

# Étude et analyse des plateformes de maintenance existantes

---

1. Introduction .....	35
2. Principales plateformes de maintenance existantes .....	36
2.1- Introduction .....	36
2.2- Plateformes de Projets.....	36
2.3- Plateformes académiques.....	47
2.4- Tableau de synthèse.....	56
3. Conclusion.....	60



## 1. Introduction

Une plateforme de e-maintenance est caractérisée principalement par la coopération et l'intégration des composants logiciels intelligents de support, tandis qu'une plateforme de s-maintenance prend appui sur la collaboration et l'intégration de fonctionnalités autonomes (auto-apprentissage et autogestion du processus de maintenance). Les points clés communs entre ces plateformes sont l'intégration, la synchronisation et la prise en compte du processus complet de maintenance.

Dans l'objectif de mettre en place une plateforme de s-maintenance, nous nous sommes intéressés aux plateformes existantes classées sous la dénomination e-maintenance.

Avant toute chose, nous faisons un tour d'horizon sur les différentes études et revues qui ont été faites sur l'élaboration de plateforme logicielle concernant des projets de maintenance aussi bien théoriques qu'académiques.

Dans le cadre de la maintenance conditionnelle (CBM) Jardine et al (Jardine, Daming, & Banjevic, 2006) ont présentés une revue sur les systèmes de diagnostic et de pronostic, en mettant l'accent sur les modèles, les algorithmes et les technologies utilisés pour le traitement des données et la prise de décision.

Campos (Campos, 2009) a fait une étude sur l'application et l'intégration des technologies d'informations et de communications et plus spécifiquement sur les outils d'intelligence artificielle, les technologies des agents et le web dans le cadre de la surveillance conditionnelle et la maintenance. Campos classe ces systèmes selon les technologies utilisées et les catégorise par rapport aux couches OSA-CBM et conclut à la limitation d'exploitation des TIC dans ce cadre.

D'autre part, Levrat et al (Levrat, Iung, & Crespo-Marquez, 2008) ont étudié les différentes définitions de la e-maintenance et ont proposé un cadre de e-maintenance basé sur le framework de Zachman (Sowa & Zachman, 1992), dans l'objectif de donner une méthodologie de déploiement d'un projet de e-maintenance à l'aide de services, de processus, d'organisation et d'infrastructures. Cette étude était plutôt orientée sur la e-maintenance. Par contre, dans (Muller, Marquez, & Iung, 2008) Muller et al après avoir examiné les différents points de vue et définitions de la e-maintenance, ont passé en revue la plupart des systèmes de e-maintenance existantes en les classant selon les objectifs sous jacents à leurs constructions et leurs champs d'exploitation.

Dans ce chapitre, nous allons faire un état de l'art sur les plateformes de e-maintenance existantes qu'elles soient théoriques ou issues du monde industriel.

Par ailleurs, l'étude que nous ferons dans la deuxième section de ce chapitre analyse ces plateformes suivant quatre caractéristiques en lien direct avec la définition du concept de s-maintenance à savoir :

1. Le type de maintenance traité (CBM, diagnostic, processus complet, ...)
2. L'interopérabilité entre les applications de la plateforme
3. Services dynamiques et ingénierie des connaissances au cœur de la plateforme
4. Le degré d'intégration (fonction de coopération, collaboration)

Cette analyse a comme objectif l'identification de la ou des plateformes en concordances avec la définition de la plateforme de s-maintenance. Nous en déduisons ainsi s'il y a lieu de recycler une plateforme de e-maintenance ou de redévelopper une nouvelle plateforme de s-maintenance avec sa propre architecture.

## **2. Principales plateformes de maintenance existantes**

### **2.1- Introduction**

Plusieurs plateformes de e-maintenance ont été développées ces dernières années, et elles sont opérationnelles aujourd'hui (CASIP, ICAS-AME, INID, WSDF, PROTEUS, TELMA, etc.). Muller et al (Muller, Marquez, & Jung, 2008) constatent qu'il y a quatre types de plateformes : les plateformes propriétaires (i.e. ICAS), les plateformes développées dans des projets européens (i.e. PROTEUS, DYNAMITE) ou bien des plateformes de recherche et/ou d'éducation (i.e. TELMA) et enfin des plateformes issues d'étude théorique (Ribeiro, Barata, & Silvério, 2008) (Han & Yang, 2006). Ils classent ces contributions selon les capacités et les besoins qui sont destinés à être satisfait. Ces différentes contributions sont développées pour la e-maintenance afin de répondre à une (ou plusieurs) des quatre questions suivantes: (1) l'établissement de normes, (2) la conception d'une plateforme de e-maintenance, (3) la formalisation des processus de e-maintenance, et (4) la mise en place de système d'e-maintenance (c.à.d. plateforme + processus).

Nous allons étudier dans cette section deux types de plateformes à savoir des plateformes de projets instanciés entre des industriels et des universitaires et des plateformes académiques lancées par des universitaires et des groupes de chercheurs.

### **2.2- Plateformes de Projets**

Nous ferons l'étude de sept plateformes définies dans le cadre de projets d'envergures : le projet international MIMOSA, les projets européens ESPRIT-REMAFEX, PROTEUS, PROMISE, SAMMART, et DYNAMITE et finalement le projet suédois e-Maintenance 24/7–NFFP4.

#### **2.2.1- Le projet MIMOSA**

La première initiative était prise par le projet MIMOSA (The Machinery Information Management Open Systems Alliance) pour développer un système complexe d'information unique pour la gestion de la maintenance (Kahn, 2003). Le projet a eu pour objectif de développer un réseau de collaboration de maintenance en proposant la norme open de protocole EAI (Enterprise Application Integration). L'organisation préconise et développe des caractéristiques d'intégration de l'information pour permettre la gestion et le contrôle de la valeur ajoutée par les solutions ouvertes, intégrées et orientées vers l'industrie. Les solutions développées à partir des îlots d'informations afin de créer une plateforme d'e-maintenance ont été proposées dans ce projet (Mitchell, Bond, Bever, & Manning, 1998).

Une architecture fonctionnelle OSA/CBM (Open System Architecture for Condition-Based Maintenance) (Thurston, 2001) dédiée au développement de stratégies de maintenance conditionnelle ou prévisionnelle (Lebold & Thurston, 2001) a été développée à partir du schéma relationnel MIMOSA CRIS pour répondre au

besoin d'une norme concernant le flux transactionnel de l'information entre les composants logiciels dans un système CBM et en vue de pour garantir l'interopérabilité entre eux. Elle contient sept modules flexibles dont le contenu (méthodologie et algorithmes) est configurable par l'utilisateur (Figure 2-1). Elle peut être simplifiée et adaptée à chaque besoin industriel en réduisant des modules.

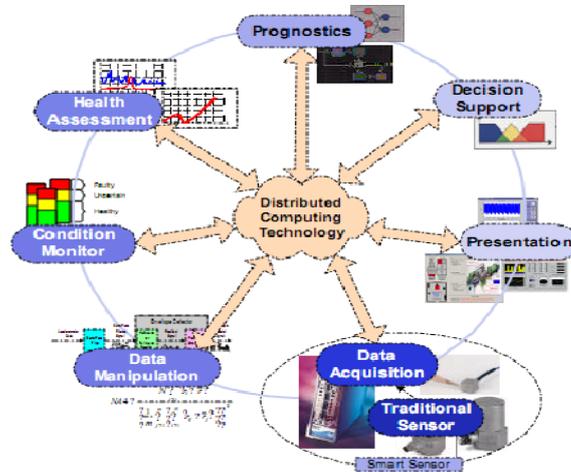


Figure 2-1. Architecture MIMOSA- OSA/CBM (Lebold & Thurston, 2001)

#### Type de maintenance :

En effet, OSA/CBM a mis le point sur les différentes couches présentant le cadre complet du processus de maintenance mais sans prendre en considération la partie gestion. Ainsi il ne traite qu'un seul type de processus à savoir le CBM (Condition Based Maintenance).

#### Interopérabilité :

Concernant l'aspect d'interopérabilité, MIMOSA a traité l'interopérabilité syntaxique en fournissant le schéma de données partagé CRIS. Mais ceci n'assure pas un niveau d'interopérabilité sémantique.

#### Gestion des connaissances :

Aucune démarche d'ingénierie ou de management de connaissances ni de services dynamiques évoluant selon les besoins de l'utilisateur n'est donné. En effet, MIMOSA s'intéresse à la normalisation et la standardisation des flux transactionnels dans un système complexe de maintenance.

#### Mécanisme de synchronisation :

Dans MIMOSA on parle de mécanisme de collaboration sans plus de détail. Toutefois, il n'y a pas vraiment du partage de ressource, ce qui nous oriente plutôt vers une coopération entre les composants de ce système.

### **2.2.2- POMAESS**

Dans le cadre du projet ESPRIT-REMAFEX, Yu et al proposent un système d'e-maintenance basé sur un système multi-agents fournissant une aide à la décision grâce au raisonnement à partir des cas (Yu, Iung, & Panetto, 2003). Celui-ci se nomme «Problem-Oriented Multi-Agent based E-Service System (POMAESS) ». Chaque agent dans ce système est un agent expert de résolution de problèmes (homme ou machine autonome) avec des connaissances différentes et une localisation dans des sites différents.

Le but principal de POMAESS est de créer un système qui interconnecte les agents séparés (agents réactifs et cognitifs), leur donner la possibilité de coopérer entre eux dans un environnement ouvert tout en respectant des contraintes temporelles. Ce qui permettra ainsi à l'ensemble des agents à opérer au-delà des capacités de l'un de ses membres et de résoudre les problèmes au sein du processus industriel.

Nous rappelons quand même que l'architecture de la coordination dans POMAESS est une combinaison du modèle de coordination maître-esclave (voir la structure tableau noir et la formation du comité). A cela deux types d'agent sont adoptés par le système, agent de négociation (NA), qui lance et organise la négociation, calcule le résultat mis sur la base des propositions par l'agent expert (EA), mais il ne donne pas son opinion sur la solution du problème. La négociation et la coordination entre agents se fait par rapport a un ensemble de règles déjà définies. L'application de ces règles permet de trouver une solution, qui pourrait être apprise et réutilisée dans d'autres situations (illustration sur la Figure 2-2).

