

وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR-ANNABA UNIVERSITY  
UNIVERSITE BADJI MOKHTAR-ANNABA



جامعة باجي مختار-عنابة

Faculté : *Sciences d'ingénieur*

Année : 2003

Département : *Informatique*

## Mémoire

Présenté en vue de l'obtention du diplôme de MAGISTER

# Vers Un Système De Gestion Des Connaissances Pour L'aide Au Diagnostic De Pannes Dans Un Système Industriel

Option  
Intelligence Artificielle

Par  
*Anguel Fouzia*

Directeur De Mémoire : Mr M.Sellami MC, U. Annaba

DEVANT LE JURY

Président : M<sup>me</sup> M. Babes MC, U. Annaba

Examineurs : Mr M<sup>ed</sup>. Baatouche PR, U. Constantine  
M<sup>me</sup> Z. Boufaïda PR, U. Constantine  
M<sup>me</sup> H. Merouani MC, U. Annaba

## *DEDICACE*

*A mes Parents auxquels je dois tant et je ne rendrais jamais assez.*

*A mon époux*

*A ma famille*

*A mes amies.*

## *REMERCIEMENTS*

*Je tiens tout d'abord à remercier M<sup>r</sup> Sellami Mokhtar qui a su m'accueillir dans le domaine de la recherche en informatique et me le faire découvrir. Je lui suis très reconnaissante pour son encadrement et son soutien dans ce projet.*

*Je remercie également M<sup>me</sup> M.Babes d'avoir accepté de présider le jury .*

*J'adresse mes remerciements à M<sup>r</sup> M<sup>ed</sup>.Baatouche, M<sup>me</sup> Z.Boufaïda et M<sup>me</sup> H.Merouani d'avoir accepté de juger et critiquer ce mémoire.*

*Merci aussi à toutes les personnes qui ont contribué de près ou de loin pour la réalisation de ce travail.*

*Et enfin, merci à mes proches.*

-

...

.

....

-

.

## **Résumé**

La connaissance se trouve dans l'intelligence, le savoir-faire, la mémoire, l'expérience des personnes autant que dans les documents, base de données, etc. Il faut donc l'extraire pour espérer la conserver et pouvoir la transmettre en vue de sa réutilisation autrement dit il faut gérer ces connaissances. L'objectif de ce travail est de gérer les connaissances concernant le diagnostic de pannes dans une station de compression en développant un outil dont le but n'est pas de résoudre automatiquement le problème de diagnostic, mais d'aider l'utilisateur en lui fournissant les informations appropriées et en lui laissant la responsabilité d'une évaluation contextuelle de ces informations et ce en exploitant la mémoire construite à travers les différentes sources de connaissances.

## **Mots clés**

Gestion des connaissances, mémoire d'entreprise, ontologie, Système à base de connaissances, diagnostics de pannes, CommonKADS, protégé-2000, Jess.

## **Abstract**

The knowledge are to be found in the understanding, the know-how, the memory, data bases in the person's experience than documents, etc. Therefore, we have to extract it for preserving, transmitting and using it again . So, we have to manage these knowledge. The purpose of this Work is to manage the knowledge of the diagnosis of the troubleshootings in compression station . We have developed a tool that the main doesn't to resolve automatically the problem of the diagnosis, but it's to help the user to supply him the responsibility for a contextual evaluation for these information through the exploitation of the memory which is constructed with the different knowledge sources .

## **Keywords**

Knowledge management, Corporate memory, Ontology, KBS, Troubleshooting's diagnosis, CommonKADS, protégé 2000, Jess

## **Liste des Tableaux**

01	Comparaison des méthodes de capitalisation des connaissances.....	47
----	---	----

## Table des figures

1.1	Les connaissances dans l'entreprise .....	6
1.2	Distinction des connaissances à deux niveaux .....	11
1.3	Conception d'un SBC selon une approche knowledge level .....	12
1.4	La problématique de gestion des connaissances .....	18
1.5	Gestion d'une mémoire d'entreprise .....	21
2.1	Les modèles de CommonKADS .....	36
3.1	La structure des composants d'une turbine .....	50
3.2	Le principe de fonctionnement de la turbine .....	51
3.3	Concepts d'un système de diagnostic.....	53
3.4	Le modèle de diagnostic basé sur des heuristiques .....	55
3.5	Structure des inférences de la tâche de diagnostic systématique .....	56
4.1	Une vue de décomposition structurelle du système .....	62
4.2	Aperçu général du modèle de propagation de pannes .....	63
4.3	Architecture générale du système de gestion de connaissances .....	65



# Sommaire

ملخص

Résumé

Abstract

Table des figures

Liste des tableaux

## Introduction générale

1 Introduction générale .....	1
2 Objet du mémoire .....	2
3 Structure du mémoire.....	3

## Partie I – Etat de l’art & Problématique

### Chapitre 1 Les SBC & la gestion des Connaissances

1 Introduction.....	4
2 Définition des connaissances de l’entreprise .....	4
2.1 Donnée, information et connaissance .....	5
2.2 Les connaissances de l’entreprise .....	5
3 Les systèmes à base de connaissances (SBC).....	8
3.1 Les systèmes experts .....	8
3.2 Problèmes soulevés par les systèmes experts de première génération.....	9
3.3 Les systèmes experts de seconde génération .....	10
3.4 L’approche niveau connaissance ou “Knowledge level “.....	11
3.5 Développement des systèmes à base de connaissances .....	12
3.6 Acquisition des connaissances .....	13
3.7 Modélisation .....	13
3.8 Elicitation dirigée par les modèles .....	15
4 Ingénierie des connaissances .....	15
5 La gestion des connaissances « Knowledge Management ».....	15
5.1 Définition et objectifs.....	17
5.2 Les composantes de la gestion des connaissances .....	17
5.2.1 La capitalisation des connaissances .....	19
5.2.2 La mémoire d’entreprise .....	19
5.2.2.1 Caractéristiques d’une mémoire d’entreprise .....	21
5.2.2.2 Le cycle de vie d’une mémoire d’entreprise .....	25
5.2.2.3 Typologie d’une mémoire d’entreprise .....	29
5.2.3 L’ontologie .....	29
6 Conclusion .....	32

### Chapitre 2 Les méthodes de gestion des connaissances

1 Introduction .....	33
2 La méthodologie CommonKADS .....	34
3 La méthodologie MKSM .....	37
4 La méthodologie REX .....	41

5 La méthodologie Cygma .....	43
6 Comparaison des différentes méthodes de capitalisation des connaissances .....	46
7 Conclusion.....	48

### **Chapitre 3 Le problème de diagnostic de pannes**

1 Introduction .....	49
2 Présentation du système réel .....	49
2.1 Les turbines .....	48
2.2 Le compresseur .....	48
3 Le problème de diagnostic .....	47
4 Les concepts associés .....	50
5 Le diagnostic de pannes avec la méthode CommonKADS .....	52
6 Conclusion.....	55

## **Partie II – Conception & Réalisation**

### **Chapitre 4 Analyse & Conception**

1 Démarche adoptée .....	58
2 Acquisition des connaissances .....	59
3 Modélisation .....	60
4 Approche de modélisation .....	62
4.1 Le modèle de décomposition structurelle et fonctionnelle du système .....	62
4.2 Le modèle causal de propagation des pannes .....	63
4.3 Le modèle de tâches .....	64
5 Architecture du système .....	65

### **Chapitre 5 Environnement de développement & Application .....**

1 Introduction .....	67
2 Environnement de développement .....	67
2.1 Protégé 2000 .....	67
2.2 Jess .....	71
2.3 Jess-Tab .....	72
3 Réalisation .....	73
3.1 présentation de l'application .....	73
3.2 Description de l'ontologie .....	74
3.3 Description des instances de la base de connaissances .....	77
3.4 Exemples extraits de l'application .....	77

### **Conclusion & Perspectives**

1 Conclusion .....	85
2 Perspectives .....	86

<b>Références</b> .....	87
-------------------------	----

### **Annexe**

# Introduction Générale

## **1. Introduction générale**

L'avènement des nouvelles technologies de l'information et de la communication a des impacts insoupçonnés sur l'évolution des systèmes de contrôle et de surveillance de pannes notamment dans les grandes installations industrielles. Une politique de maintenance industrielle est un mélange harmonieux d'entretien préventif systématique et d'entretien correctif. La surveillance des installations industrielles s'intègre dans cette politique; elle doit assurer entre autres :

- La prévention des risques majeurs par l'arrêt des machines lorsque les conditions de sécurité pour l'homme et la machine ne sont plus satisfaites.
- La détection précoce des anomalies pour éviter au maximum les arrêts par un remplacement des pièces défectueuses si possible avant l'incident et au meilleur moment pour limiter les pertes de production.
- L'analyse après incident pour remédier aux défauts constatés et éviter à l'avenir de les retrouver ou définir les modifications nécessaires.

Il s'agit alors de considérer le diagnostic comme un processus permanent et dynamique se déroulant en parallèle avec la vie d'une installation . Ceci relève du domaine de la « gestion des connaissances » (Knowledge Management) orientée réutilisation, et dont l'objectif est de mettre le savoir-faire d'un individu et de ses connaissances au service de toute une communauté d'individus. Cependant, contrairement aux systèmes experts au lieu de viser une résolution automatique pour une tâche avec des capacités automatiques de raisonnement, une mémoire d'entreprise doit plutôt aider l'utilisateur en lui fournissant des informations appropriées de l'entreprise mais en lui laissant la responsabilité d'une évaluation contextuelle de ces informations. Selon le cas, la mémoire d'entreprise construite pourra intégrer ou non des connaissances formelles. Les motivations de la capitalisation des connaissances sont diverses :

- Eviter la perte de savoir-faire d'un spécialiste après sa retraite ou sa mutation .
- Exploiter l'expérience acquise des projets passés et conserver les leçons du passé afin d'éviter de reproduire certaines erreurs.

- Améliorer la circulation de l'information et la communication dans l'organisation .
- Améliorer l'apprentissage des employés de l'organisation.
- Intégrer les différents savoir-faire d'une organisation.

