafford, and I. Hamzaoglu, Web Based Parallel/Distributed Medical Data Mining Using Software Agents, Proceedings of 1997 Fall Symposium, American Informatics Association, 1997.

[139] H. Kargupta, I. Hamzaoglu, and B. Stafford, Scalable, Distributed Data Mining Using an Agent Based Architecture, Proceedings of Knowledge Discovery and Data Mining, AAAI Press (1997) 211–214.

[140] S. Bailey, R. Grossman, H. Sivakumar, and A. Turinsky. Papyrus: A System for Data Mining over Local and Wide Area Clusters and Super-Clusters. Published in: Supercomputing, ACM/IEEE 1999 Conference. Date of Conference: 13-19 Nov. 1999.

[141] S. Ben Yahia, G. Gasmi, and E. Mephu Nguifo. A New Generic Basis of Factual and Implicative Association Rules. Intelligent Data Analysis (IDA), Volume 13, Isue 4, p. 633–656, 2009.

[142] I. Brahmi, S. Ben Yahia, H. Aouadi, et P. Poncelet, Towards a Multiagent-Based Distributed Intrusion Detection System Using Data Mining Approaches, Proceedings of International Workshop on Agents and Data Mining Interaction, Volume 7103, pp. 13-194. Springer 2011.

[143] M. Klusch, S. Lodi, and G. Moro. Agent-based distributed data mining: The kdec scheme. In Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), Volume 2586, p. 104-122, 2003.

[144] J. W. Reed, T. E. Potok, and R. M. Patton. A multi-agent system for distributed cluster analysis. In Proceedings of Third International Workshop on Software Engineering for Large-Scale Multi-Agent Systems (SELMAS'04) W16L Workshop - 26th International Conference on Software Engineering, p.152–155, Edinburgh, Scotland, UK, IEE, 2004.

[145] Q. Cen, J. Zhao, and X. Zhu., The Data Mining system based on multi-agent under the circumstance of e-commerce. In Proceedings - Third International Conference on Natural Computation, ICNC 2007, volume 3, p.34–38, 2007.

[146] K. Albashiri, F. Coenen, and P. Leng. Emads: An extendible multi-agent data miner. Journal of Knowledge Based Systems, Volume 22, Issue 7, p. 523–528, 2009.

[147] I. Czarnowski and P. Jdrzejowicz, Agent-based non-distributed and distributed clustering. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics),5632LNAI:347–360, 2009.

[148] TF. Gharib, H. Nassar, M. Taha, A. Abraham, An efficient algorithm for incremental mining of temporal association rules. Data Knowledge Eng; Volume 69:800,15. 2010

[149] F. Ariwa, B. Mohamed, M. Mohamed. Informatization and Ebusiness model application for distributed Data Mining using mobile agents. International Conference WWW/Internet 2003,2003.

[150] K.A. Albashiri, F. Coenen, P. Leng, An investigation into the issues of Multi-Agent Data Mining. In Bouca, D. and Gafagnao, A. (Eds), Agent Based Computing, Nova Science Publishers, ISBN: 978-160876-684-0, 2010.

[151] F. Coenen, P. Leng, Optimising association rule algorithms using itemset ordering. In: Proceedings of the AI Conference, Research and Development in Intelligent Systems XVIII. Springer, p. 53–66. 2006.

[152] V. Pudi, J. Haritsa, ARMOR: Association rule mining based on Oracle. In: ICDM workshop on frequent itemset mining implementations, Florida, USA; 2003.

[153] KA. Albashiri, EMADS: an investigation into the issues of multiagent data mining. PhD Thesis, The University of Liverpool, Liverpool L69 3BX, United Kingdom; 2010.

[154] S. McConnell, D. Skillicorn, Building predictors from vertically distributed data. In: Proceedings of the 2004 conference of the centre for advanced studies conference on collaborative research 04-07. Markham, Ontario, Canada, p. 150–62, 2004.

[155] KA. Albashiri, Agent based data distribution for parallel association rule mining. International Journal of Computers; Volume 8, 24–32, 2014.

[156] R. Kishore, H. Zhang, Enterprise integration using the agent paradigm: foundations of multi-agent-based integrative business information systems. International Journal of Decision Support Systems, Volume 42, Issue 1, 48-78. 2006.

[157] J. Kaufmann, P.J. Rousseeuw. Finding groups in data : an introduction to cluster analysis. Wiley, 1990.

[158] J. Dimple S. Aarti, S. Rashmi, M. Saurabh, A Thorough Insight into Theoretical and Practical Developments in MultiAgent Systems. International Journal of Ambient Computing and Intelligence. Volume 8, Issue 1, 23-49, 2017.