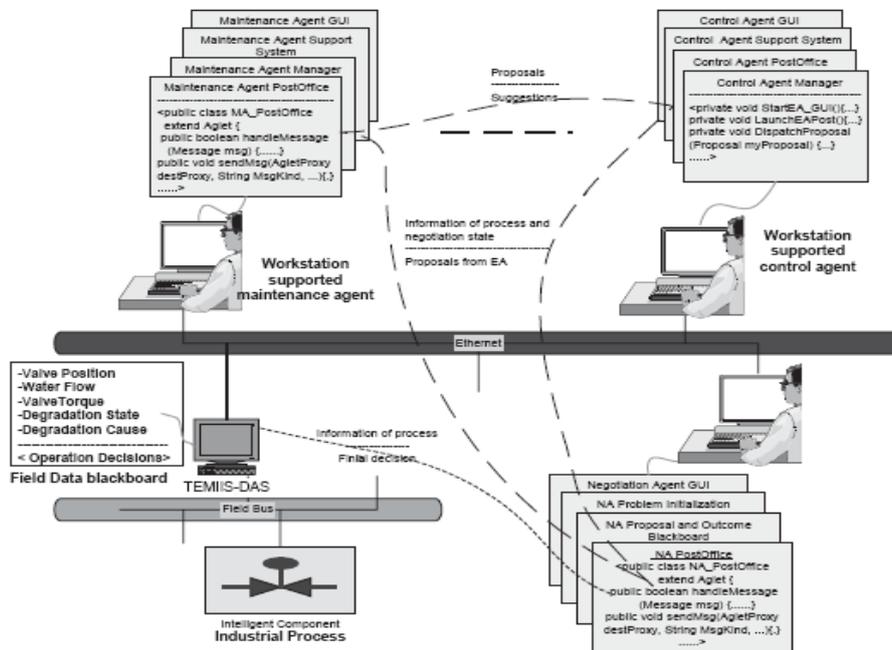


Figure 2-2 Architecture de négociation des agents dans POMAESS (Yu, Jung, & Panetto, 2003)

#### Type de maintenance :

La plateforme proposée dans ce travail vise la résolution des problèmes et est une plateforme d'aide à la décision dédiée principalement au diagnostic, et n'assure pas une gestion complète du processus de maintenance.

#### Interopérabilité :

La négociation entre les agents est faite à base de règles prédéfinies, mais la communication entre ces agents ne paraît pas être standardisée et n'a pas de sémantique partagée.

#### Gestion des connaissances :

Les connaissances dans cette plateforme sont sous forme de règles de négociations, mais la résolution de problèmes n'est pas intégrée comme étant des connaissances globales du domaine de maintenance.

Nous tenons à rappeler ici que chaque agent utilise des connaissances locales et non pas partagées.

En ce qui concerne la proposition de service dynamique à la demande, ceci n'était pas inclus ou pensé dans ce travail. Le service fourni d'aide au diagnostic est statique.

Mécanisme de synchronisation :

Nous savons que les règles de négociations sont partagées entre agents grâce à des mécanismes mis en place. Il n'y a donc pas de collaboration entre agents pour la prise de décision, mais il y a un mécanisme de coopération.

### **2.2.3- PROTEUS**

Dans le cadre du projet Européen PROTEUS une première plateforme d'e-maintenance fournissant des services web a été développée (Bangemann, et al., 2006). Le principal objectif du projet est le développement d'une plateforme logicielle pour l'intégration et l'interopérabilité des modules logiciels de et de maintenance à distance. L'idée originale du projet dédiée à la maintenance industrielle, est liée à l'intégration de tout outil nécessaire à l'activité de maintenance, dont les fonctions vont de la détection des alarmes à la gestion des pièces de rechange, avec comme finalité d'optimiser les coûts et améliorer la productivité de l'équipement maintenu. L'objectif de la plateforme PROTEUS vise à l'intégration de ces sous-systèmes grâce à une description de l'équipement unique et cohérente (à travers une description d'ontologie de l'équipement), une architecture générique (basée sur la technologie "Web Services") et des modèles cohérents de composants hétérogènes et des solutions technologiques d'intégration (Figure 2-3). Ces techniques permettent de garantir l'interopérabilité de systèmes hétérogènes afin d'assurer l'échange et le partage des informations, des données ainsi que des connaissances.

Bien que PROTEUS ait touché plusieurs points et ait été considéré comme une révolution en 2006 dans les systèmes de maintenance, il ne répond plus aux recommandations et aux besoins actuels d'un système de s-maintenance.

Type de maintenance :

L'un des objectifs de PROTEUS est intégrer tous les outils et applications autour de la maintenance et permettant de couvrir tout le processus de maintenance. Sur ce point PROTEUS est l'un des rares projets à avoir une vision complète sur le processus de maintenance.

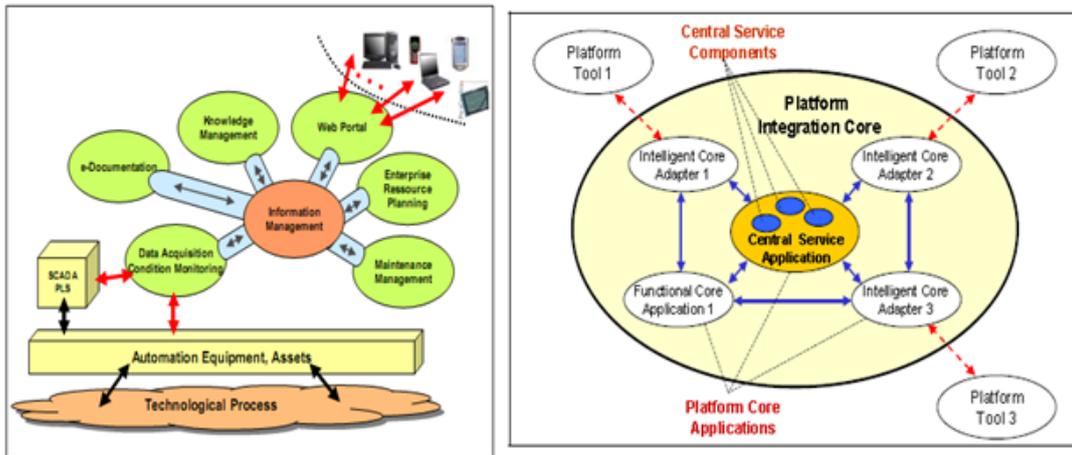


Figure 2-3 Architecture d'intégration de PROTEUS (Bangemann, et al., 2006).

#### Interopérabilité :

PROTEUS traite du problème d'interopérabilité au niveau syntaxique via les services web ce qui permet un échange facile des données sans le partage des connaissances entre ses services.

#### Gestion des connaissances :

Concernant l'ingénierie et le management des connaissances, PROTEUS se base sur un modèle de connaissances de l'équipement. Ce dernier est exploité par le service de diagnostic basé sur le raisonnement à partir de cas, tout en contenant une mémoire d'entreprise. Toutefois, il n'y a pas d'ontologie du domaine de maintenance dans la plateforme.

Ainsi, PROTEUS a prévu l'évolution de la plateforme via l'intégration de nouveaux services qui ne sont pas dynamiques.

#### Mécanisme de synchronisation :

La nature « boîte noire » des services élimine la possibilité de partage des ressources internes entre ces services. De plus, comme tout système orienté service, la plateforme PROTEUS utilise un modèle d'orchestration entre les services d'où l'aspect de coopération.

### 2.2.4- PROMISE

Le projet européen PROMISE (*P*ROduct lifecycle Management and *I*nformation tracking using *S*mart Embedded systems) a pour objectif la gestion des flux d'informations au-delà du client, de clôturer la boucle d'information de gestion du cycle de vie du produit (*PLC*, *Product Life Cycle*) et de transformer les données brutes du PLC à des connaissances exploitables (Kiritsis, 2004).

Une architecture de plateforme a été proposée à l'issue de ce projet comme le montre la Figure 2-4. Cette architecture utilise un modèle peer-to-peer pour l'échange d'informations, où tout dispositif implémentant l'interface de messagerie de PROMISE (PMI-PROMISE Messaging Interface) basée sur les services Web peut communiquer avec tout autre dispositif prise en charge par le PMI, peu importe la taille de l'appareil (Främling & Nyman, 2008). L'interface PMI est essentielle et permet une approche basée sur les web-services.

La plateforme utilise la PMI pour la transmission des informations codées en XML principalement destinées à la transformation automatique des systèmes d'information.

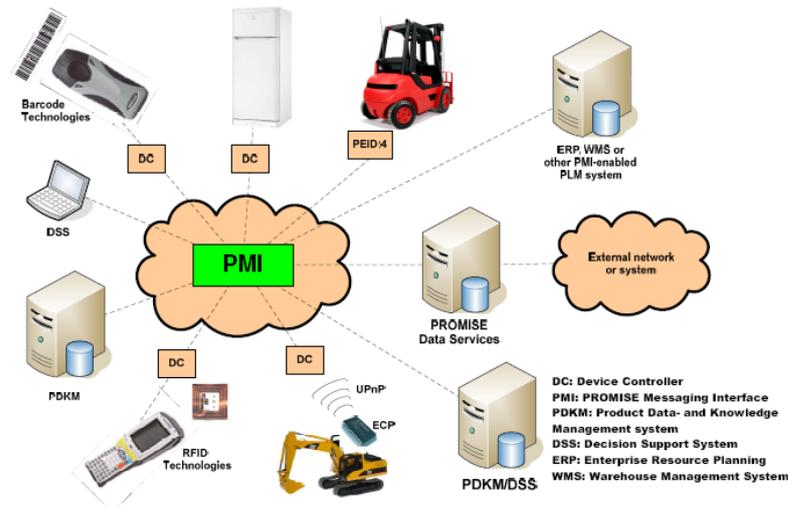


Figure 2-4 Architecture de la plateforme de PROMISE (Främling & Nyman, 2008)

#### Gestion des connaissances :

Dans le cadre de ce projet, un modèle ontologique appelé SOM (*Product Data and Knowledge Management Semantic Object Model*) représenté par un diagramme de classe UML fournit les concepts génériques du cycle de vie d'un produit ainsi que les relations entre ces concepts. L'objectif du SOM est d'avoir une sémantique commune entre les différents intervenants durant le cycle de vie d'un produit. Bien que l'architecture proposée intègre tout un système de gestion de connaissances (le PDKM, voir figure 2-4) et un système d'aide à la décision basé sur la connaissance enregistrée dans le PKDM. Le modèle ontologique n'est pas exploité par des méthodes d'ingénierie des connaissances dans l'architecture de plateforme proposée. . L'approche générale du PDKM est d'intégrer les données relatives aux produits de toutes les phases, qu'elles proviennent de bases de données ou de PEIDs (Product Embedded Information Device), et de permettre l'intégration des fonctions d'analyse en vue de soutenir les processus de décision dans toutes les phases du cycle de vie. En outre PDKM permet de gérer les connaissances sur les produits qui peuvent être facilement récupérées par les utilisateurs et présentées dans leurs contextes respectifs. Toutefois, le système de gestion de connaissances n'est exploité que par l'utilisateur mais non pas par les applications et services intégrés dans le système.