## **2. Objet du travail**

Le domaine de diagnostic et de maintenance de matériel dans les stations de compression nécessite un savoir-faire technique spécifique et une expertise. Cette dernière est acquise sur le terrain à travers les études et les expériences par quelques spécialistes. Cependant, ces spécialistes ne sont pas disponibles à tout moment et dans chaque station.

Recueillir, préserver et exploiter les connaissances de diagnostic de pannes dans ce domaine est donc un besoin important ressenti comme une nécessité ou au moins une aide importante dans l'organisation considérée qui en effet centralise une expertise et un retour d'expérience importants sur certains équipements. Ce besoin nous a conduit à fixer deux objectifs principaux :

- 1- Formaliser une démarche générale pour capitaliser les connaissances expertes de diagnostic et de maintenance afin de concevoir des aides au diagnostic exploitant ces connaissances expertes.
- 2- Développer un environnement de gestion de connaissances de diagnostic, servant de support à cette démarche et offrant les possibilités suivantes :
  - a)- Recueil et gestion de connaissances de diagnostic et de maintenance de matériels.
  - b)- Consultation et analyse de connaissances.
  - c)- Exploitation des connaissances capitalisées.

Cet environnement doit fournir une assistance automatique pour le diagnostic des équipements industriels.

### **3. Structure du mémoire**

Après cette introduction, dans le premier chapitre, nous présentons un état de l'art sur le domaine des systèmes à base de connaissances et la gestion des connaissances en définissant la notion de connaissance, les SBC, la gestion de connaissances et les différents composants de gestion de connaissances. Le second chapitre présente d'une manière concise un aperçu sur les méthodologies de gestion des connaissances. Dans le troisième chapitre, nous présentons tout d'abord le système réel étudié et le problème de diagnostic des pannes ainsi que différentes approches associées. Ensuite une des méthode de résolution du problème de diagnostic. Le quatrième chapitre représente l'analyse de l'activité de diagnostic de pannes dans la station, nous décrivons l'ensemble des modèles élaborés pour représenter formellement le domaine de l'étude. Enfin nous décrivons l'architecture générale du système proposé ainsi que les fonctionnalités du système.

Le dernier chapitre sera consacré aux grandes lignes de la phase d'implémentation du prototype du système, en présentant les outils exploités : Protégé2000, JESS, et JESSTAB et les étapes de réalisation de l'application. Nous terminons par une conclusion et quelques perspectives à court, moyen et long terme.

**Partie I**  
**Etat de l'art & Problématique**

## **1. Introduction**

Le système d'information (SI) a acquis dans la plupart des organisations une importance capitale. De nombreux outils de gestion du SI se sont développés pour l'exploiter au mieux. Cependant les organisations disposent toujours d'une masse d'informations et de connaissances qui ne sont pas exploitées efficacement : le savoir\_faire des experts réside dans leur tête, les documents sont le plus souvent sous format papier et ne sont pas répertoriés de façon à être rendus accessibles par les membres de l'organisation. Souvent des bases de données sont maintenant implantées dans beaucoup d'organisations, elles sont hétérogènes, réparties géographiquement et ont été mises en place pour les besoins de projets spécifiques. Ainsi, leur existence reste inconnue en dehors du contexte de leur création et les données qu'elles renferment ne peuvent pas être exploitées pour d'autres besoins.

Au delà de la gestion de l'information on parle aujourd'hui de gestion de connaissances. La distinction entre l'approche de gestion du SI et celle de gestion des connaissances se fonde essentiellement sur la différence qu'il peut y avoir entre les concepts d'information et de connaissance. De façon générale l'information est le support du transfert de connaissances entre deux individus et elle possède une représentation syntaxique. Par contre les connaissances sont propres à chaque individu qui fait correspondre à toute information une sémantique dans un contexte donné. Gérer les connaissances consiste donc à gérer les informations en prenant en compte leur contexte et leur sémantique. L'intégration cohérente des connaissances dispersées et disparates au sein d'une organisation est appelée mémoire corporative, mémoire d'entreprise ou mémoire organisationnelle. Un système de gestion et de capitalisation des connaissances vise à construire, exploiter et maintenir une mémoire d'entreprise afin de la rendre vivante et évolutive.

## **2. Définition des connaissances de l'entreprise**

Pour pouvoir gérer les connaissances, il faut savoir les identifier, surtout au sein de l'entreprise. Ceci nous amène à porter un éclairage sur les concepts de donnée, d'information, de connaissance, de savoir, de savoir-faire, etc



## 2.1. Donnée, information et connaissance

Une donnée peut être définie comme toute représentation à laquelle une signification peut être attachée [16], c'est un fait discret et objectif résultant d'une acquisition, d'une mesure effectuée par un instrument ou construite par l'homme. Elle peut être qualitative ou quantitative. La donnée n'a pas de sens en elle-même, c'est seulement en associant un sens à cette donnée que l'individu possède une information. L'information est donc une association significative de données organisées pour donner forme à un message, elle permet de savoir un fait, un objet, un événement, un phénomène [66].

La connaissance<sup>(1)</sup> est un ensemble composite d'informations mais aussi de comportements de gestes, de mots, de sons .... qui est un savoir théorique ou un savoir faire (savoir expérimental), voir un savoir être davantage comportemental. La connaissance implique forcément l'homme « porteur ». La connaissance se définit comme suit : « La connaissance s'acquiert par l'accumulation et l'organisation d'informations dans la tête de chaque individu, elle va ensuite se structurer, se codifier et se transformer selon le contexte et les besoins dans lesquels elles évoluent » [66]. La notion de connaissance peut être confondue avec celle de compétence, la différence est que la compétence se crée à partir de l'évolution des connaissances.

En résumé, les données permettent d'accéder à l'information et à la connaissance, l'information permet de savoir un fait, un événement ou un phénomène ; tandis que la connaissance permet de comprendre (donc d'évaluer de comparer, de juger, etc.) et de reproduire, voire soutenir un processus d'apprentissage et d'intervention sur le terrain, ou le pilotage d'une organisation.

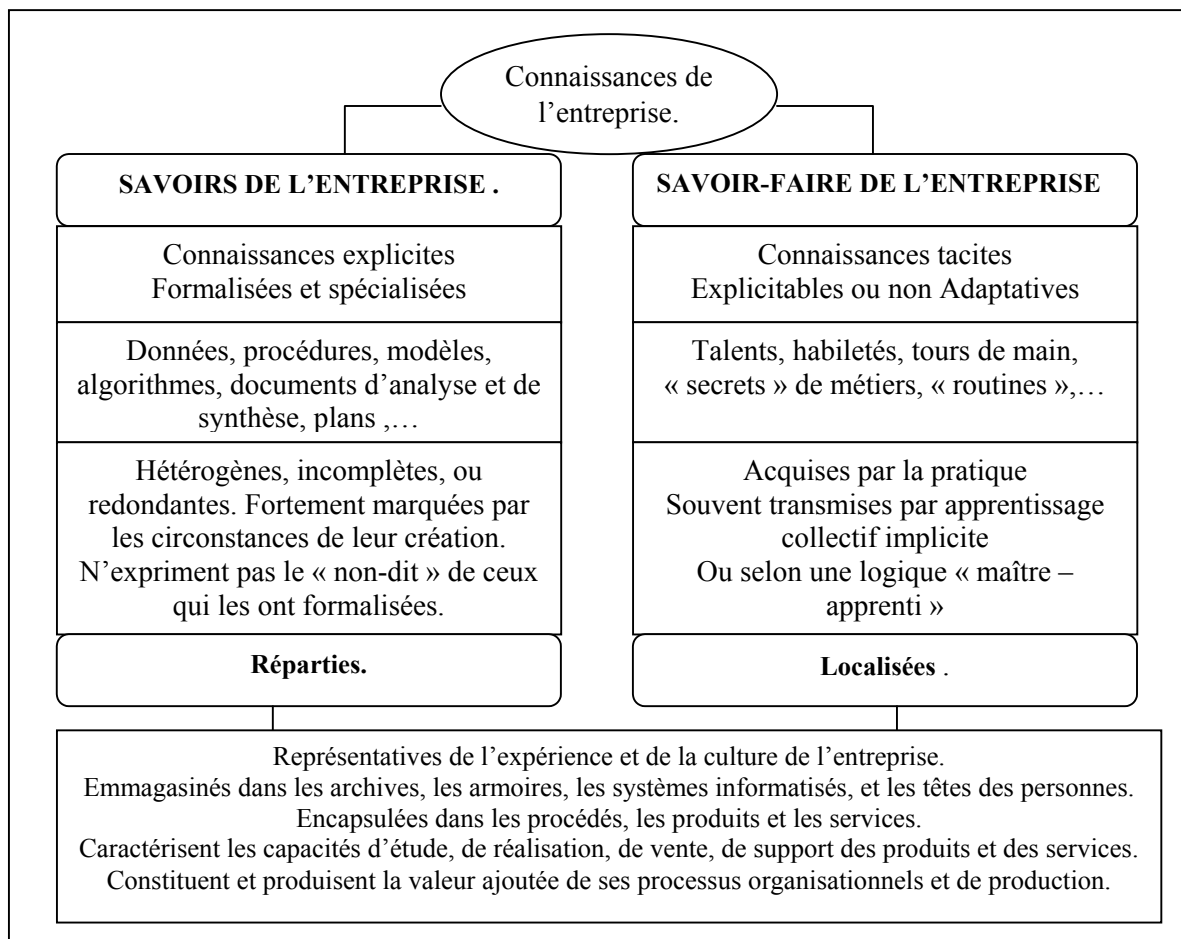
## 2.2. Les connaissances de l'entreprise

Les connaissances de l'entreprise sont généralement présentées selon deux catégories essentielles « Les savoirs de l'entreprise » et « les savoirs-faire de l'entreprise » La figure (1.1) de *Grundstein* résume les caractéristiques de ces deux catégories [38].