[159] R. Agrawal, J. Gehrke, D. Gunopulos, et P. Raghavan, «Automatic subspace clustering of high dimensional data for Data Mining applications», ACM. Volume 27, 1998.

[160] DW. Cheung, Y Xiao, Effect of data skewness in parallel mining of association rules. In: Proceedings of the 2nd Pacific-Asia conference on knowledge discovery and data mining, Melbourne, Australia, p. 48–60, 1999.

[161] S. Goil, H. Nagesh, et A. Choudhary, « MAFIA: Efficient and scalable subspace clustering for very large data sets », Technical Report CPDC-TR-9906-010, Northwestern University, Evanston IL June, 1999.

[162] M. Deypir, MH, Sadreddini. Distributed association rules mining using non-derivable frequent patterns .Iran J Sci Technol, Trans B: Eng Volume 33, issue 6 511–26. 2009

[163] h. Necir h. DRIAS, Approche Data Mining pour la gestion de la relation client : application à la personnalisation d'un site de e-commerce, 4th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications, SETIT 2007, Tunisia March 25-29,2007.

[164] A. Adhikari, J. Adhikari, W. Pedrycz, Data analysis and pattern recognition in multiple databases. Intelligent systems reference library 61, @ Springer International Publishing, Switzerland; 2014. p. 21–42.

[165] C. Aflori, F.Leon. Efficient Distributed Data Mining using Intelligent Agents . In Proceedings of the 8th International Symposium on Automatic Control and Computer Science , pages 1–6, 2004.

[166] A.O. Ogunde a, *, O. Folorunso b, A.S. Sodiya A partition enhanced mining

algorithm for distributed Association rule mining systems, Egyptian Informatics Journal Volume 16, 297–307, 2015.

[167] A. O. Ogunde, O. Folorunso, A. S. Sodiya, J. A. Oguntuase & G. O. Ogunleye, " Improved cost models for agent based association rule mining in distributed databases", Anale SEria Informatica,Volume 9, no. 1, pp. 231–250,2011

[168] Y. L. Wang, Z. Zhi Li, and Hai-Ping Zhu. Mobile agent based distributed and incremental techniques for association rules. In Proceedings of the International Conference on Machine Learning and Cybernetics(ICMLC 2003), volume 1, pages 266–271, 2003.

[169] U. P. Kulkarni, P. D. Desai, Tanveer Ahmed, J. V. Vadavi, and A. R. Yardi. Mobile Agent Based Distributed Data Mining. In Proceedings of the International Conference on Computational Intelligence and Multimedia Applications(ICCIMA 2007), pages 18–24. IEEE Computer Society, 2007.

[170] G. Hu, and D. Shaozhen, An Agent-Based Framework for Association Rules Mining of Distributed Data. In Roger Lee and Naohiro Ishii, editors, Software Engineering Research, Management and Applications 2009, volume 253 of Studies in Computational Intelligence, pages 13–26. Springer Berlin - Heidelberg, 2009.

[171] S. B. Gurpreet, "A framework for association rule mining of distributed data", These de doctorat, Thapar Universtiy Patiala 147004, Regn. No. 90703508, India, 2015.

[172] Frank A, Asuncion A. UCI machine learning repository. Irvine, CA: University of California, School of Information and Computer Science; 2010.

[173] S. Chaimontree, K. Atkinson, and F. Coenen. A multi-agent based approach to clustering: Harnessing the power of agents. In Proceedings of the 7th international conference on Agents and Data Mining interaction, ADMI'11, pages 12–29, Berlin, Heidelberg, 2011. Springer-Verlag LNAI 7103.

[174] A. S. Al-Mudimigh, U. Zahid, S. Farrukh, A Framework of an Automated Data Mining Systems Using ERP. Model International Journal of Computer and Electrical Engineering, Volume l, Issue. 5 December, 2009 1793-8163.

[175] S. Tiwari", S. Choudhary, Architecture of an Automated CBA System Using ERP Model" International Journal of Scientific and Research Publications, Volume 2, Issue 10, ISSN 2250-3153, October 2012.

[176] A. Almalaise, C. Alghamdi, Rules Generation from ERP Database: A Successful Implementation of Data Mining, Faculty of Computing and Information Technology, King Abdulaziz University, IJCSNS International Journal of Computer Science and Network Security, Volume 12 Issue 3, March 2012

[177] A. S. Al-Mudimigh, S. Farrukh, U. Zahid, "The Effects Of Data Mining In ERP-CRM Model – A Case Study Of MADAR" International Journal of Education and Information Technologies, Volume 3, Issue 2, pp 135-144- 2009.