#### Type de maintenance :

. L'architecture de PROMISE a été conçue pour répondre aux besoins spécifiques du PLM (Product Life Cycle Management) et permette la récupération et la mise à jour d'informations sur les produits durant tout leurs cycle vie, et ne s'est pas intéressé au process complet de maintenance.

#### Interopérabilité :

L'architecture basée sur les services Web et le modèle SOM nous ont permis de constater que PROMISE propose une solution d'interopérabilité syntaxique et sémantique. Rappelons que l'avancée technologique proposée par PROMISE a introduit l'intégration de nouvelles technologies comme le RFID, et les réseaux informatiques, mais ne s'est pas intéressés à la dynamicité des services.

#### Mécanisme de synchronisation :

L'approche Peer-to-peer adoptée par PROMISE facilite la communication, l'orchestration et la collaboration entre les serveurs de données et les services web du PMI.

#### **2..2.5- SMMART**

Le projet européen SMMART (System for Mobile Maintenance and Accessible in Real-Time) (système de maintenance mobile et accessible en temps réel) vise à fournir les étiquettes des nouvelles technologies intelligentes. Ce système surveille l'utilisation et les données de la maintenance à travers le cycle de vie des pièces critiques.

Par ailleurs, ce système fournit aussi de nouveaux services comme les outils avancés de dépannage qui contiennent un contrôleur de configuration, un outil de planification des ressources et un service de traçabilité afin d'améliorer celle-ci (Kusper, 2007).

En outre, ce projet vise également à établir un référentiel normatif en termes d'organisation des procédures et d'outils impliquant le MRO (Maintenance Repair & Overhaul) des intervenants, les fabricants, les opérateurs, les organismes de régulation ainsi que les compagnies d'assurance (Zephir, 2007).

L'architecture proposée dans ce projet est basée sur la technologie RFID servant à la traçabilité des unités logistiques et dans le gestionnaire du moteur de configuration qui sert à la récupération des données de configurations des composants critiques (voir Figure 2-5).

#### Type de maintenance :

Le projet s'intéresse à une partie du processus de maintenance et particulièrement à la gestion des unités logistiques.

#### Interopérabilité :

L'architecture de ce projet est bâtie sur une base de données commune créée à partir d'un référentiel normatif. Ceci assure une interopérabilité syntaxique entre les intervenants du système.

#### Gestion des connaissances :

Dans ce projet l'exploitation des connaissances est ignorée, sachant qu'elle vise l'intégration des services de traçabilité.

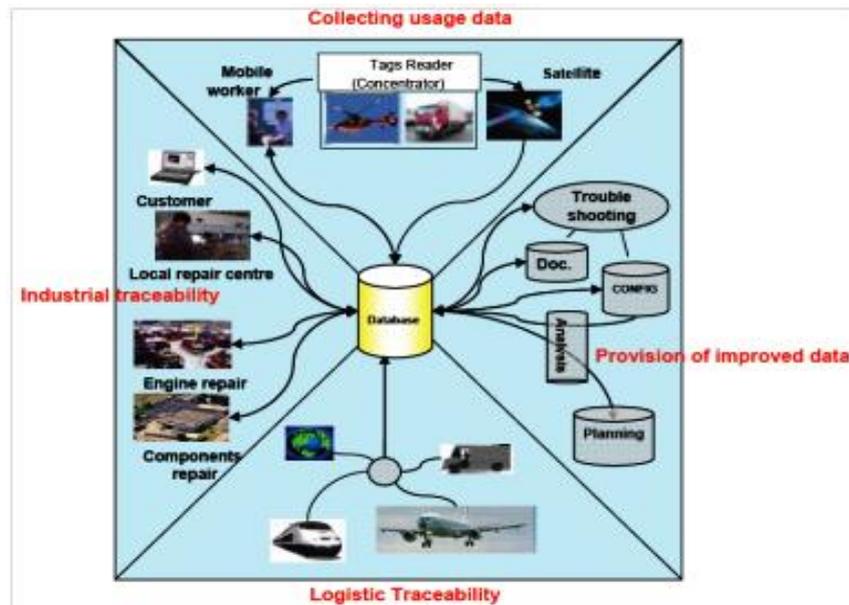


Figure 2-5 Architecture de SMMART (Zephir, 2007)

#### Mécanisme de synchronisation :

Les mécanismes de synchronisation entre les intervenants du système n'étant pas détaillés dans ce projet, nous ne pouvons identifier ce mécanisme à un mécanisme de coopération ou de collaboration.

### 2.2.6- DYNAMITE

Dans la continuité des projets Européen, le projet DYNAMITE (Dynamic Decisions in Maintenance) a eu comme objectif de créer une infrastructure pour les technologies mobiles de surveillance et de créer de nouveaux appareils/ instruments intelligents. Ces réalisations permettront des avancées majeures dans la capacité des systèmes de décision intégrant capteurs et algorithmes (Holmberg, Helle, & Halme, 2005).

Selon les objectifs déclarés par DYNAMITE, ce projet est centré sur la partie hardware et l'intégration des technologies mobiles et de nouveaux types de capteurs dans une plateforme de maintenance. Les dispositifs principaux incluent de la télémétrie sans fil, la présence d'un historique local dans des balises actives, et une instrumentation en ligne.

Dans ce contexte, une plateforme de maintenance DynaWeb a été développée (Arnaiz, Iung, Jantunen, Levrat, & Gilabert, 2007) et se réfère à une architecture basée sur les services web et sur des logiciels de communication qui prennent appui sur des fonctionnalités avancées de maintenance en lien avec le diagnostic, le pronostic et le CBM (voir Figure 2-6). C'est une plateforme d'information et de communication permettant une interaction opérationnelle entre les différents acteurs dans le cadre d'un scénario applicatif de gestion de maintenance distribuée respectant le processus OSA-CBM (Figure 2-7).

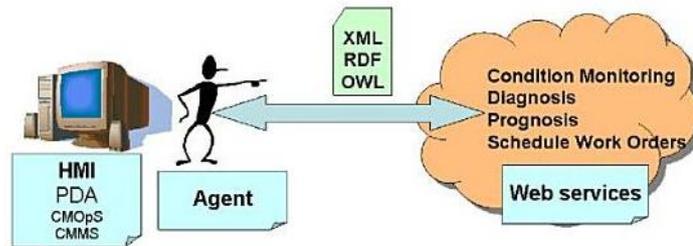


Figure 2-6 DynaWeb : une plateforme basée sur les services Web (Arnaiz, Iung, Jantunen, Levrat, & Gilabert, 2007)

L'architecture logicielle de DynaWeb est basée sur des composants logiciels offrant des services distribués sur le Web, ce qui est un point central. L'information est traitée dans un système distribué et de collaboration, où il ya différents niveaux d'entités qui peuvent accomplir des tâches intelligentes. Compte tenu de cela, une architecture de système a été définie en prenant appui sur la norme OSA CBM pour identifier les interactions entre les acteurs et les fonctions requises, notamment la surveillance des états, l'évaluation de la santé, le pronostic, et l'aide à la décision (Lebold & Thurston, 2001).

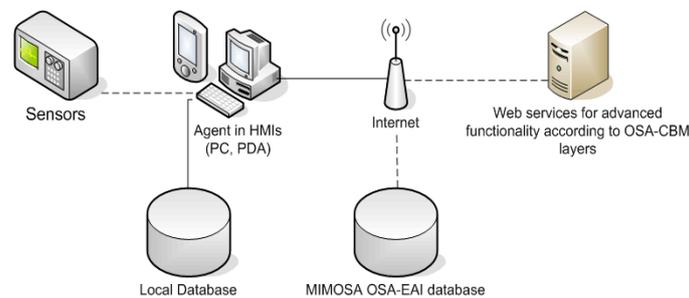


Figure 2-7 Architecture web de DynaWeb (Arnaiz, Iung, Jantunen, Levrat, & Gilabert, 2007)

Une plateforme dédiée à l'enseignement appelée TELMA a été définie comme plateforme de formation (et d'expérimentation) dans les domaines de la maintenance, télémaintenance et e-maintenance (Levrat & Iung, 2007).

Selon Iung, cette plateforme est en cohérence avec la philosophie de e-maintenance globale qui maintient un processus physique connectés à la fois à l'architecture d'automatisation et à l'architecture de maintenance en intégrant :

- L'ingénierie et le déploiement de CBM et les stratégies de maintenance proactives compatibles avec la proposition d'OSA/CBM (Lebold & Thurston, 2001);
- L'évaluation de ces stratégies sur la productivité du système industriel global (la disponibilité, l'exploitation), la Qualité, le Prix ...°

La plateforme TELMA (voir la Figure 8) est une plateforme dite « infotronic » assurant des services en ligne et hors ligne soutenant le traitement, le stockage et la communication entre les trois niveaux de données, d'information et de connaissances. Elle contient des agents intelligents assurant les services en ligne. Les services hors ligne sont généralement les services d'aide à la décision, qui contrairement aux nouveaux besoins ne sont pas des services à la demande et ne sont pas dynamiques.