---

<sup>(1)</sup> Du verbe « connaître » qui vient du latin *congoscere* et qui a donné « cognition » et « cognitif » [16]

En effet, nous remarquons que les connaissances peuvent être soit explicites soit tacites. Les *connaissances explicites* se résument à ce qui est chiffrable, directement compréhensible et exprimable par chaque individu dans l'organisation, pour ces connaissances il existe toujours une trace visible sous forme d'information (dans des documents par exemple). Les *connaissances tacites*, communément appelées savoir-faire sont propres à chaque individu, elles sont constituées, d'une part, de son expertise technique informelle et d'autre part de ses croyances et aspirations personnelles. Ces connaissances sont celles pour lesquelles l'information associée n'est pas explicite.



**Fig 1.1** *Les Connaissances dans l'entreprise* [38]

Deux grandes catégories de connaissances sont distinguées, les savoirs de l'entreprise où connaissance de domaine et les savoirs faire de l'entreprise où connaissance de contrôle.

**2.2.1 Les connaissances du domaine** : Cette catégorie contient l'ensemble des connaissances clairement partagées par toutes les personnes de l'entreprise travaillant dans le domaine visé. En d'autres termes, ce sont les connaissances de base du domaine dénuées de toute interprétation particulière.

**2.2.2 Les Connaissances de contrôle** : Cette catégorie recouvre toutes les connaissances permettant d'effectuer des raisonnements dans le domaine. Elles décrivent les tâches et les inférences relatives au processus étudié. L'analyse de ces connaissances permet de voir qu'il existe 03 types de connaissances combinés :

- **Les Connaissances Pragmatiques** : décrivent des réflexes, issus de l'expérience, il s'agit de connaissances procédurales, représentées sous la forme de type « situation → action ». Par exemple [65] : *si bruits anormaux du compresseur, alors balourd du rotor du compresseur et équilibrer de nouveau : sinon soupçonner paliers usés par l'encrassement présent dans l'huile, etc.*
- **Les connaissances Causales** : Décrivent des relations de cause à effet plus au moins complexes entre phénomène du domaine. Exemple [65] : Problème de démarrage de la pompe auxiliaire d'une turbine.  
*Si pression de refoulement de la pompe supérieure à la pression de service entre la pompe et manomètre M1 et pression indiquée entre manomètre M1 et manomètre M2 Alors fuite dans le collecteur pompes et vérifier pressostat 63QA-1*  
*Si défaillance du pressostat 63QA-1 Alors étalonnage ou changement du pressostat 63QA-1.*
- **Les connaissances structurelles** : enfin décrivent la structure fonctionnelle du matériel ainsi, à chaque entité fonctionnelle sont attachés les phénomènes la concernant.  
par exemple [65] : Pompe à piston étanchéité HP

**Pompe** : etat (marche, arrêt, insuffisante, bloquée)

Puissance : 106CV

Vitesse :1450 tour/mn

Débit :162 L/mn

Pression aspiration : 1.5à2 At

Pression refoulement : 2At

Produit traité : huile étanchéité

Viscosité : à 65°, 1.9°

### **3. Les systèmes à base de connaissances**

Les systèmes à base de connaissances (SBC) constituent une branche particulièrement florissante de l'intelligence artificielle (IA). L'hypothèse principale qui sous-tend la construction de SBC est la reconnaissance du rôle majeur que jouent les connaissances dans la résolution d'un problème. La construction d'un SBC nécessite donc l'acquisition de connaissances. Ces connaissances peuvent être acquises auprès d'experts du domaine. On pourra alors parler de système expert au sens strict, historiquement les premiers SBC réalisés relèvent plutôt de cette catégorie. Cette tâche d'acquisition des connaissances a longtemps été, et reste encore, une tâche délicate. Il est indéniable que les difficultés rencontrées en acquisition de connaissances lors de la construction de SBC de première génération ont joué un rôle majeur dans l'avènement des systèmes experts dits de deuxième génération.

#### **3.1. Les Systèmes Experts**

Les systèmes experts (SE) constituent les SBC, de première génération, constituaient un premier pas pour la résolution des problèmes complexes, que la rigidité des méthodes algorithmiques utilisées dès le début de son ère ne permettaient pas. Un SE est un logiciel constitué de deux principales composantes : une base de connaissances traduisant le savoir d'un expert dans un domaine précis, et un moteur d'inférence qui opère sur ces connaissances. Les systèmes experts de première

génération utilisent essentiellement des connaissances de type associations (heuristiques) [23] reliant caractéristiques du problème et éléments de solution. Ces connaissances sont représentées de manière uniforme en général sous forme de règles de production qui combinent de manière implicite les tâches à réaliser. Les connaissances ont été également codées sous diverses formes, dont les schémas et les réseaux sémantiques, mais la plupart des SE à succès ont été à base de règles tels que le système MYCIN pour le diagnostic et le traitement des maladies infectieuses, qui s'est affirmé comme étant un excellent système d'aide au diagnostic médical, PROSPECTOR un SE dont le domaine d'expertise est la prospection géologique, XCON qui concerne la configuration des ordinateurs [67].

### **3.2. Problèmes soulevés par les systèmes experts de première génération**

Il convient de rappeler brièvement les difficultés rencontrées dans la construction des premiers SBC. Ces difficultés concernent principalement l'acquisition des connaissances, la production d'explications, le manque de robustesse et l'impossibilité de réutiliser tout ou partie de ces systèmes.

#### **3.2.1. Acquisition des connaissances**

L'acquisition des connaissances (AC) s'est très vite avérée un problème important. Ceci est tout d'abord dû au fait que la construction d'un SE de première génération est essentiellement un travail d'AC, la préoccupation principale est de collecter les règles qui constitueront la BC du système. Ce problème est également dû au fait que les connaissances que l'on cherche à acquérir ne sont pas nécessairement de par leur nature les plus faciles à éliciter pour les experts. Enfin cette difficulté provient du fait qu'aucun guide et aucune méthode ne permet de structurer cette tâche.

#### **3.2.2. Explication**

La capacité des SE de première génération à expliquer leur raisonnement ou à justifier leurs conclusions s'est vite avérée extrêmement limitée, ceci est dû en partie

à la nature de connaissances représentées et le faible niveau d'abstraction utilisé pour décrire le processus de résolution

### **3.2.3. Manque de robustesse**

Les systèmes experts de première génération présentent également le défaut d'être fragiles. Cette fragilité se manifeste essentiellement lorsqu'il s'agit de les faire évoluer, soit parce que le problème lui-même évolue, soit plus simplement lorsqu'on souhaite corriger un comportement non satisfaisant. Dans la pratique, les règles se trouvent si reliées les unes aux autres, que toute modification de la base de connaissances risque de provoquer des modifications inattendues du comportement du système.

### **3.2.4. Réutilisabilité**

Enfin parce que les SE de première génération combinent de manière implicite les tâches à réaliser, les méthodes et les connaissances du domaine, il est impossible d'utiliser telle méthode ou tel modèle d'une application à une autre.

L'essentiel des limites énumérées trouvent leurs causes dans l'uniformité des connaissances et représentations adoptées et dans le faible niveau d'abstraction utilisé pour décrire le système.

## **3.3. Les systèmes experts de seconde génération**

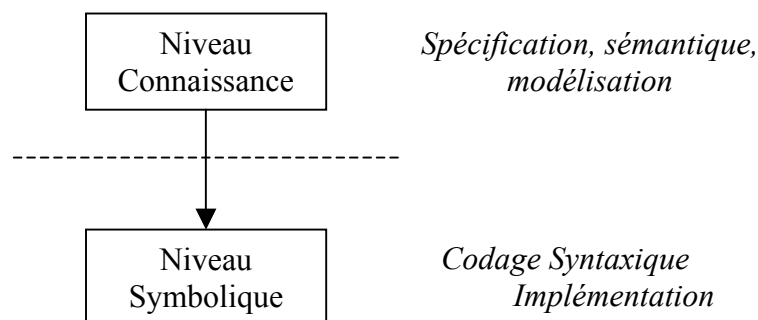
Il n'existe pas de définition claire et précise de ce que sont les SE de seconde génération (SE2G). Ce terme recouvre en fait un ensemble de travaux parfois très différents [23], mais dont la finalité est la même à savoir dépasser les difficultés rencontrées dans la construction des premiers SBC. Ces différents travaux ont toutefois deux directions. Tout d'abord, d'une part à l'uniformité des connaissances utilisées dans les premiers SE, s'est substituée l'hétérogénéité des SE2G : différentes connaissances sont mises en œuvre, différentes représentations sont utilisées, différentes méthodes les exploitent par exemple combinaison de raisonnement causal et heuristique. D'autre part, dans les SE2G la distinction est nécessaire entre les connaissances dont dispose un système (la façon dont le système résoud le

problème) et les représentations utilisées pour les manipuler (la façon dont le système sera implémenté). Ceci a permis de décrire le processus de résolution à un meilleur niveau d'abstraction : le « knowledge level » ce qui a permis d'améliorer la compétence de SBC, leur robustesse, leur capacité d'explication et de réutilisation

### 3.4. L'approche niveau de connaissance ou « Knowledge Level »

Un des principaux apports des SE2G a été de distinguer *un niveau connaissances* (ou « knowledge level »), distinct du niveau implémentation (ou symbolique), dans le but de décrire les SBC à un meilleur niveau d'abstraction. Le terme « Knowledge level » a été introduit par A.Newell [23] comme un moyen de décrire un système comme si ce système possédait certaines connaissances, sans aucun a priori sur la représentation ou l'implémentation utilisée.

Cette thèse d'indépendance a eu un impact appréciable sur les investigations y succédant, en ce sens qu'elle a permis à chacun des deux aspects d'être développés techniquement de manière séparée. Le premier niveau celui des connaissances, nécessite une modélisation de celle-ci. Les modèles construits sont ainsi des spécifications. Les SBC de première génération avaient un niveau unique, symbolique, et c'est cette unicité qui a engendré tous les problèmes de rigidité inhérents à ces systèmes.