[178] A. S. Al-Mudimigh, S. Farrukh, U. Zahid, "The Effects Of Data Mining In ERP-CRM Model – A Case Study Of MADAR" WSEAS Transactions on Computers, Volume 8 Issue 5Pages 831-843, May 2009.

[179] N. Mesbahi, O. Kazar, « Une approche Multi-Agents pour la modélisation d'un progiciel de gestion intégré (ERP) », 2e Conférence Internationale sur les "Systèmes d'Information et Intelligence Economique", SIIE 2009 Hammamet, Tunisie 12-14,

page 704-724,Février 2009.

[180] R. Guttman, G. Alexandros, and M. Pattie: Agent-mediated Electronic Commerce: A Survey, Software Agents Group, MIT Media Laboratory, 1998.

[181] W. Benchikh, « L'Architecture Contractuelle des Groupements de contrats de Progiciel de Gestion Intégrée (PGI/ERP) », Mémoire de D.E.E.S, Université de Paris Π, 2004.

[182] D. CHAMI, Une plate forme orientée agent pour le data mining, mémoire de Magister, Université de Batna, Algerié, 2010.

[183] M. Ester, H.-P. Kriegel, J. Sander, et X. Xu, « A density-based algorithm for discovering clusters in large spatial databases with noise. », 2nd international conference Knowledge Discovery and Data Mining (KDD'96), p. 226-231, 1996.

[184] A. Boudina, L. Tayeb, « L'intelligence Artificielle Distribuée et les SystèmesMulti-Agentff»,htttp://opera.inrialpes.fr/people/Tayeb.Lemlouma/Papers/ IAD_Presentation.pdf, Février 2007.

[185] F. Adam, P. O'Doherty, «Lessons from enterprise resource planning implementation in Ireland» – toward smaller and shorter ERP projects, Journal of Information Technology, 2000, 15, pp. 305-316.

[186] P. Paquet, «De l'information à la connaissance», Cahier de recherche, Université d'Orléans, 2006.

[187] G. Hébrail, Y. Lechevallier, «Data Mining et Analyse des données in Analyse des données », G.Govaert éditeur, Hermes, 2003, pp. 323-355.

[188] A. Hinneburg et D. A. Keim, « An efficient approach to clustering in large multimedia databases with noise », 4th international conference Knowledge Discovery and Data Mining (KDD'98), p. 58-65, 1998.

[189] W. Wang, J. Yang, R. Muntz, et others, « STING: A statistical information grid approach to spatial Data Mining », The 23rd International Conference on Very Large Data Bases, p.186-195, August 25-29, 1997

[190] J-L. Lequeux, « Manager avec les ERP », Editions d'organisation, 1998.

[191] O. Kazar « Conception et réalisation d'un modèle de réseau sémantique », Mémoire de magister, pp.6-45, Université de Constantine, 1996.

[192] R. Rahmani, Découverte d'associations sémantiques dans les bases de données relationnelles par des méthodes de Data Mining, Mémoire de magister, 2003

[193] G. Sheikholeslami, S. Chatterjee, et A. Zhang, « Wavecluster: A multi-resolution clustering approach for very large spatial databases », 24th VLDB Conf., pp. 428-439, 1998.

[194] M. H. Haddad, Extraction et Impact des connaissances sur les performances des Systèmes de Recherche d'Information, Thèse doctorale. Université Joseph Fourier, Grenoble, France, 2002.

[195] D. Louani, DATAMINING : Objectifs, Méthodes et Outils, LSTA, Université de Paris 6 et Université de Reim, 2010.

[196] N. Mekroud, Integration des techniques du Datamining dans le processus de gestion des connaissances basée sur le raisonnement à partir de cas, Mémoire de magister,

Université de Ferhat Abbes -Sétif, 2009.

[197] P. Preux, Fouille de données Notes de cours, Université de Lille 3, 2011.

[198] S. saidna, « Plates-formes des systèmes multi-agents », Master de recherche en Informatique, Université Paris Sud XI, 2007.

[199] F. Y. Villemin, Agent Communication Language, Systèmes Intelligents NFP212, Année 2009-2010.