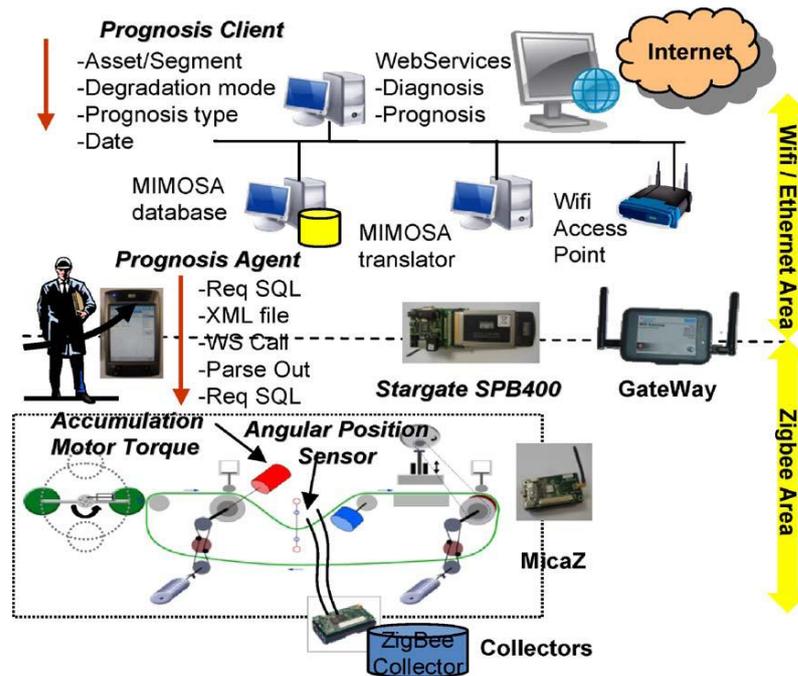


Figure 2-8 Architecture de TELMA (Levrat &amp; Iung, 2007)

Interopérabilité :

L'architecture logicielle offre une interopérabilité entre les applications logicielles indépendantes sur Internet grâce au protocole SOAP<sup>19</sup> assurant la communication entre ces applications. L'échange de données information étant un des points majeurs de DYNAMITE, ce projet a traité le point d'interopérabilité syntaxique via les services web, une interopérabilité technique sans se préoccuper du niveau sémantique. DYNAMITE a adopté le modèle CRIS MIMOSA comme base de données commune et partagée et non pas une ontologie commune.

Type de maintenance :

D'autre part, le projet s'est focalisé sur le processus OSA-CBM, notamment la maintenance proactive mais n'ont pas développés les autres stratégies de maintenance, ce qui limite la gestion de la maintenance dans la plateforme.

Gestion des connaissances :

L'architecture de la plateforme contient une base de connaissances, et des agents intelligents de TELMA assurant le passage de niveau entre données et connaissances, ainsi que des services hors ligne d'aide à la décision. La base de connaissance est développée pour alimenter le niveau « health assesment » et « prognostics » du processus OSA-CBM dans la plateforme, ce qui permet de définir des services intelligents. Toutefois, la structure de cette plateforme ne permet pas d'être auto-apprenante et de proposer des services dynamiques à la demande.

Mécanismes de synchronisation :

Les différents modules logiciels de DynaWeb communiquent entre eux afin d'accomplir une tâche spécifique tout en partageant la base de données MIMOSA. La plateforme utilise donc un mécanisme de collaboration.

<sup>19</sup> Simple Object Access Protocol : <http://www.w3.org/TR/soap/>

### 2.2.7- eMM

Le projet initié dans le cadre d'un programme de recherche (NFFP4), et supporté par « Saab Aerotech », l'Agence Nationale Suédoise de l'aéronautique et l'Agence suédoise de la Force aérienne « Wing F21 », un environnement logiciel de travail pour la gestion de la e-maintenance (*eMaintenance Management Framework [eMM]*) basée sur une architecture orientée service a été proposée (Candell, Karim, & Söderholm, 2009). Cet environnement logiciel comportera:

1. un modèle de gestion de eMaintenance [*eMaintenance Management Model (eMMM)*]
2. une plateforme de e-maintenance [*eMaintenance Platform (eMP)*]. (voir Figure 2-9).

L'eMMM est l'ensemble de rôles, de processus et d'entrepôt de données nécessaires pour la gestion de la plateforme. Cette dernière a une Architecture Orientée Service(AOS) (Bell, 2008) visant à fournir à ses utilisateurs des informations personnalisées pour la prise de décisions. Le but de cette plateforme est d'extraire toutes les informations nécessaires à une certaine tâche de maintenance, dans l'objectif d'analyser, de synthétiser ainsi que de les compacter dans un processus de maintenance dédié à l'intervention.

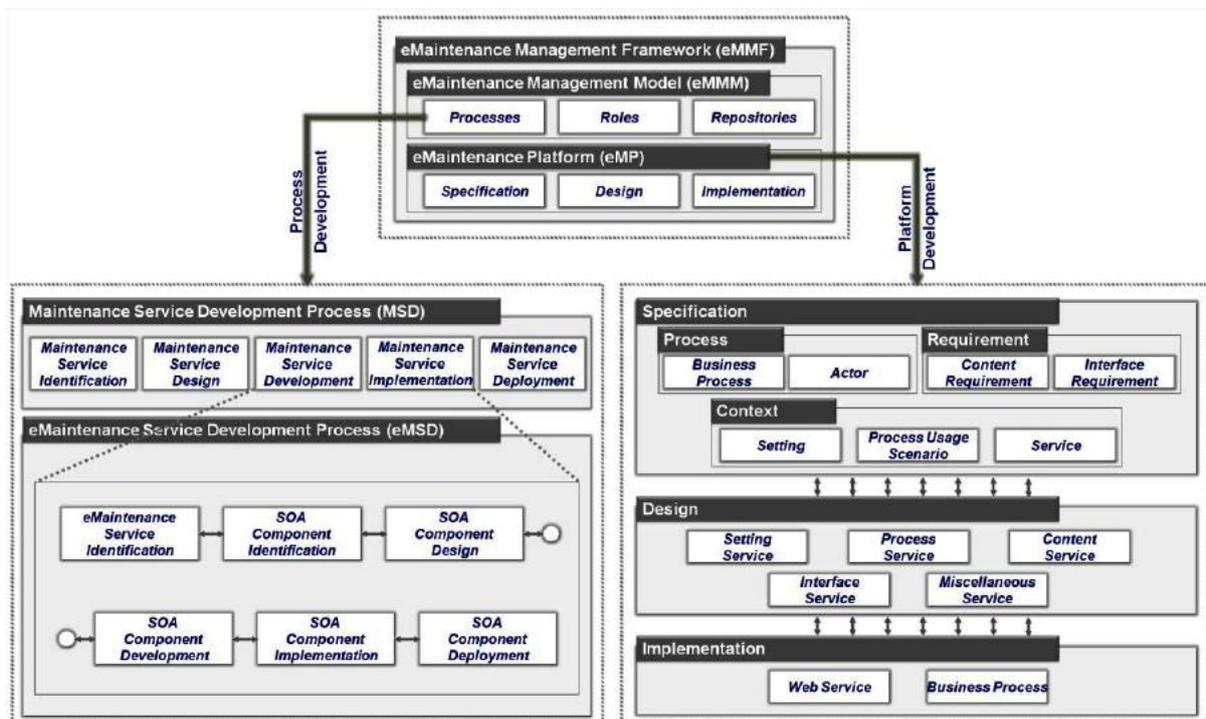


Figure 2-9 Architecture de eMMF (Karim, Candell, & Söderholm, 2009)

L'architecture eMP est divisée en trois niveaux: la spécification, la conception et l'implémentation (voir Figure 2-9). Le niveau de spécification contient toutes les informations nécessaires à la réalisation de la maintenance. Il est réalisé de différentes façons en utilisant tout type de données (notations, modèles, etc.). Le niveau de conception identifie les composants de conception. Il est réalisé en tant que composants AOS, en transformant la plupart des composants de spécification en composants AOS. Enfin, le niveau de mise en œuvre matérialise les composants AOS dans des services Web et dans les processus métiers orchestrés pour les besoins. Dans le cadre

de l'ePM, tout technicien de maintenance doit fournir des entrées utiles à l'application. L'application délivre alors un processus métier pertinents adapté à ses besoins.

Ceci montre qu'il n'y a pas de connaissances communes partagées entre ces techniciens et chacun d'eux adapte la plateforme à ses besoins spécifiques ce qui élimine toute possibilité de réutilisation des connaissances ou même d'exploitation, sachant que la sémantique des concepts et des visions peut changer d'un technicien à un autre.

#### Interopérabilité :

Il est à noter que l'AOS est une excellente approche pour faciliter la maintenance industrielle. Elle met l'accent sur la conception d'un point de vue processus métier, et est l'une des principales conditions préalables pour parvenir à une approche flexible, extensible et rentable pour la mise en place de la plateforme eMP.

Toutefois cette approche d'intégration, est similaire aux plateformes des projets PROTEUS et DYNAMITE basées sur les services Web qui est une dérivée de l'AOS. L'approche AOS utilisée dans l'eMP assure par conséquent une interopérabilité syntaxique entre les différents services déployés par la plateforme.

#### Gestion des connaissances :

Nous constatons que l'ePM est applicable par toute organisation assurant la maintenance industrielle, car elle est basée sur un processus de maintenance industrielle générique et complet. Par contre, la gestion et l'exploitation des connaissances ne sont pas mentionnées dans ce projet. ,

#### Type de maintenance :

L'objectif de cette plateforme est de faciliter les tâches d'interventions par les techniciens, et ne s'intéresse pas à l'exploitation des services par les autres acteurs impliqués dans le processus de maintenance.

#### Mécanismes de synchronisation :

Finalement, l'objectif sous jacent à l'architecture orientée service dans eMM est l'intégration des services dans un seul système permettant l'accès à l'information de façon unifiée. Toutefois, l'aspect synchronisation entre ces services n'a pas été détaillé.