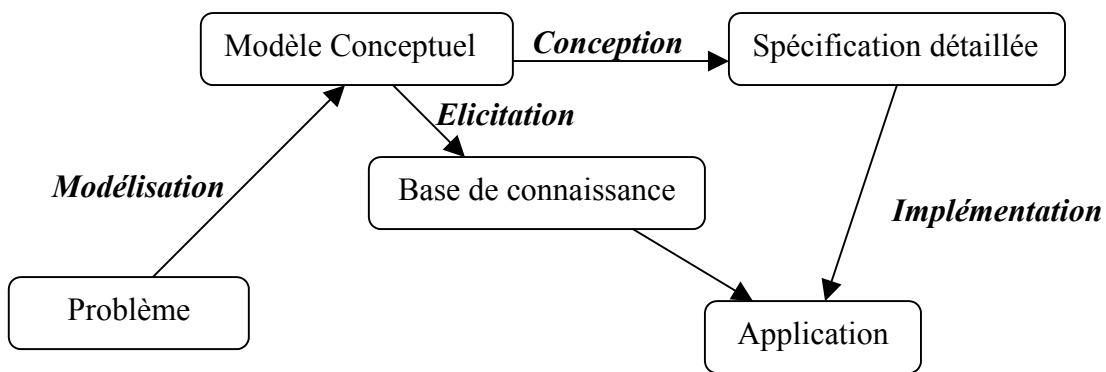


**Fig 1.2** Distinction des connaissances à deux niveaux (Newell 82) [23]

### 3.5. Développement des systèmes à base de connaissances

L'approche « knowledge level » est intimement liée à une méthodologie de conception descendante (Top-Down). La figure (1.3) décrit de manière générale le cycle de développement des SBC selon cette approche.

La tâche de modélisation a pour but de construire le modèle conceptuel du système, c'est à dire une description abstraite du processus de résolution et des connaissances utilisées. Un cadre assez largement répandu pour décrire ce modèle conceptuel s'appuie sur les concepts de tâches, méthodes et modèle du domaine . Les tâches décrivent ce qui doit être fait : quels sont les buts et sous buts du système : les méthodes de résolution décrivent comment ces buts peuvent être atteints, les modèles du domaine décrivent les connaissances qui sont nécessaires pour que ces méthodes puissent s'exécuter.



**Fig 1.3** Conception d'un SBC selon une approche Knowledge Level [23]

Une fois le modèle conceptuel défini, celui-ci va servir de cadre pour l'élicitation des connaissances , c'est à dire la construction effective de la base de connaissances. Il s'agit d'instancier le modèle conceptuel sur le domaine considéré. Nous parlerons donc d'un processus d'élicitation dirigé par les modèles ; l'acquisition des connaissances regroupe ces deux activités : modélisation et élicitation . Le modèle conceptuel est également le point de départ pour la conception détaillée et l'implémentation du système. On parlera alors d'opérationnalisation du modèle conceptuel.

Ainsi le modèle conceptuel pourra être affiné et modifié dans la suite du projet en particulier lors de la phase d'élicitation des connaissances .



### **3.6. Acquisition des connaissances**

L'Acquisition des connaissances, dans la construction des systèmes experts de première génération, était essentiellement considérée comme une activité de transfert, consistant à traduire le discours de l'expert dans le formalisme de représentation choisi (en général les règles de production). Pour répondre au problème d'acquisition de connaissances des premiers SE, le domaine de l'Acquisition des connaissances est devenue une branche de l'IA à part entière, elle s'intéresse donc à l'ensemble des processus permettant d'éliciter, d'explicitier, de modéliser et d'opérationnaliser les connaissances nécessaires au bon fonctionnement du SBC qu'on cherche à le concevoir. L'AC est maintenant principalement considérée comme une activité de modélisation [8]. L'expert, en collaboration avec le cognicien, fournit une description de son modèle de la situation. Ceci relève du domaine de représentation de connaissances. La représentation des connaissances (RC) constitue un des thèmes centraux de la recherche en Intelligence Artificielle (IA). C'est un thème multidisciplinaire qui utilise les théories et techniques de trois domaines : la logique, l'ontologie, le calcul. La logique fournit la structure formelle et les règles d'inférence. L'ontologie, définit les différents types d'éléments qui existent dans le domaine d'application. La RC est l'application de la logique et de l'ontologie à la conception de modèles pour un domaine donné.

Cette phase d'AC est primordiale car de la qualité des résultats qu'elle permet de produire dépend la qualité des performances du système final. Les différents aspects de l'acquisition des connaissances sont la construction du modèle conceptuel, la construction de la BC, dite (l'instanciation du modèle à partir des connaissances du domaine) et dans la mesure où elle est fortement liée à l'AC la validation.

### **3.7. Modélisation**

La construction du modèle conceptuel joue un rôle central dans le processus de réalisation d'un SBC. Outre de permettre une meilleure description du système, ce

modèle va servir à structurer la constitution de la BC et de point de départ pour sa réalisation informatique.

C'est une théorie que nous élaborons sur la manière dont le SBC va résoudre le problème. Les connaissances et raisonnements des experts constituent bien évidemment une source d'inspiration que l'on pourra utilement mettre à profit pour concevoir ce modèle. Différentes approches ont été proposées pour aider à la modélisation comme MACAO[18] ou KOD [18], qui considèrent la construction du modèle comme une activité essentiellement constructive. Le modèle est le résultat d'une démarche ascendante, d'abstraction à partir d'observations sur le comportement des experts. La construction du modèle peut être également vue comme une activité de sélection, le modèle est choisi parmi une liste de modèles existants – cas de l'approche « role limiting methods » ou composé à partir de briques sélectionnées dans une bibliothèque – cas des tâches génériques-

Le cas général est une combinaison de ces deux approches. En effet, la modélisation d'une nouvelle application s'appuie bien souvent sur la connaissance de modèles préexistants, adapter et raffiner des modèles génériques – ou au moins des modèles déjà rencontrés- pour en concevoir un nouveau est ainsi la manière habituelle de procéder. On mêle alors étroitement construction et sélection. La plupart des méthodologies proposent ainsi des bibliothèques de modèles dans le but de supporter ce processus, telle la méthodologie CommonKADS.

La qualité du modèle peut se décliner par rapport aux trois autres entités auxquelles il est directement relié dans le cycle de développement : le problème à résoudre, la BC à construire et le modèle de conception à spécifier.

Du point de vue du problème, un bon modèle doit produire ce qu'on attend de lui, du point de vue de la BC, et de sa construction ultérieure, il est nécessaire que le modèle puisse être effectivement instancié, il ne servirait en effet à rien de concevoir un modèle mettant en œuvre des connaissances que les experts ne pourront pas fournir – ou difficilement –. Enfin, et c'est la troisième perspective, le modèle conceptuel doit pouvoir déboucher sur un modèle opérationnel.

### **3.8. Elicitation dirigée par les modèles**

Une fois le modèle de résolution défini, la construction effective de la base de connaissances peut commencer. Cette deuxième phase s'appuie directement sur le modèle, ainsi défini c'est à dire les connaissances à acquérir, le rôle que vont jouer ces différentes connaissances dans le processus de résolution, quand et comment le système va les utiliser. Le modèle sert ainsi de cadre pour l'élicitation des connaissances.

## **4. L'ingénierie des Connaissances**

L'ingénierie des connaissances est une discipline qui constitue depuis des années un domaine important de recherche en matière d'intelligence artificielle. Elle propose des conceptions et modélisations de systèmes de connaissances ou systèmes à base de connaissances à travers des méthodes et techniques permettant l'acquisition et la représentation des connaissances. L'ingénierie des connaissances peut être définie comme « l'étude des concepts, méthodes et techniques permettant de modéliser et/ou d'acquérir les connaissances pour des systèmes réalisant ou aidant des humains à réaliser des tâches se formalisant à priori peu ou pas » [18]. Les études en ingénierie des connaissances sont surtout dirigées vers l'acquisition des connaissances pour évoluer vers la représentation de connaissances.

## **5. La gestion des connaissances « Knowledge Management »**

### **5.1. Définition et objectifs**

Depuis quelques années, la gestion des connaissances (GC), encore appelée Knowledge Management (KM) s'est imposée comme un enjeu majeur au sein des entreprises, c'est un domaine actuellement en plein essor. L'actualité de ce domaine résulte d'une prise de conscience par un nombre croissant d'entreprises que les connaissances constituent un capital « immatériel » qu'il convient de le gérer au même titre que le capital financier. En effet la mobilité accrue volontaire ou contrainte des employés (mutations, licenciements, retraites ) expliquent que la

gestion des connaissances soit devenue, en tant que telle une préoccupation majeure de nombreuses organisations.

L'objectif de la GC est précisément de formaliser les connaissances tacites afin de les rendre modélisables et opérationnelles au niveau de l'organisation entière. Il s'agit aussi de créer les conditions propices à la création de connaissances nouvelles.

Cette discipline présente une triple dimension : Elle vise à

- a) Rassembler le savoir et le savoir faire sur des supports permanents facilement accessibles, à des fins d'exploitation diverses et rendre explicites les connaissances qui semblent les plus pertinentes.
- b) La diffusion et le soutien de la mise en œuvre des connaissances ce qui permet de rendre les connaissances disponibles aux utilisateurs potentiels qui peuvent être répartis géographiquement.
- c) Aider et orienter la production de connaissances et promouvoir ainsi l'enrichissement des connaissances.

D'une façon plus académique, reprenons la définition donnée dans [13] « La gestion des connaissances recouvre un ensemble de modèles ou méthodes pouvant mettre en œuvre des outils de traitement de l'information et de communication visant à structurer, valoriser et permettre un accès par toute l'organisation aux connaissances qui ont été développées et qui y ont été ou sont encore mises en pratique en son sein ». Il existe deux façons de concevoir la gestion des connaissances. D'un côté elle est envisagée comme un simple processus de communication qui peut être amélioré grâce à un certain nombre d'outils (messagerie électronique, Groupware, Intranet, workflow, Système hypertexte, etc). D'autre côté, il s'agit de capitaliser les connaissances à l'aide d'une mémoire d'entreprise par analogie avec l'esprit humain qui nous permet de construire sur des expériences passées et éviter la répétition d'erreurs, la mémoire d'entreprise doit en effet capturer l'information de différentes sources d'une organisation et la rendre disponible pour effectuer différentes tâches . Le rôle de la mémoire d'entreprise consiste donc à capitaliser les connaissances existantes. Le KM selon *Grundstein* englobe trois composantes essentielles :

- ***Le processus de capitalisation des connaissances*** qui cherche à répondre à la problématique de gestion de connaissances à travers 04 facettes : le repérage

des connaissances cruciales, leur préservation, leur valorisation et leur actualisation.