[200] M. Emre Celebi, Hassan A. Kingravi, Patricio A. Vela, A Comparative Study of Efficient Initialization Methods for the K-Means Clustering Algorithm, September 2012

[201] N. Mesbahi, and O. Kazar, (2014-B). A cooperative multi-agent approach for knowledge discovery from the enterprise resource planning. Paper presented in the 3rd IEEE International Workshop on Advanced Information Systems for Enterprises (IWAISE'14). Tunis, Tunisia.

[202] N. Mesbahi, and O. Kazar., (2015). Multi-Agents approach for Data Mining based k-Means for improving the decision process in the ERP systems. International Journal of Decision Support System Technology. 7 (2), 1-14. (**Disponible en ligne**).

[203] J. McQueen. Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, AD 669871, Univ. of California Press, Berkeley, Volume 1, pages 281297, 1967.

[204] R. T. Ng and J. Han. Efficient and effective clustering methods for spatial data mining. In Proc. of the VLDB Conference, Santiago, Chile, September 1994.

[205] A. El Golli, B. Conan-Guez, F. Rossi, et others, « Self-organizing maps and symbolic data », Journal of Symbolic Data Analysis, Volume 2, Issue 1, 2004.

[206] Z. Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. In Data Mining and Knowledge discovery, Volume 2, pages 283-304 ,1998.

[207] A. P. Dempster, N. M. Laird, et D. B. Rubin, « Maximum likelihood from incomplete data via the EM algorithm », Journal of the Royal Statistical Society. Series B (Methodological), p. 1–38, 1977.

[208] S Dasgupta, P. M. Long. Performance guarantees for hierarchical clustering. Journal of Computer and System Sciences archive : Special issue on COLT 2002 , Volume 70 , Issue 4 :555-569, June 2005.

[209] E. Diday, G. Celeux, G. Govaert, Y. Lechevallier, et H. Ralambondrainy, « Classification Automatique des Données », Dunod, Paris, Volume 49, p. 80, 1989.

[210] N. Mesbahi, O. Kazar, S. Benharzallah, M. Zoubeidi, D. Rezki. A clustering approach based on cooperative agents to improve decision support in ERP. Technological Innovations in Knowledge Management and Decision Support. IGI Publishing Hershey, PA, USA, 2018.

[211] R. Reix, et F. Rowe. La recherche en Systèmes d'Information: de l'histoire au concept. In ROWE, F. Faire de la recherche en Systèmes d'Information. Paris, Ed. Vuibert, FNEGE, Septembre 2002, p.355

[212] J. Chaabouni. le concept de performance dans les théories du management », Actes de Colloque, FSEG Sfax, 1992.

[213] M. Kalika. Structure de l'entreprise, réalité, déterminants et performance, Economica, Paris, 1988.

[214] K. JONES. L'ABC de la gestion intégrée : Guide d'introduction pour les dirigeants. 2006.

[215] T. Willis-Brown, et A. Mcmillan. Stratégies de maîtrise des coûts lors de l'implantation de systèmes ERP, Revue Française de Gestion Industrielle, Issue 1, Volume 22, 2003.

[216] L.M. MARKUS, et C. TANIS. The Entreprise System Experience_from adoption to succes », in Framing the domains of IT management, R.W.Zmud Editor, Pinnaflex, Cincinatti. 2000.

[217] A. Rabaa'i, W. Bandara and G. Gable. ERP systems in the higher education sector: a descriptive study. In Proceedings of the 20th Australasian Conference on Information Systems, Monash University: Caulfield Campus, Melbourne, pp. 456-470. 2009.

[218] B. Vincent, et S. Gharbi. Impact du déploiement de SAP R/3 sur la performance globale d'une entreprise et facteurs clés de succès: proposition d'un tableau de bord et application dans le secteur de l'industrie pharmaceutique, Journée de recherche à l'IAE de Montpelier, 1-27, 2004.

[219] R. Marciniak. Piloter les technologiques de l'informatique et des télécoms – modèles et outils, ouvrage collectif, éditions Weka, 2001.

[220] N. Mesbahi, O. Kazar, S. Benharzallah, M. Zoubeidi, D. Rezki, and A. Merizig. A new approach agent-based for distributed of association rules by business to improve the decision process in ERP systems. International Journal of Data Mining, Modelling and Management, Volume. xx, Issue. xx, 2018 Pages xx- xx – Inderscience (**En cours d'evaluation**).

[221] N. Mesbahi, O. Kazar, S. Benharzallah, M. Zoubeidi. A Cooperative Multi-Agent Approach-Based Clustering in Enterprise Resource Planning. International Journal of Knowledge and Systems Science, Volume 6 Issue 1, January 2015, Pages 34-45, IGI Publishing Hershey, PA, USA. (**Disponible en ligne**).