## **2.3- Plateformes académiques**

Dans cette section nous ferons l'étude de six plateformes académiques instanciées dans le cadre universitaire ou de centre de recherche. La plupart de ces plateformes restent dans une sphère de proposition théorique sans réalisation concrète à l'exception de IMS-WATCHDOG dans D2B<sup>TM</sup>. Les plateformes que nous étudierons sont : WSDF, D2BTM, Plateforme Web, Plateforme Orientée Agents, EMASST et IIMED.

### **2.3.1- WSDF**

Hung et al en (Hung, Chen, Ho, & Cheng, 2003) ont proposés *un cadre de e-diagnostics basé sur des services Web (WSDF)* visant à améliorer les systèmes de diagnostic à distance en fournissant automatiquement le mécanisme d'intégration des informations de diagnostic par l'Internet. La plateforme WSDF a comme objectif

soutenir les tâches de e-diagnostics dans les usines de semi-conducteurs. Les technologies utilisées dans cette plateforme (voir Figure 2-10) sont les Services Web, XML Signature, chiffrement XML, https, SOAP, ISDN...

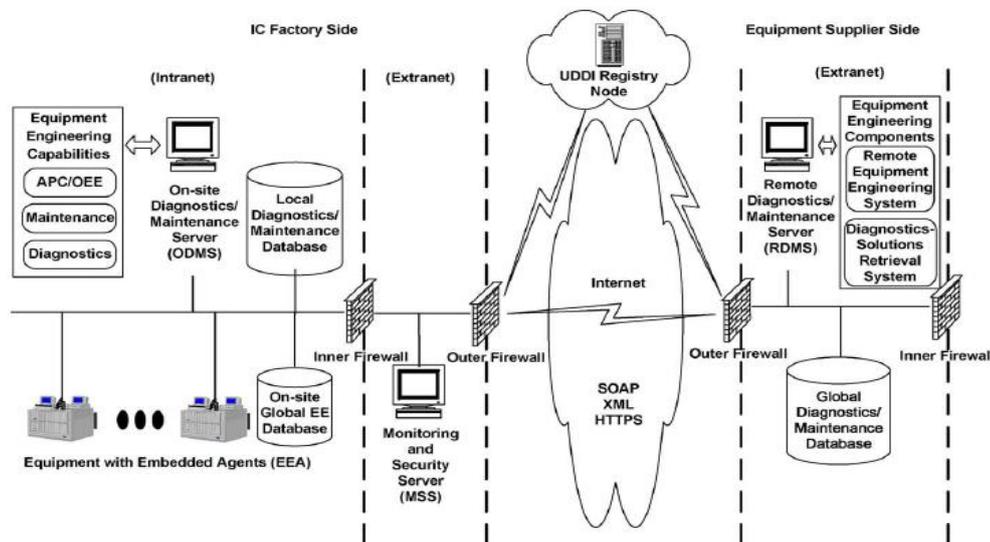


Figure 2-10 Architecture WSDF (Hung, Chen, Ho, & Cheng, 2003)

#### Interopérabilité :

L'architecture technologique basée sur les services web de cette plateforme, grâce aux échanges XML et aux services Web prend en compte l'interopérabilité syntaxique mais pas son niveau sémantique.

#### Gestion des connaissances :

D'autres parts, l'architecture proposée ne se base et n'exploite dans aucune partie ou composants l'ingénierie des connaissances.

#### Type de maintenance :

Cette plateforme traite essentiellement la partie diagnostic du processus de maintenance, la communication et la sécurité.

#### Mécanismes de synchronisation :

La synchronisation entre les acteurs de cette plateforme, n'étant pas détaillée nous ne pouvons pas identifier si ces mécanismes sont orientés collaboration,...

### **2.3.2- D2B™**

L'un des premiers projets académiques en e-maintenance a été élaboré au centre de systèmes intelligents de maintenance (IMS) aux Etats-Unis. Ce projet supporte le déploiement et l'expérimentation de la plateforme « dispositif-to-business » (D2B™) prenant appui sur un élément de base qui est le Watchdog Agent™ (Lee & Ni, 2004). L'objectif de cette plate-forme est de transformer les données équipement à des données au format compatible Web (par exemple XML) de sorte que de nombreuses applications Web peuvent être exécutées (Ali, Chen, Lee, & Koc, 2002). Via ce dispositif d'information les utilisateurs de différentes parties du réseau de

collaboration distribuée géographiquement peuvent partager la même information pour différentes applications synchronisées (Koc, Ni, Lee, & P, 2003).

Le « Watchdog agent » développe des outils et des algorithmes de pronostics innovants, des technologies de maintenance prédictives locales et à distance pour prédire et prévenir les défaillances des machines (voir **Erreur ! Source du renvoi introuvable.**).

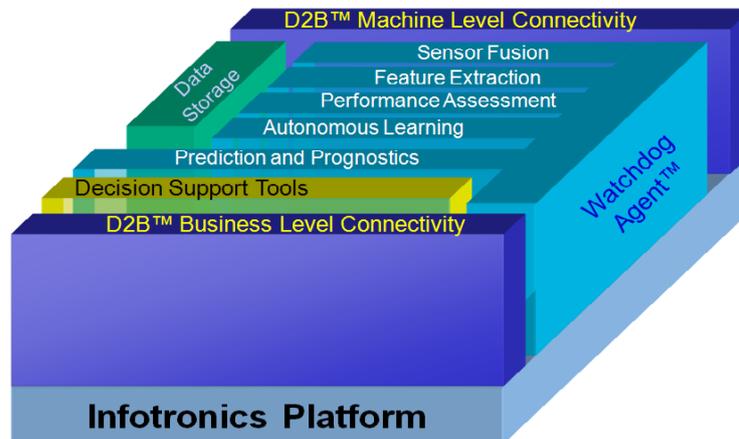


Figure 2-11 Integrated *infotonics* platform (Lee & Ni, 2004)

#### Type de maintenance :

Dans la continuité de ce projet, une plateforme d'*infotonics* intégrant différents composants intelligents a été proposée par IMS center. Cette plateforme ne s'intéresse qu'à la maintenance prédictive, et plus précisément au pronostic qui est la stratégie de maintenance la plus évoluée, mais néglige les autres stratégies. Par contre cette plateforme assure quelques services avant-gardistes comme le *self-learning*.

#### Interopérabilité :

L'interopérabilité est assurée au niveau syntaxique par un échange standardisé sous un format XML, et ne traite pas le problème d'interopérabilité sémantique.

#### Gestion des connaissances :

En ce qui concerne la gestion des connaissances, la plateforme contient des agents de connaissance permettant d'extraire des connaissances à partir de la base de données ainsi que de les acquérir à partir de connaissances pour aider à la décision.

#### Mécanismes de synchronisation :

Par contre, les notions de synchronisation et d'intégration ne sont pas décrites dans ce travail, mais d'après la description de l'architecture du système, on peut déduire l'existence d'un processus de coopération entre ces différents modules.

### 2.3.3- Plateforme web

Han et al (Han & Yang, 2006) ont proposé un nouveau système d'e-maintenance dépendant de la coordination, la coopération et de la négociation en utilisant internet, les technologies de la communication *tether-free* (web,

télécommunication sans fil). Ce système permet de réaliser des opérations de production pour atteindre la performance zéro temps d'arrêt sur une plateforme distribuée grâce à l'intégration des technologies avancées de réseaux. La structure du système de e-maintenance, contrairement aux systèmes classiques, comprend un centre de maintenance (fournisseurs) et une maintenance locale (clients) ce qui réduit efficacement les coûts de maintenance, le temps de conception de cette maintenance ce qui résout le problème du manque d'experts.

Le centre de e-maintenance contient en fonction de l'application un système de maintenance de base incluant la surveillance temps réel, le diagnostic de panne, la prédiction de dégradation et la gestion de la stratégie de maintenance. D'autre part, la maintenance locale se compose du service maintenance interne à l'entreprise ayant le rôle d'évaluation continue de la santé des équipements, de la gestion du processus d'intervention, de réparation et d'entretien ainsi que la compréhension et présentation des données (voir Figure 2-12).

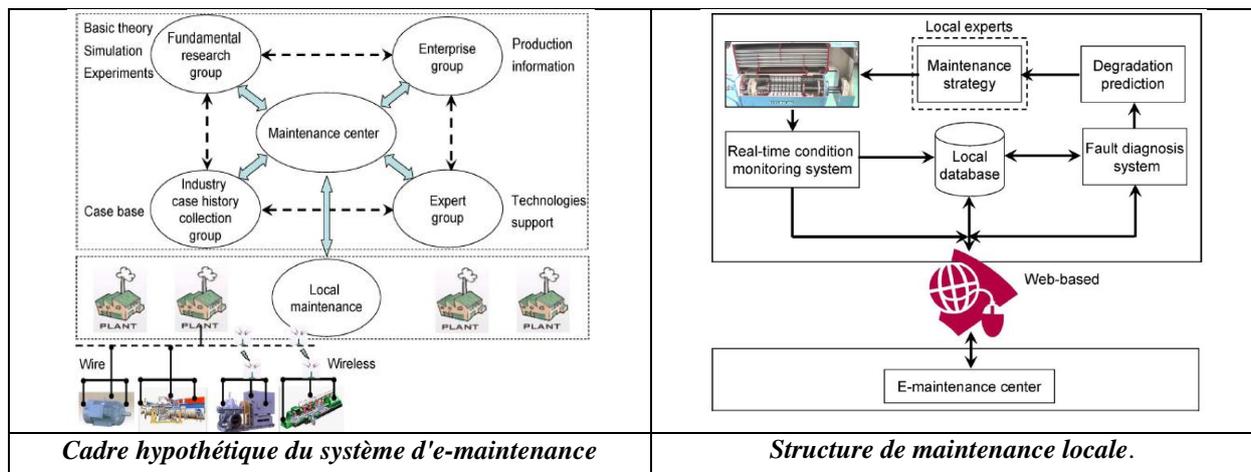


Figure 2-12 Architecture d'une plateforme web (Han & Yang, 2006)

#### Type de maintenance :

L'avantage du processus de maintenance est que les experts peuvent définir la stratégie ainsi que le processus de maintenance qu'ils jugent le mieux adapté pour chaque situation, le système ne s'autogère pas.