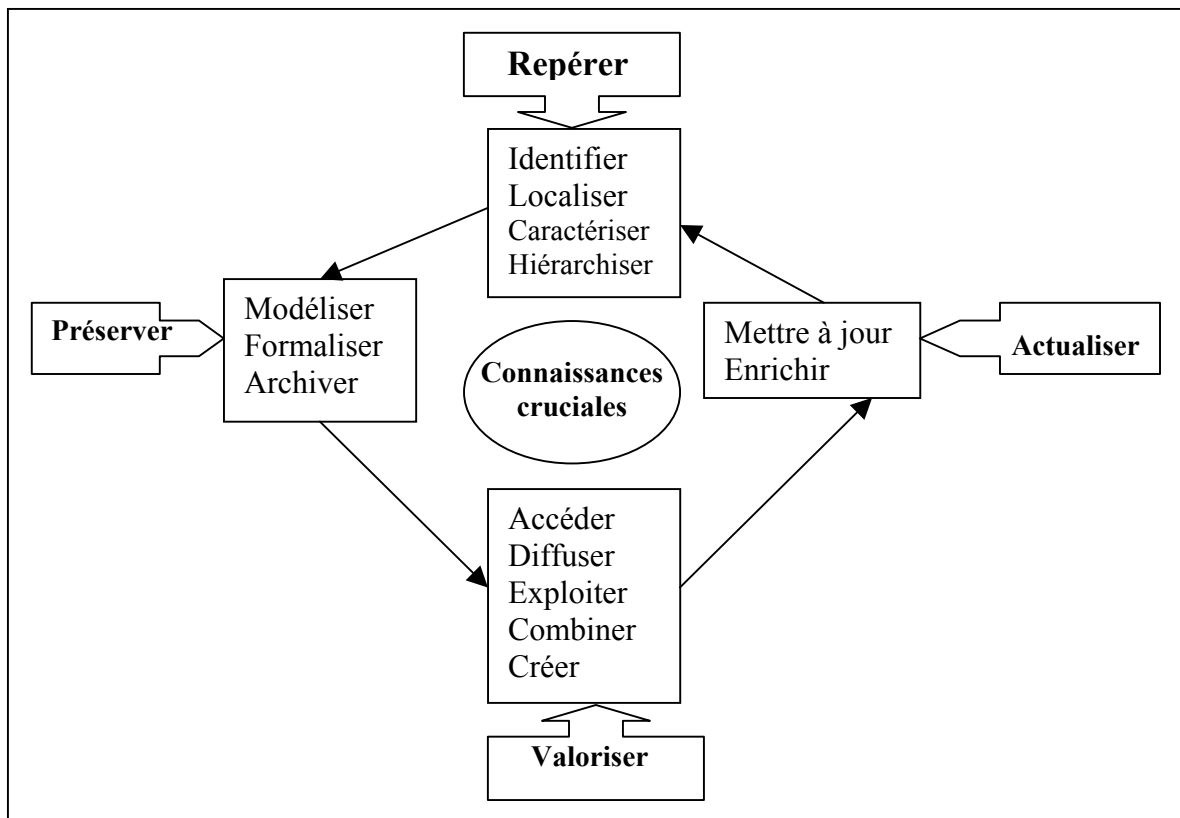
- Développer les actions de revitalisation des connaissances tacites (text-mining, data-mining and knowledge discovery, information search and retrieval, intelligent agents, visualization models, etc.), encourager le retour d'expérience (REX, case-based reasoning, ...etc.) afin d'organiser *la mémoire d'entreprise* (knowledge based systems, knowledge repository, datawarehousing, ...etc.).
- Rendre les connaissances compréhensibles par tous en s'appuyant sur une *ontologie de connaissances* en utilisant des technologies comme l'Intranet et le Groupware.

## **5.2. Les composants de la gestion des connaissances**

### **5.2.1 La capitalisation des connaissances**

Selon *Grundstein* [38], «Capitaliser les connaissances de l'entreprise c'est considérer les connaissances utilisées et produites par l'entreprise comme un ensemble de richesses constituant un capital, et en tirer des intérêts contribuant à augmenter la valeur de ce capital ». La capitalisation des connaissances constitue selon *Grundstein* la cinquième facette de la problématique de la gestion des connaissances, qui essaye de répondre aux quatre autres aspects : le repérage, la préservation, la valorisation et l'actualisation (figure 1.4.)

Le repérage des connaissances cruciales (savoirs et savoir-faire) consiste à les identifier, les localiser, les caractériser et les hiérarchiser, une fois repérées, il est nécessaire de les préserver, les modéliser, les formaliser et les conserver. Ensuite, il convient de diffuser ces connaissances en les valorisant. Ceci permet leur accès à l'ensemble des membres de l'entreprise, leur exploitation et leur combinaison dans le but de créer de nouvelles connaissances. Enfin il faudra constamment actualiser ces connaissances, les mettre à jour, les enrichir en favorisant le retour d'expérience. Ceci confère à la gestion de connaissances son aspect dynamique et non statique.



**Fig 1.4** La problématique de gestion de connaissances (d'après Grundstein.M)[38]

La connaissance devient ainsi le principal moteur de développement de l'entreprise et sa gestion s'inscrit désormais dans la réalité de l'entreprise qui dépasse le stade de capitalisation. La vision stratégique du knowledge management s'inscrit autour de trois axes principaux selon *Ermine* [29] c'est à dire « Capitaliser, partager et créer » le capital de connaissances.

- « *Capitaliser* » : savoir d'où l'on vient, savoir où l'on est , pour mieux savoir où l'on va .
- « *Partager* » : Passer de l'intelligence individuelle à l'intelligence collective.
- « *Créer* » : Créer, innover pour survivre.

Dans le contexte de notre travail, nous nous focaliserons sur la problématique de la **préservation** et de la **valorisation** des connaissances. La préservation renvoie à la question de la mise en œuvre de démarches de modélisation de connaissances en partie tacites. La valorisation renvoie à la question de l'élaboration de supports informatiques permettant l'accès et la diffusion des connaissances modélisées au sein de l'entreprise.

Ceci nous conduit à la deuxième composante de la gestion de connaissances : la mémoire d'entreprise.

### **5.2.2. La mémoire d'entreprise « ME »**

La mémoire d'entreprise est définie comme «la représentation explicite et persistante des connaissances d'une organisation» [44]. La construction d'une mémoire d'entreprise consiste à « préserver, afin de réutiliser les raisonnements, les comportements, les connaissances, même en leurs contradictions et dans toute leur variété » [66]. C'est ainsi que *Dieng* et *Gandon* [40] la définissent comme « la représentation persistante, explicite, désincarnée, des connaissances et des informations dans une organisation, afin de faciliter leur accès, leur partage et leur réutilisation par les membres adéquats de l'organisation, dans le cadre de leurs tâches ». L'entreprise considérée peut être une entreprise réelle dans son ensemble ou un organisme public mais elle peut également juste se restreindre à un département ou un service donné, elle peut également consister en un groupe ou une communauté, voire une entreprise virtuelle (constituée de membres provenant éventuellement de différentes entreprises rassemblées pour un objectif commun). Cette mémoire peut inclure par exemple pour une entreprise commerciale, les connaissances sur les produits, les principes de production, les clients, les stratégies de ventes, les résultats financiers, les plans et buts stratégiques.

#### **5.2.2.1 Caractéristiques d'une mémoire d'entreprise**

Malgré la diversité des définitions proposées pour la notion de mémoire d'entreprise, il existe un certain nombre de constantes ou caractéristiques communes :

**1-** Une mémoire d'entreprise est toujours vue comme l'ensemble des connaissances d'une entreprise résultant de l'expérience et des acquis des personnes y travaillant.

**2- Diversité des connaissances :** On peut trouver les connaissances de type domaine telles que des connaissances sur les produits. On peut également y

trouver des connaissances relatives aux raisonnements mis en oeuvre par les personnes dans une entreprise telles que des stratégies, des plans, des règles, etc. Les connaissances contenues dans une mémoire d'entreprise sont donc très hétérogènes.

**3- Diversité des sources des connaissances :** En plus de l'hétérogénéité des connaissances elles-mêmes, Une mémoire d'entreprise est caractérisée aussi par l'hétérogénéité des sources d'où proviennent ces connaissances. Les connaissances nécessaires à une mémoire d'entreprise résident aussi bien dans :

- *Des documents* de types variés (cours rapports, notes...)
- *La tête des personnes* travaillant dans l'entreprise à tous les niveaux, qui au fil des années ont acquis une certaine expérience et dans ce cas on parle d'experts .
- *Des bases de données*, des bases de connaissances déjà créées par l'entreprise.

**4- Dispersion des connaissances :** Les connaissances d'une ME sont de types variés et proviennent de sources également très variées. Ces deux caractéristiques engendrent donc directement une répartition des connaissances concernées à tous les niveaux de l'entreprise.

**5- Généricité d'une mémoire d'entreprise :** Réaliser une mémoire d'entreprise, ce n'est pas seulement stocker des connaissances, ce qui reviendrait à l'archivage traditionnel. C'est aussi permettre diverses exploitations des connaissances de la mémoire. Cependant sa conception et son implantation ne doivent être influencées par aucun mode d'utilisation particulier. Ainsi les connaissances qui y sont stockées, doivent l'être dans un format permettant plusieurs types d'utilisation.

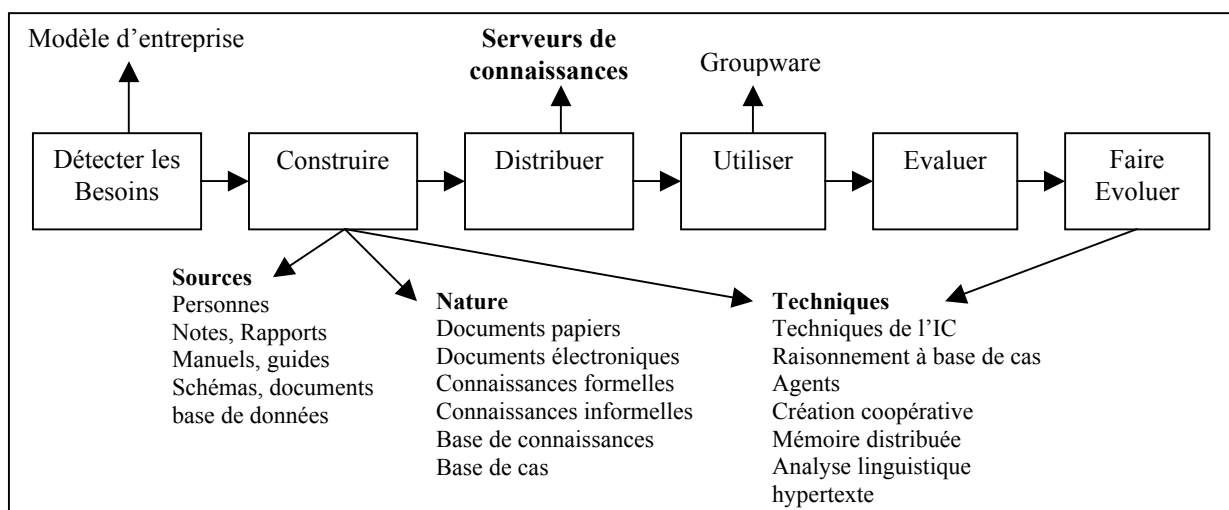


**6- Evolutivité d'une mémoire d'entreprise :** Une mémoire d'entreprise est amenée à évoluer constamment. La principale conséquence de cette caractéristique est qu'une mémoire d'entreprise ne peut se construire que dans la durée et jamais en une seule fois. De plus au fur et à mesure des nouvelles exploitations qui seront envisagées pour cette mémoire, il sera sûrement nécessaire de l'enrichir.

L'ensemble de ces caractéristiques permet de cerner un peu mieux ce que peut être une mémoire d'entreprise et ce qu'elle est susceptible de contenir .

### 5.2.2.2. Le cycle de vie d'une mémoire d'entreprise

Selon *Dieng*, la gestion d'une mémoire d'entreprise passe par six étapes résumés dans la figure (1.5) suivante :



**Fig 1.5** Gestion d'une mémoire d'entreprise [25]

#### a)- Détection des besoins en ME

L'identification des besoins est fondamentale pour la réussite de la démarche comme il est tout aussi fondamental de s'assurer de l'utilisation de la mémoire et de sa maintenance. Comme pour le développement de systèmes d'informations , la détection des besoins réels des utilisateurs constitue la première phase de développement d'une mémoire d'entreprise. Dans cette tâche qui n'est pas toujours évidente, les concepteurs doivent identifier les pré-requis :

- Quels seront les utilisateurs de la ME.
- Quelles sont les tâches à réaliser
- Quelle situation.
- Quel est le type des connaissances à mémoriser et à capitaliser pour la réalisation des tâches
- Quels sont les outils à utiliser.
- --etc

La phase de détection des besoins peut aider à déterminer le type de la mémoire d'entreprise (mémoire de projet, mémoire métier, mémoire organisationnelle, individuelle,.....), les utilisateurs potentiels et les modes possibles d'exploitation utiles et adaptés à leur environnement de travail.

### ***b)- Construction de la mémoire d'entreprise***

Une mémoire d'entreprise est bien sûr différente d'un système à base de connaissances . Les techniques adoptées pour construire une ME dépendent des sources disponibles : spécialistes humains, document papier, documents électroniques tels que les rapports et la documentation techniques, e-mails, base de données existantes, bibliothèque de cas, etc qui dépendent des besoins de la ME. Cette ME peut consister en une mémoire documentaire ou une mémoire à base de connaissances. Dans ce qui suit, On présentera différentes approches pour la construction d'une ME.

### ***c)- Diffusion et utilisation de la mémoire d'entreprise***

#### 1- Les Modes de diffusion possibles

Les éléments adéquats de la ME doivent être distribués aux membres impliqués de l'entreprise : cette distribution peut être active ou passive :

- ***Active*** : un groupe chargé d'administrer la mémoire décide que certaines connaissances sont d'intérêt général et les diffuse au sein de l'entreprise.
- ***Passive*** : chaque utilisateur potentiel de la mémoire la consulte lui-même lorsqu'il a besoin d'informations spécifiques.

Van Heijst [44] propose une classification de ME selon le mode de diffusion et le mode de collecte des informations de cette ME, ce dernier peut être également passif ( la mémoire est enrichie par toute personne de l'entreprise) ou actif (un groupe de personnes est chargé d'observer les échanges d'informations dans l'entreprise pour enrichir sa ME). Selon ces deux critères on distingue :

- *Le grenier à connaissance* (knowledge Attic) : c'est la forme de gestion de mémoire d'entreprise la plus simple mais aussi la plus risquée. La mémoire d'entreprise est utilisée comme un ensemble d'archives consultables à volonté : chaque membre de l'entreprise la fait évoluer et peut l'exploiter. C'est la gestion la plus facile à mettre en œuvre.
- *L'éponge à connaissances* (knowledge sponge) : Dans ce cas l'entreprise, via un groupe de personnes spécifiquement chargées de cette tâche, essaie activement de construire une mémoire la plus complète possible. Par contre l'exploitation de cette mémoire est laissée à la charge de chacun des membres de l'entreprise qui est libre de consulter ou de ne pas consulter la mémoire afin d'améliorer son activité.
- *L'éditeur de connaissances* : (Knowledge publisher) : La collecte des informations est assurée par chacun des membres de l'entreprise, l'exploitation de la mémoire est assurée par un groupe de personnes spécifiquement chargés de cette tâche.
- *La Pompe à connaissances* : (Knowledge pump) ce type de mémoire nécessite la mise en place à la fois d'un groupe de personnes spécifiquement chargées de l'évolution de la mémoire et d'un autre groupe de personnes spécifiquement chargées de l'exploitation de la mémoire .

## 2- Diffusion Intranet/Internet

La diffusion des connaissances peut par exemple passer par un accès à l'Internet ou à l'Intranet de l'entreprise. La diffusion peut dépendre d'un serveur de connaissances sur le Web ou d'un éditeur sur le Web. Différents types d'éléments peuvent y être accédés à travers Internet/Intranet : documents(classique ,Html,..),

base de données, ontologies, base de connaissances, base de cas, articles dans des journaux numériques, etc. donc plusieurs types de serveurs de connaissances peuvent être considérés : un serveur de document, un serveur d'ontologie, un serveur de base de connaissances, un serveur de base de données, un serveur de journaux ou une bibliothèque numérique. Les principaux problèmes à résoudre sont :

- (1) Récupération des éléments de la ME en réponse à une requête .
- (2) Adaptation de la réponse à l'utilisateur selon son profit .

#### ***d)- Evaluation de la mémoire d'entreprise***

Une évaluation d'un projet de capitalisation des connaissances est un problème important qui peut être abordé de plusieurs points de vue : économique-financier, socio-organisationnel et technique .

D'un point de vue économique-financier il faut une évaluation du gain obtenu suite à l'introduction d'une ME . En général ce gain concernant l'amélioration de la productivité du point de vue du manager, le revenu sur l'investissement est important pour justifier l'intérêt de construction de la ME.

D'un point de vue socio-organisationnel, la ME peut viser l'amélioration de l'organisation du travail des employés ainsi que la satisfaction des employés dans leurs travaux. Les critères pour une telle évaluation sont en général qualitatifs. Parmi ces critères, la facilité de récupération des informations, l'adéquation des informations récupérées ,confidentialité éventuelle des informations ...etc.

D'un point de vue technique, le transfert du savoir-faire à l'intérieur de l'entreprise semble avoir un bénéfice évident, mais le transfert réel dépend d'une utilisation réelle de la ME, ainsi l'introduction d'une ME peut impliquer des changements dans le travail individuel et collectif dans l'entreprise.

En conclusion on doit distinguer entre deux types d'évaluation : évaluation par les utilisateurs (avec des critères basé sur la satisfaction des utilisateurs) et une évaluation stratégique par le manager (avec des critères basés sur le revenu sur l'investissement)

### ***e)- Maintenance et évolution de la ME***

Une mémoire d'entreprise contient un ensemble de connaissances beaucoup trop important pour être stocké et modélisé en une seule fois. De plus, au fur et à mesure de nouvelles exploitations qui seront envisagées pour cette mémoire, il sera sûrement nécessaire de l'enrichir. Donc, une mémoire d'entreprise est vouée à évoluer en permanence. D'une part, au fil des années, l'entreprise s'enrichit de nouvelles expériences dont il est possible de tirer des connaissances pertinentes pour la mémoire. D'autre part, les connaissances stockées et modélisées dans une ME n'ont de sens que par rapport à une dimension temporelle. Au fil du temps, certaines connaissances peuvent devenir obsolètes et doivent donc être soit supprimées soit mises à jour. Par exemple, il se peut très bien qu'une mémoire d'entreprise contienne des connaissances sur un équipement particulier utilisé par cette entreprise si cet équipement vient à évoluer ou disparaître, les connaissances le concernant devront être mises à jour.

### **5.2.2.3. Typologie de mémoire d'entreprise**

Il existe de multiples façons de classer les mémoires d'entreprise. L'ensemble des typologies utilise des critères de distinction de natures différentes, les trois principaux critères considérés sont : Le support employé pour la représentation de la mémoire d'entreprise, le deuxième critère souvent utilisé est le contenu de la mémoire. On s'intéresse ici à différencier les mémoires en fonction du type des connaissances qu'elles contiennent et enfin le type d'exploitation.

Suivant les objectifs poursuivis pour la construction d'une mémoire d'entreprise et suivant la nature des connaissances à capitaliser on peut distinguer mémoire organisationnelle, technique, métier, société, individuelle et de projet .

❶ ***Mémoire organisationnelle*** : appelée aussi mémoire société ou mémoire managériale. Une telle mémoire permet de préserver les connaissances, partagées implicitement par toutes les personnes d'une entreprise et qui contribuent au bon fonctionnement de cette entreprise. Il s'agit en tant ici de

connaissances ayant trait à l'entreprise autant qu'organisation (Ressources humaines, management...).