#### Interopérabilité :

Les échanges d'information effectués sur le web entre le centre d'e-maintenance et le centre de maintenance locale sont basés sur des messages XML standardisés. Ceci n'assure qu'une interopérabilité syntaxique pour ces échanges.

#### Gestion des connaissances :

L'ingénierie des connaissances n'est pas utilisée dans cette plateforme. Ceci est dû au fait que toute constatation et toute opération (présentation, compréhension et analyse) sur les données sont faites par les experts (humain) de maintenance locale.

#### Mécanismes de synchronisation :

Le partage des connaissances entre les acteurs sur site local et à distance confirme la collaboration entre ces acteurs à des fins de maintenance.

### 2.3.4- Plateforme orientée Agents

Zhang a proposé un système multi agents (Wooldridge, 2002) de e-maintenance basé sur la gestion des connaissances (Zhang, 2007). Ce système multi-agents permet d'échanger des connaissances/informations entre des systèmes d'automatisation industrielle. La figure 2-13.a montre l'infrastructure à base d'agent dédiée à la gestion de connaissances dans ce système. Toutes les sources d'information dans les systèmes d'automation industriels sont analysées et modélisées par des agents. Les relations entre ces sources sont mises en évidence dans le contexte de la maintenance, notamment la maintenance proactive<sup>20</sup>.

L'architecture de la plateforme est composée principalement d'un AMC (Agent Management Core) et de cinq types d'agents à savoir l'agent de connaissance [Knowledge Agent (KA)], l'agent de configuration [CA (Configuration Agent)], l'agent de diagnostic [DA (Diagnosis Agent)], l'agent de récupérations des données [FA (Field Agent)] et l'agent de management [MA (Management Agent)].

Ainsi, comme le montre la figure 2-13.b chaque agent est composé de cinq modules à savoir le module de connaissances local (LKM) qui définit les méta-connaissances relatives à la connaissance du système d'automatisation qui est représentée par l'agent; le module de connaissance sociale (SKM) qui définit les associations communautaires de l'agent; le module de prise de décision DMM qui définit les algorithmes et les modes de prise de décision proactive, par exemple, la logique floue, réseaux neuronaux artificiels, ou de raisonnement à base de cas ; le module de communication (CoM) définit les mécanismes de communication et des protocoles; et enfin le module d'interface (ItfM) qui définit les spécifications pour les interfaces et les adaptateurs.

Type de maintenance :

Cette orientation connaissance adoptée par Zhang, argumentée et orientée vers la maintenance proactive. Selon lui, la maintenance proactive peut être défini comme l'association entre la maintenance prévisionnelle avec la gestion de la connaissance. Ceci permettra d'intégrer de la dualité entre le fonctionnement et les dysfonctionnements des systèmes à maintenir qui est en soit même l'un des services requis par la nouvelle génération des systèmes de maintenance.

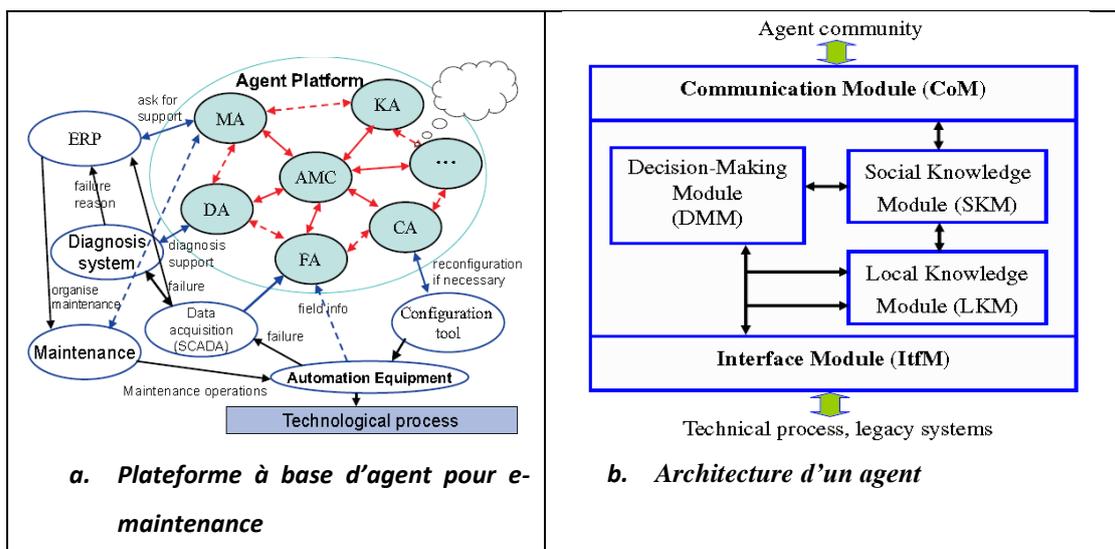


Figure 2-13 Architecture Plateforme orientée agents (Zhang, 2007)

<sup>20</sup> Maintenance proactive : optimisation des conditions d'utilisation dans le but d'augmenter la durée de vie des équipements.

Interopérabilité :

Le module de connaissance sociale de chaque agent permet d'assurer une interopérabilité sémantique entre ces agents en adaptant les connaissances locales avec celles de la communauté, sans pour autant partager toute la connaissance experte de maintenance et communique avec les autres agents via le module de communication qui échange avec le module de connaissance.

Gestion des connaissances :

Cette plateforme prend appui sur l'ingénierie des connaissances grâce à l'agent de connaissance qu'elle intègre ainsi que les modules internes de chaque agent qui sont orientés connaissances, mais ne propose pas de fonctionnalités auto-X, ni des services dynamiques ni à la demande et autre.

Mécanismes de synchronisation :

Selon cette architecture, malgré le partage des connaissances de l'environnement, chaque agent a ses propres ressources qu'il ne partage pas avec les autres agents et limite la communication à la coopération et non pas à la collaboration.

### 2.3.5- EMAST (E-Maintenance Architecture to Support on-site Teams)

Ribiero et al proposent une architecture de e-maintenance pour supporter les équipes de maintenance sur le site de production (Ribeiro, Barata, & Silvério, 2008).

L'architecture EMAST propose une solution verticale allant du diagnostic et/ou pronostic de bas niveau à la création de séances de maintenance collaborative. Chaque module dans cette architecture effectue du *self-monitoring/diagnosis* (auto-surveillance et autodiagnostic) d'où sa capacité à utiliser ces informations pour émettre des alarmes de maintenance prédictive, en plus des alarmes basées sur le temps de la maintenance préventive. Ainsi l'ensemble de documentation (manuels techniques, plans, procédures de réparation / maintenance, etc) est stocké localement sur le site.

Le module de configuration de l'équipe de maintenance « *The Maintenance Team Configurator (MTC)* » est l'un des composants de base de l'architecture. Il gère le groupe de techniciens de maintenance disponibles, leurs compétences et l'instanciation des tâches de maintenance et/ou de réparation (Figure 2-14).

Toutes les opérations de maintenance sont stockées dans une base historique et caractérisées par des variables qui sont : la nature de l'opération (critiques, non programmée ...), le taux de réussite de l'opération (mesure de la performance des techniciens et les orientations fournies par la plateforme) et les techniciens impliqués (chaque technicien dispose d'un score qui lui accorde du crédit pour effectuer des opérations plus complexes).

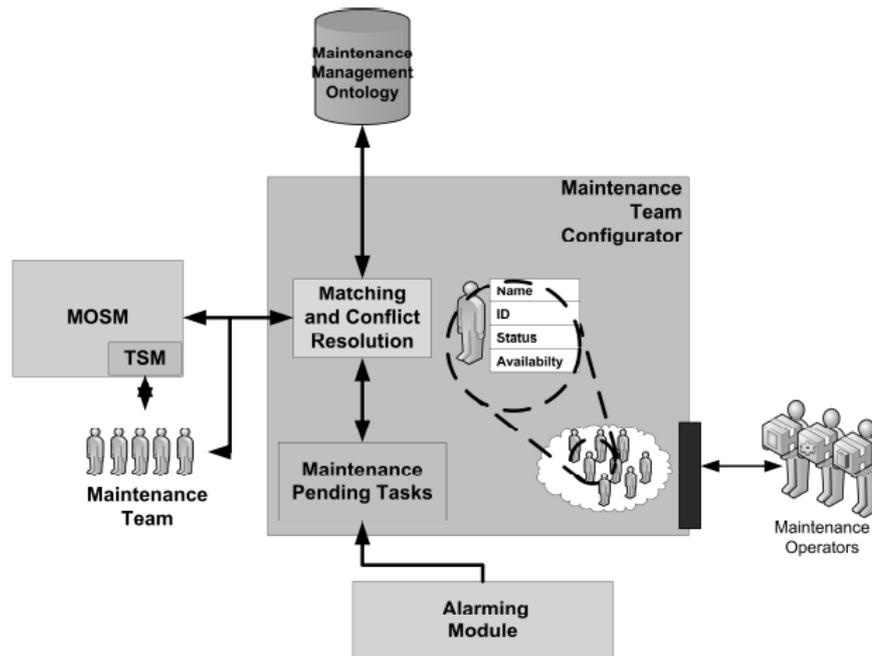


Figure 2-14 Architecture EMAS (Ribeiro, Barata, & Silvério, 2008)

Quand une alarme de réparation est reçue, le MTC propose une équipe de maintenance en se basant sur l'historique de maintenance des alarmes antérieures. Le MTC vérifie ces informations par rapport à la connaissance dans le gestionnaire d'ontologie de Maintenance « Maintenance Management Ontology (MMO) » et la disponibilité des techniciens. Les conflits de correspondance lors de cette vérification sont résolus par les règles du MMO où les règles de composition de l'équipe sont mappées.

Le processus de formation d'une équipe de maintenance comprend (1) la définition du critère d'urgence par tâches de maintenance, (2) l'attribution des rôles aux participants, (3) l'information sur la nature et l'emplacement de la tâche, (4) l'affectation du contrôle au module de support de maintenance opérationnelle « *Maintenance Operational Support Module (MOSM)* ». Grâce à une interface de soutien technique (TSM), le MOSM guide les équipes de maintenance en cours des opérations.

Le MTC gère, planifie et ordonnance la totalité des actions de maintenance et des équipes mais il peut aussi éventuellement différer l'exécution de la maintenance en reconnaissant que la meilleure équipe sera disponible dans un intervalle de temps raisonnable.

#### Interopérabilité :

Aucune information ne mentionne le caractère centralisé ou distribué de l'architecture de EMAS et sa possibilité à intégrer des modules externes. On ne peut pas savoir comment est traité le problème d'interopérabilité.

#### Type de maintenance :

D'autre part, le travail traite essentiellement le problème d'affectation des équipes de maintenance par rapport aux alarmes reçus, la planification des tâches de maintenance et le stockage de l'historique des interventions.

#### Gestion des connaissances :

Bien que cette architecture mentionne la traçabilité et l'exploitation des opérations de maintenance précédemment faites, il n'y a aucune explication sur l'exploitation des connaissances et sur leur formalisation et l'objectif de celle-ci.

#### Mécanismes de synchronisation :

L'une des caractéristiques majeures de cette plateforme est la collaboration entre les différents acteurs par le partage des ressources qui est la base de données historique ainsi que l'ensemble des documentations.

### **2.3.6- IIEMD**

Cao et al (Cao & Jiang, 2008) développent une plate-forme de maintenance intelligente coopérative et distribuée IIEMD (*Integrated Intelligent Equipment Maintenance Decision*) qui propose un ensemble d'outils d'aide à la décision pour différentes activités de maintenance. IIEMD est un système hybride, soutenu par une ontologie où deux types de raisonnement cohabitent le RBR (*Rules Based Reasoning*) et le CBR (*Case Base Reasoning*).

L'architecture de la plateforme IIEMD comme le montre la Figure 15 est une AOS (Architecture Orientée de Service [SOA en anglais]). Cette architecture basée sur la technologie Web, est modulaire, offrant la possibilité d'une intégration pervasive pour plus d'agilité en temps réel. L'architecture d'implémentation de cette AOS est une architecture à base de composants.

Cette plateforme peut réaliser la présentation et la synthèse des données d'une manière compréhensive, l'évaluation continue de la santé des équipements, la prise de décisions sur les opérations de maintenance et de réparation, ainsi que l'intégration d'autres systèmes.

Dans cette architecture, les outils de décision de maintenance (tous sont définis comme des services Web) peuvent enregistrer, récupérer l'information à partir d'autres systèmes, distribuer les informations aux autres systèmes intégrés dans la plateforme. Ainsi, elle fournit un processus de décision automatisé, transparent, continu de façon unifiée, indépendamment de son origine, du fabricant, des intégrateurs et des utilisateurs finaux de l'équipement grâce des méthodes de raisonnements à base de cas et à base de règles associées à des règles d'adaptation.

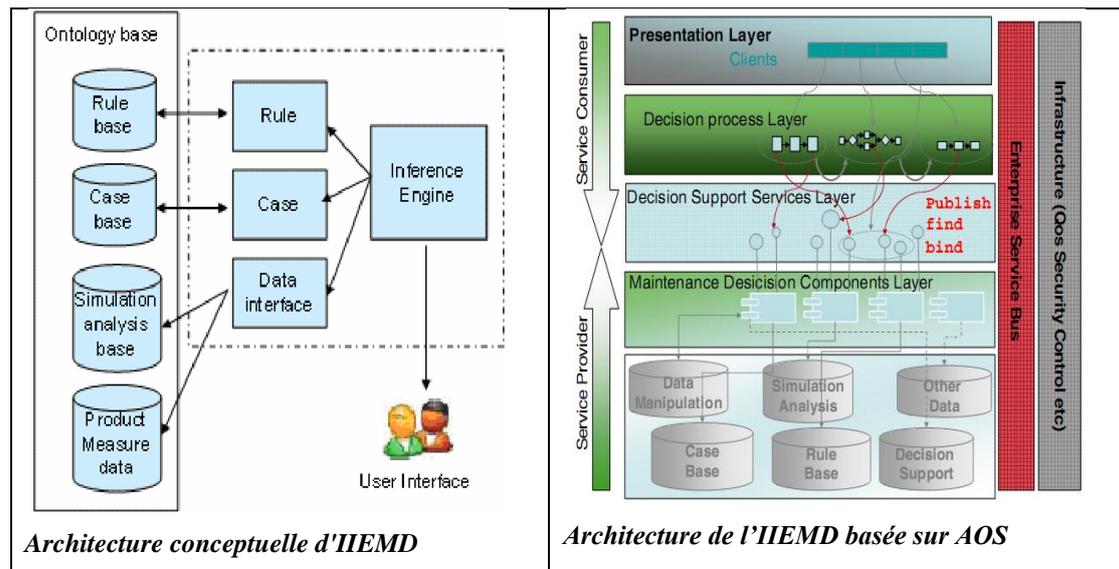


Figure 2-15 Architecture IIMED (Cao &amp; Jiang, 2008)

Cette plateforme comprend une base de connaissances prenant appui sur une ontologie. Cette dernière est formalisée à partir du modèle standard CRIS défini par MIMOSA (Lebold & Thurston, 2001) présentée sous forme de base de règle OWL<sup>21</sup> ainsi qu'un moteur d'inférence.

Le moteur d'inférence traite les requêtes des utilisateurs grâce au module de règle, le module de cas, le module d'interface de données, la base de connaissance contenant l'ontologie, le modèle des données, la base de l'analyse par simulation et les données de mesure.

#### Interopérabilité :

Bien qu'il ne soit pas mentionné, l'ontologie a été utilisée au profit de l'interopérabilité sémantique, car les services web utilisés dans l'AOS manipulent les concepts de l'ontologie.

#### Types de maintenance :

La plateforme fournit les services d'aide à la décision pour l'intervention et la réparation dans le processus de maintenance. Toutefois, la plateforme ne propose pas les services concernant toutes activités inclus dans le processus de maintenance (comme le diagnostic, le pronostic, l'optimisation des ressources, la planification) bien qu'elle peut les intégrer grâce à son AOS.

#### Gestion des connaissances :

La plateforme est un système hybride prenant appui sur l'ingénierie des connaissances par l'exploitation d'un moteur d'inférence qui exécute différents algorithmes de raisonnement (raisonnement à base de règles et raisonnement à base de cas) appliqué sur une base de connaissances contenant une ontologie faisant référence MIMOSA-CRIS.

#### Mécanismes de synchronisation :

<sup>21</sup> Web Ontology Language: <http://www.w3.org/TR/owl-ref/>

IIMED est une plateforme de coopération, en effet, ceci est confirmé grâce à l'architecture AOS qui assure un échange de ressources entre ses différents outils intégrés. De plus, l'absence de ressources partagées lui confère le statut de coopération.

## 2.4- Tableau de synthèse

Nous résumons dans le tableau 1 les caractéristiques des plateformes présentées dans ce chapitre selon les critères utilisés dans l'analyse de ces plateformes à savoir le processus de maintenance, l'interopérabilité, l'exploitation des connaissances, les services innovants fournis par ces plateformes, le degré de synchronisation et les objectifs des projets dans les quels les plateformes étaient construites.

Tableau 2-1 Tableau récapitulatif des plateformes de maintenance

<i>Plateforme</i>	<b>Processus de maintenance</b>	<b>Interopérabilité</b>	<b>Exploitation des connaissances</b>	<b>Services autonomes</b>	<b>Degré de synchronisation</b>	<b>Objectifs</b>
<i>MIMOSA</i>	CBM	Syntaxique (MIMOSA-CRIS)	--	--	Coopération	Intégration
<i>WSDF</i>	Diagnostic	Syntaxique (fichier XML)	--	--	--	Communication, sécurité
<i>PROTEUS</i>	Tout le processus de maintenance	Syntaxique (services web)	Mémoire d'entreprise pour le diagnostic.	--	Coopération	Intégration, interopérabilité
<i>SMMART</i>	Cycle de vie des pièces critiques	Base de données commune, Référentiel normatif.	--	--	--	Maintenance Mobile
<i>PROMISE</i>	BOL (début de vie),MOL (Milieu de vie), EOL(fin de vie)	SOM (Modèle sémantique d'objets)	--	Chien de garde pour le pronostic	Collaboration	Gestion du cycle de vie
<i>DYNAMITE</i>	OSA/CBM	Syntaxique (services web)	--	--	Collaboration	Création d'une infrastructure pour les technologies mobiles de surveillance
<i>TELMA</i>	OSA/CBM	Syntaxique (services web)	Base de connaissances pour le pronostic.	--	Collaboration	Enseignement, Télémaintenance.
<i>D2B™ / IMS WATCHDOG</i>	Maintenance prédictive	Syntaxique (fichier XML)	Agent de connaissances pour l'aide à la décision.	Self-learning	Coopération	Prédire et prévenir les défaillances des machines, Assurer un temps d'arrêt près de

						zéro.
<i>eMM</i>	Aide à l'intervention	Syntaxique (services web)	--	--	--	Intégration, de faciliter les tâches d'interventions
<i>Plateforme Web</i>	Tout le processus de maintenance	Syntaxique (messages XML)	--	--	Collaboration	Identification de la stratégie de maintenance adaptée pour chaque situation, assurer un temps d'arrêt près de zéro.
<i>Plateforme Orientée Agents</i>	Maintenance proactive	Sémantique (module de connaissances des agents)	Gestion et échange de connaissances	--	Coopération	Augmenter la durée de vie des équipements.
<i>POMAESS</i>	Diagnostic	--	Base de cas pour le diagnostic	--	Coopération	Aide à la décision pour le diagnostic.
<i>EMAST</i>	Maintenance prédictive et maintenance préventive	--	Maintenance Management Ontology (MMO) »	--	Collaboration	Supporter les équipes de maintenance sur le site de production.
<i>IIEMD</i>	Aide à la décision pour la réparation	Sémantique (Ontology OWL)	Ontologie, raisonnement et inférence : RBR (Rules Based Reasoning) et le CBR (Case BaseReasoning)	--	Coopération	Service d'aide à la décision de maintenance des équipements.