❷ **Mémoire technique** : Une mémoire technique est centrée sur un domaine relativement pointu, elle fait appel au savoir-faire d'experts, appelé expertise

❸ **Mémoire métier** : La mémoire métier fait référence à un domaine d'activité particulier de l'entreprise, le plus souvent technique. Elle est Composée de :

- Un référentiel : désigne l'ensemble des termes techniques, des définitions et des concepts issus du domaine visé par la mémoire métier
- Les documents : Ce sont les ouvrages de références du domaine , des articles...
- Les outils : ce sont l'ensemble des outils informatiques utilisés par l'entreprise dans le domaine comme par exemple des bases de données ou des logiciels.
- Les méthodes : L'ensemble des règles et procédures utilisées par les personnes du domaine dans l'entreprise.

❹ **Mémoire individuelle** : Elle permet de décrire les caractéristiques de l'ensemble des employés de l'entreprise. Quatre ensembles de connaissances sont à considérer : Le statut, les compétences, le savoir-faire et les activités d'un membre donné de l'entreprise.

❺ **Mémoire de projet** : comportant la définition du projet, ses activités, son historique et ses résultats afin de pouvoir réutiliser des expériences antérieures.

Outre ces typologies, on distingue la mémoire interne correspondant aux connaissances et aux informations internes de l'entreprise et la mémoire externe correspondant aux connaissances et aux informations utiles pour l'entreprise mais provenant du monde extérieur. Ces types de mémoires d'entreprises ne sont pas exclusifs, en effet une mémoire métier qui regroupe des connaissances techniques est considérée également comme une mémoire technique.

Suivant la nature des sources d'expertise disponibles et les objectifs visés de la mémoire d'entreprise au niveau des utilisateurs, la ME peut prendre la forme d'un système documentaire ,base de connaissance ,base de cas , serveur de connaissance via le web, collecticiel ou un système multi-agents .....

- a) **Mémoire documentaire** : C'est un système de consultation de documents techniques. Pour le construire, on se base sur des techniques d'ingénierie documentaire et des approches hypertextes en particulier le standard XML qui constitue un format d'échange standard de documents et de données structurées et hétérogènes sur le Web. La constitution d'une telle mémoire consiste donc à considérer l'ensemble des documents de l'organisation qui ne sont pas généralement bien indexés, donc la constitution d'une mémoire d'entreprise débute par l'indexation de tous les rapports, documents de synthèse et référentiels utilisés par les différents experts et elle doit contenir une interface pour la gestion des documents (addition des documents, retrait de documents, .... Etc).
- b) **Mémoire à base de cas** : Permet de décrire les expériences antérieures (succès ou échecs) de l'entreprise dans une base de cas ,le raisonnement à partir de cas permettra ensuite de trouver, pour un problème présent, une solution mémorisée d'un problème similaire au problème actuel.
- c) **Le Web sémantique** : Dans le cadre d'une mémoire d'entreprise matérialisée dans des documents, il est intéressant d'associer à de tels documents une connaissance formelle sur laquelle pourra être effectué un raisonnement afin de rechercher les documents adéquats ou les parties adéquates du document. Cette connaissance formelle peut soit représenter une partie du document soit consister en des méta-informations sémantiques sur le document. Une ontologie peut guider la recherche documentaire des documents accessibles par le Web et annotés sémantiquement selon cette ontologie Une telle approche vise à améliorer

les moteurs classiques de recherche de Web par des capacités basées sur la recherche d'informations sémantiques. Cette approche est similaire à celle du Web, qui vise à rendre le contenu sémantique du Web interprétable par la machine.

- d) **Mémoire à base de connaissances** : L'ingénierie des connaissances est utile pour construire une mémoire d'entreprise basée sur le recueil et la modélisation explicite des connaissances de certains experts ou spécialistes de l'entreprise. Elle peut aussi servir pour une représentation formelle des connaissances sous-jacentes à un document. Cependant, contrairement aux systèmes experts, au lieu de viser une résolution automatique pour une tâche (avec des capacités automatiques de raisonnement), une mémoire d'entreprise doit plutôt aider l'utilisateur, en lui fournissant des informations appropriées de l'entreprise mais en lui laissant la responsabilité d'une interprétation et d'une évaluation contextuelles de ces informations. Selon le cas, la mémoire d'entreprise pourra intégrer ou non des connaissances formelles. Si l'on explicite les connaissances d'un expert, on peut les écrire soit dans un document textuel ou multimédia, soit dans une base de connaissances formelle (représentée dans un formalisme de représentation des connaissances tel que les formalismes objets, les réseaux sémantiques, les graphes conceptuels ...etc).
- e) **Système Multi-agents** : La ME et les SMA ont un point commun, ils sont tous deux des paysages d'informations hétérogènes et distribués. Ils partagent également le même problème de la pertinence des résultats obtenus lors d'une recherche dans leur contenu. Cependant, comparée au Web, une ME a un contexte, une infrastructure et une portée délimités et mieux définis : L'entreprise. En tenant compte des caractéristiques d'une ME, on peut adopter une approche combinant la technologie XML conçue pour le Web et la nature distribuée des SMA.



La gestion des connaissances implique impérativement la capitalisation des connaissances cruciales. Ceci peut se faire grâce à la construction d'une mémoire d'entreprise, qui doit être comprise par tous afin d'être utilisée et diffusée. Mais afin de gérer cette mémoire d'entreprise, il faut établir une ontologie des connaissances de l'entreprise. C'est ce qu'affirment *Dieng* [16] : « les ontologies sont très utiles dans une mémoire métier ou dans une mémoire technique car elles explicitent une terminologie et une conceptualisation partagées par un métier donné ou par une communauté donnée dans une organisation ».

### **5.2.3 L'ontologie**

Une ontologie est définie par *Gruber* comme une spécification explicite d'une conceptualisation [19]. Une conceptualisation étant définie [68] comme étant « une structure sémantique intentionnelle qui capture les règles implicites contraignant la structure d'un morceau de la réalité ».

La représentation d'un domaine donné nécessite de se restreindre à un certain nombre de concepts significatifs suffisants pour interpréter ce domaine; le choix d'un ensemble donné de concepts plutôt qu'un autre s'effectue selon des engagements ontologiques [16].

L'ontologie permet de fournir un vocabulaire conceptuel dans un domaine donné afin de permettre la communication au sujet de ce domaine. Elle permet de définir les concepts utilisables pour décrire une connaissance consensuelle ainsi que les relations qui existent entre les concepts et leurs contraintes d'utilisation.

Une ontologie fournit donc un cadre unificateur des concepts manipulés ce qui favorise la communication entre les membres de l'entreprise. L'ontologie peut aussi permettre l'interopérabilité entre différents méthodes, paradigmes, langages ou outils. Elle peut permettre le multipoints de vues sur des connaissances. Enfin, elle permet de favoriser la réutilisabilité, la fiabilité et la spécification d'un système informatique en particulier d'une mémoire d'entreprise.

Comme une mémoire d'entreprise vise l'intégration de sources de connaissances hétérogènes, elle peut donc comprendre différents types d'ontologies pour formaliser ces connaissances à différents niveaux d'abstraction :

- L'ontologie du domaine spécifie les concepts relatifs du domaine, en liant les différentes sources de connaissances.
- L'ontologie de tâches spécifie les concepts nécessaires à la description des activités considérées par la ME apportant ainsi le point de vue de raisonnement sur les connaissances du domaine.
- L'ontologie de cas spécifie les concepts nécessaires à la description des cas.

Notons que les ontologies de domaine sont les plus utilisées. Pour pouvoir utiliser puis exploiter une ontologie, deux problèmes essentiels se posent. Tout d'abord en terme d'élaboration de l'ontologie, il est nécessaire de trouver une méthodologie permettant sa construction. Ensuite pour pouvoir exploiter une ontologie, il est nécessaire de la représenter (structure et sémantique des concepts qu'elle définit) au moyen d'un langage plus au moins formel. La construction d'une ontologie des connaissances utilise donc plusieurs techniques pour l'acquisition et la représentation des connaissances, il s'agit de la modélisation des systèmes de connaissances par l'ingénierie des connaissances.

### **Les méthodologies de création d'ontologies**

Différentes méthodologies ont été élaborées pour la création d'ontologies. Parmi les projets qui ont proposé des méthodes et des outils de construction d'ontologies, citons : Methontology, (KA)<sup>2</sup>, Terminae, Ontolingua, .... Chaque méthode définit un processus de développement. Une généralisation de ces processus de mise au point d'une ontologie repose sur l'enchaînement des phases suivantes [68]:

- **Le cadrage** consiste à cerner l'étendue de l'ontologie et le domaine à prendre en compte.
- **La construction** de l'ontologie correspond au travail central qui est celui de la conceptualisation.
- **L'évaluation** doit permettre de vérifier la validité de l'ontologie. En effet différentes méthodologies mettent en évidence la nécessité d'évaluer les ontologies. Pour certaines méthodes il s'agit d'une véritable étape qui est réalisée après l'étape de construction, notamment en vérifiant que l'ontologie

permet bien de répondre aux besoins spécifiés au départ. Dans d'autres cas, l'évaluation est réalisée tout au long du processus d'élaboration

Les différentes activités relatives à la phase de construction de l'ontologie sont :

**a- Le recueil d'informations :** comprenant notamment la sélection des sources d'informations. Différentes sources d'informations peuvent être utilisées : corpus, entretiens avec les experts, base de données, et ontologies réutilisables

**b- L'analyse :** devant permettre la détermination des concepts, souvent l'analyse s'effectue de façon manuelle. Néanmoins des outils automatiques peuvent être utilisés en particulier ceux pour l'analyse des corpus, la technique des scénarios utilisée avec les experts et les techniques d'ECD.