D'après cette analyse, nous avons constaté que PROTEUS est l'un des rares projets qui présentait une vision complète sur le processus de maintenance. Par contre, en ce qui concerne l'exploitation de la connaissance, malgré l'intégration de l'ingénierie des connaissances dans l'application de diagnostic. Cette connaissance n'était pas exploitée au cœur de la plateforme, (c.à.d. la plateforme n'est pas un système orienté connaissances) comme défini pour la s-maintenance.

La plateforme PROMISE est basée sur un modèle de données dit sémantique. En effet, l'objectif de ce modèle est de fournir une compréhension commune aux utilisateurs finaux de cette plateforme, n'est exploité par les

applications et services intégrés dans la plateforme, et par conséquent ne fournit pas de raisonnement (point fondamentale requis dans une plateforme de s-maintenance).

La plateforme TELMA fournit un ensemble de services d'aide à la décision en hors ligne, service en inadéquation avec les services dynamique en ligne et/ou services à la demande.

La plateforme D2B<sup>TM</sup> assure des services intelligents de *self-learning* dédié à l'application de pronostic, ce qui impacte la robustesse de la maintenance prévisionnelle et non pas sur l'auto-apprentissage de la plateforme et on retour d'expérience sur toutes les applications de maintenance. L'utilisateur ne pourra bénéficier de ce retour d'expérience dans ses services à la demande.

La Plateforme Orientée Agents possède quelques points de concordance avec la s-maintenance au niveau de l'interopérabilité sémantique et l'exploitation de l'ingénierie des connaissances. Néanmoins, cette exploitation de connaissances est faite localement par un agent et n'est partagées et réutilisées par toutes les applications d'aide à la maintenance intégrées dans la plateforme. De plus, les agents ne fournissent pas de services dynamiques.

La plateforme EMAST est basée sur une ontologie de domaine et exploite l'ingénierie des connaissances. Mais l'architecture de cette plateforme ne fournit aucuns détails ni sur l'utilisation de cette ontologie ni sur son exploitation par rapport à l'interopérabilité sémantique et par rapport aux services fournis aux utilisateurs.

La plateforme IIEMD est l'une des plus proches de la s-maintenance. Elle se basant sur l'ingénierie des connaissances (ontologie et moteur d'inférence) et en met l'accent sur l'aspect coopératif. Toutefois elle ne s'intéresse pas aux services dynamiques et autonomes. De plus, IIEMD ne traite que les services d'aides à la décision pour la réparation qui est une approche orientée utilisateur final (le technicien), mais elle ne développe pas sur la notion de génération de services à la demande pour les différents types d'utilisateurs de la plateforme (tous les acteurs de la maintenance).

De manière globale, le tableau de synthèse (tableau 2.1) nous permet de constater qu'il n'y a que trois plateformes sur quatorze (la plateforme de PROMISE, la plateforme Orientée Agents et la plateforme IIEMD) qui traitent de la problématique de l'interopérabilité sémantique, alors que les autres se limitent au niveau syntaxique de l'interopérabilité.

En ce qui concerne le processus complet de maintenance il n'y a que la plateforme PROTEUS et la Plateforme Web qui se sont intéressées à cela. Les plateformes eMM et IIEMD ont plus ou moins traité un ensemble d'activités de ce processus sans se focaliser sur une seule activité ou sur une seule stratégie de maintenance comme dans la plupart des autres plateformes (par exemple : MIMOSA, DYNAMITE, WSDF).

A propos de l'exploitation des connaissances, nous constatons que différentes plateformes tirent profits de l'ingénierie des connaissances et ontologique comme dans le cas de EMAST, PROTEUS et IIEMD. Par contre, cette exploitation est restreinte dans la plus part des cas à une partie de la maintenance, concernant le pronostic dans le cas de DEBTM ou l'aide à la réparation dans IIEMD, ou à une exploitation réduite par un nombre limité d'applications comme dans le cas de PROTEUS où la base de connaissance est un module extérieur au noyau de la plateforme.

Nous constatons aussi que la majorité de ces plateformes sont partagées entre la coopération et la collaboration. Les plateformes MIMOSA, PROTEUS, D2BTM, Plateforme Orientée Agents, POMAESS sont classées comme des plateformes de coopération. Par ailleurs, les plateformes classées comme plateformes de collaboration sont PROMISE, DYNAMITE, TELMA, Plateforme WEB, EMAST, IIEMD.

### **Synthèse :**

A l'issue de cette étude nous constatons qu'il y a une différence au niveau des caractéristiques et des objectifs de chacune des plateformes. Par ailleurs, la majorité de ces plateformes n'est pas en concordance avec les critères de la s-maintenance bien que quelques unes de ces caractéristiques soient prises en compte de manière partielle dans certaines plateformes.

Nous pouvons conclure aussi que les plateformes IIEMD et Plateforme Orientée Agents (POA) sont plus ou moins les plus proches de la plateforme de s-maintenance envisagée par rapport au partage et à l'exploitation des connaissances. Dans IIEMD la connaissance est gérée dans le cadre d'un système à base de connaissance contenant une base ontologique et un moteur d'inférence utilisé par des services d'aides à la décision dans la plateforme. En ce qui concerne POA, les connaissances sont partagées par des agents et chaque agent contient son propre module de décision qui manipule ces connaissances.

Dans une optique de réutilisation nous allons étudier les points clés de ces deux plateformes. Cette réutilisation pourra poser des problèmes sachant que les architectures des plateformes sont totalement différentes (SOA, multi agent).

En ce qui concerne la Plateforme Orientée Agents, son architecture se base sur la modélisation d'agent à partir des systèmes d'automation industriels existants (ERP, SCADA, etc). Le seul agent autonome dans cette plateforme qui ne fait pas référence à un système existant est l'agent de connaissances (KA-Knowledge Agent). Il est à noter que le type des agents intégrés dans POA n'est pas mentionné, nous ne savons pas si les agents sont proactifs, réactifs, cognitifs ou conversationnels animés. Cette plateforme nécessite ainsi l'intégration de nouveaux agents autonomes fournissant les fonctionnalités et les caractéristiques requises par la plateforme de s-maintenance comme des agents d'auto-apprentissage, d'autogestion, d'auto-traçabilité, etc.

L'architecture de IIEMD intègre des services concernant des activités d'aide à la réparation et l'intervention. Dans son état, cette architecture permet l'intégration de nouveaux services. Nous pouvons donc l'adopter et la faire évoluer pour qu'elle couvre le processus global de la maintenance et fournir les services requis par la s-maintenance. Ainsi, d'autres services spécifiques doivent être intégrés afin de fournir les fonctionnalités requises par la s-maintenance comme le self-X et les services à la demande.

Par ailleurs, il est à noter aussi que les deux plateformes sont conceptuelles et restent dans un cadre théorique. Les grandes lignes sont données et les détails de leurs fonctionnements sont omis, ce qui met en cause la reprise de l'une d'elles et renforce l'idée de proposer une nouvelle architecture. Toutefois, ceci n'élimine pas les possibilités d'inspiration concernant ces deux plateformes pour proposer une nouvelle architecture adaptée à la s-maintenance prenant appui sur les structures de leurs architectures.

### 3. Conclusion

Afin de définir une plateforme de s-maintenance, nous nous sommes attelés dans ce chapitre à répondre à la question suivante : « *Peut-on réutiliser une plateforme existante de maintenance pour une plateforme de s-maintenance ?* »

Pour répondre à cette question, nous avons analysé les plateformes de e-maintenance existantes suivant quatre principales caractéristiques liés à la s-maintenance.

L'analyse de ces plateformes nous a permis de constater que :

- sept plateformes ont exploré le volet « exploitation des connaissances, mais de façon locale dans chaque service pour la plupart mais n'exploitent pas les connaissances quant à la gestion de la plateforme et celle du processus globale de maintenance mais juste pour une activité particulière (comme le diagnostic dans PROTEUS).
- Sur quatorze plateformes, onze sont orientées vers un type ou une activité particulièr(e) de maintenance telle que OSA-CBM ou l'activité de réparation. Seulement trois plateformes traitent le processus global de maintenance.
- En ce qui concerne l'interopérabilité, trois plateformes se sont intéressées au volet sémantique sur neuf autres qui n'ont traitées que le volet syntaxique. Nous pouvons remarquer que le niveau sémantique de l'interopérabilité est négligé dans la plupart des plateformes de e-maintenance.
- En ce qui concerne le niveau de synchronisation, six plateformes sur quatorze ont un mode de coopération et cinq plateformes sont en collaboration. Ceci montre que neuf plateformes sur quatorze ne supportent pas le partage de ressources communes.

En effet, le nombre maximum de caractéristiques liées à la s-maintenance dans une plateforme de e-maintenance est deux. Ainsi, seulement huit plateformes sur les quatorze répondent à la fois à deux de ces caractéristiques. Ceci confirme à la fois les liens et les différences entre la e-maintenance et la s-maintenance.

Ces résultats nous ont permis de confirmer les différences qui existent entre une plateforme de e-maintenance et une de s-maintenance. Malgré les correspondances notées ci-dessus, nous pouvons conclure que dans la plupart des cas, les caractéristiques de la s-maintenance ne sont assurées que partiellement par les plateformes de e-maintenance existantes. En effet, la plateforme de e-maintenance et celle de s-maintenance sont différentes au niveau de l'objectif des services fournis par chaque type de plateforme ainsi que leur propriétés associées.

Toutefois, seulement deux plateformes (conceptuelles) sur les plateformes étudiées sont les plus proches de la plateforme de s-maintenance et sont IIMED et la Plateforme Orientée Agents (POA). En effet, bien que ces deux plateformes soient orientées vers la coopération, ce qui s'explique par l'absence de connaissances communes partagées entre leur composants logiciels (les services pour la première et les agents pour la deuxième), elles sont orientées connaissances par l'exploitation et la réutilisation des connaissances par leurs composants tout en assurant une interopérabilité sémantique.

Par conséquent, vu le gap entre les services fournis par la e-maintenance et ceux que doit offrir la s-maintenance, nous devons donc développer une architecture spécifique à une plateforme de s-maintenance partant des

caractéristiques et exigences de ce concept ainsi que les points forts de la e-maintenance. Toutefois, nous pourrions nous inspirer des aspects architecturaux et technologiques des deux plateformes IEMED et POA.