**c- La structuration et la modélisation des concepts :** C'est au cours de cette activité que l'ontologie peut être formalisée au niveau de la structuration. Une ontologie conceptuelle sera établit, cette dernière est basée sur un modèle conceptuel complet et rigoureux dans lequel les connaissances sont organisées et structurées en faisant abstraction à toute contrainte informatique. Elle sert donc de support à l'acquisition des connaissances .Pour sa structuration trois tendances sont distinguées :

1. **L'approche ascendante** qui consiste à construire l'ontologie en partant des concepts les plus spécifiques par généralisation .

2. **L'approche descendante** est l'approche inverse qui consiste à partir des concepts les plus génériques et de les spécialiser.

3. **L'approche centrifuge** consiste à identifier des concepts centraux . l'ontologie est ensuite complétée par généralisation et spécialisation de ces concepts.

Enfin au niveau de la modélisation, il est nécessaire d'utiliser un langage de représentation des connaissances pour aboutir à une ontologie opérationnelle.

plusieurs environnements pour l'édition d'ontologies ont été proposés : Ontolingua Server fondé sur KIF (Knowledge Interchange Format) langage basé sur la logique du premier ordre , Webonto, protégé-2000,etc.

## **6. Conclusion**

En résumant, les connaissances se trouvent dans l'intelligence, le savoir-faire, la mémoire, l'expérience des personnes autant que dans les documents et les bases de données, etc . L'objectif principal est d'extraire cette connaissance pour espérer la conserver et pouvoir la transmettre en vue de sa réutilisation.

En effet une simple accumulation de connaissances est loin de constituer une solution viable il est nécessaire de les structurer et de les mettre en place en intégrant des moyens permettant de les retrouver, en vue de prendre les meilleures décisions dans des environnements de plus en plus complexes, autrement dit de gérer ces connaissances . La gestion de connaissances s'occupe ainsi d'identifier, collecter, capitaliser et diffuser les connaissances d'une entreprise.

# Les Méthodes de gestion des connaissances

## Chapitre 2

*N*ous présentons dans ce chapitre quelques méthodes de gestion de connaissances avec une synthèse des comparaisons effectuées entre ces méthodes .

## 1. Introduction

La gestion des connaissances dans une organisation a pour objectif de favoriser la croissance, la transmission et la conservation des connaissances dans cette organisation. Elle peut porter sur le savoir théorique que sur le savoir-faire et nécessite la gestion des ressources de connaissances de l'entreprise afin de faciliter leur accès et leur réutilisation. Elle consiste donc à capturer et représenter les connaissances pour faciliter leur accès, leur partage et leur réutilisation.

L'ingénierie des connaissances permet de transmettre et d'enregistrer des savoirs et savoir-faire de gestion des connaissances. C'est un outil d'aide à la résolution de problèmes.

On peut distinguer trois approches méthodologiques pour la gestion des connaissances :

- Les approches descendantes, reposant sur des méthodes de modélisation des connaissances telles que CommonKADS et MKSM ;
- Les approches ascendantes reposant sur l'analyse manuelle ou semi automatique des sources de connaissances
- Les approches coopératives mettant l'accent sur l'exploitation de collecticiels.

Outre les méthodes classiques d'ingénierie des connaissances, CYGMA (Cycle de vie et Gestion des Métiers et des Applications) [25] est dédiée à la mémoire de métiers, dans le cadre d'une tâche de conception. La méthode REX (acronyme de retour d'expérience)[12], se base sur la construction d'éléments d'expérience provenant de différentes sources (humains, documents ,bases de données). La méthode MKSM (Methodology for Knowledge System Management) [29] est basée sur la théorie de systèmes complexes pour analyser les connaissances d'une organisation, considérées comme constituant un système complexe. La méthode SAGACE est utile pour construire la mémoire d'un projet de production d'un système [54].

Nous présentons dans ce qui suit les principes de quelques méthodes, puis nous élaborons un tableau récapitulatif pour la comparaison de ces méthodes .

## 2. La méthodologie CommonKADS

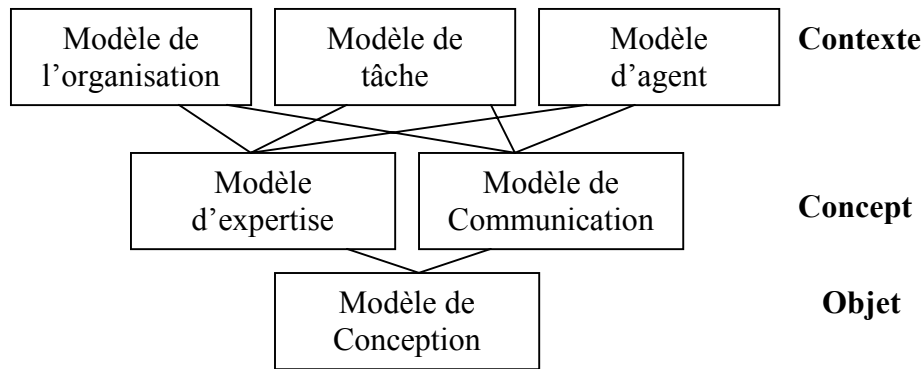
La méthodologie KADS [63] (Knowledge Acquisition and Design System) est née en 1985 dans le cadre du projet européen Esprit I. La méthodologie a été améliorée dans le but d'en faire un standard commercial. Cette méthodologie se nomme désormais CommonKADS[64]. Kads est une méthodologie permettant de mettre en place un processus d'acquisition des connaissances pour la construction de systèmes à base de connaissances. Elle permet de traiter tout le processus d'acquisition des connaissances depuis le recueil de connaissances jusqu'au développement d'un système complet. Il s'agit d'une méthode dirigée, à l'instar de MKSM par les modèles, par opposition aux méthodes dirigées par l'implémentation. La méthodologie CommonKADS a évolué encore récemment pour être industrialisée, elle a en particulier évolué pour prendre en compte les aspects de gestion de connaissances en plus de l'ingénierie des connaissances. Elle repose sur le postulat que le partage de la connaissance repose sur la communication et la création de connaissance. La méthodologie CommonKADS prend en compte l'évolution de l'extraction des connaissances de l'expert vers la construction de méthodes. Elle repose également sur le principe du Knowledge Level (ou niveau des connaissances) [23]. CommonKADS permet de modéliser l'organisation dans laquelle sera intégré le système à base de connaissances. Elle propose six modèles .

### 2.1. Les modèles de CommonKADS

Les différents modèles introduits par CommonKADS permettent de couvrir les différents problèmes devant être traités lors du développement de SBC. Néanmoins selon le projet tous les modèles ne sont pas forcément nécessaire. CommonKADS propose six sortes de modèles [45] pour analyser et représenter la connaissance : organisation, tâche, agent, communication, expertise(connaissance) et conception. C'est une méthode descendante de résolution de problèmes qui part du contexte, pour définir le concept qui aidera à la conception d'objet. Les quatre premiers modèles présentent un intérêt particulier pour l'analyse préalable à la capitalisation des connaissances

- **Le modèle de l'organisation** : il permet de décrire l'entreprise dans son ensemble avec ses grandes fonctions et ses capacités en terme de ressources, Il facilite l'identification des impacts possibles sur l'organisation lors de l'intégration d'un SBC.
- **Le modèle de tâche** : il décrit les tâches qui sont exécutées dans l'environnement de l'organisation. Ce modèle contient les tâches nécessaires à la réalisation des fonctions décrites dans le modèle de l'organisation.
- **Le modèle d'agent** : Il permet de modéliser les agents, humains ou informatiques, impliqués dans la réalisation des tâches
- **Le modèle de communication** : Il se charge de rendre compte de la communication nécessaire entre les agents (notamment de la communication homme-machine) pour la réalisation d'une tâche .
- **Le modèle d'expertise** : il décrit les capacités de résolution de problèmes d'un agent. Il permet donc de modéliser l'expertise nécessaire à la réalisation des tâches par les agents. Il différencie notamment les connaissances applicatives des connaissances stratégiques. La bibliothèque KADS propose des méthodes de résolution de problèmes (PSM) permettant d'aider à la conception du modèle d'expertise au travers de modèles d'interprétation. Un modèle d'interprétation est une sorte de méta-modèle d'expertise pour lequel la couche domaine est vide et doit être instanciée lors d'une utilisation pour un SBC particulier.
- **Le modèle de conception** : il concerne la conception du SBC qui a pour objectif l'opérationnalisation des connaissances modélisées.





**Fig 2.1** Les modèles de CommonKADS [16]

## 2.2. La réutilisation

La méthodologie CommonKADS propose une typologie standard de tâches à base de connaissances, ainsi qu'une décomposition fonctionnelle de ces tâches. On y trouve par exemple des modèles pour les tâches de diagnostic, de conception, de planification, etc. Une bibliothèque apportant une aide à la modélisation est fournie. Elle permet de s'inspirer des modèles de la bibliothèque lorsqu'on démarre une nouvelle application, plutôt que de repartir de zéro.

La bibliothèque distingue les tâches d'analyse et les tâches de synthèse. Elle repose sur la réutilisation des combinaisons d'éléments de modèles. Pour la réutilisation des connaissances du domaine, on passe de la notion de schéma de domaine à la notion d'ontologie. Pour CommonKADS, une ontologie est un point de vue sur un domaine. CommonKADS est fortement axée sur l'opérationnalisation, les modèles utilisés sont solidement formalisés. Ainsi différents outils ont été élaborés pour simplifier l'instanciation des modèles (suivant notamment les conventions de représentation graphique UML) et leur opérationnalisation. Parmi ces outils Cokace et WebCokace. Cokace (CommonKADS Centaur) [7] est un environnement dédié à la construction de modèles d'expertise décrits par le langage CML. Cokace permet une édition structurée, une validation statique et une interprétation dynamique de modèles d'expertise CML. Quant à WebCokace [7] c'est un serveur de connaissances permettant de diffuser des modèles d'expertise CommonKADS sur le Web. Il permet d'explorer des modèles d'expertise et d'effectuer à l'aide d'un navigateur Web une navigation hypertexte dans des modèles d'expertise et vers des documents électroniques associés.

CommonKADS est donc une approche descendante qui consiste à choisir dans la bibliothèque des modèles génériques, un modèle type et l'adapter à un domaine . Elle donne la sémantique générique pour guider à établir une sémantique spécifique à un domaine d'application.

### **2.3. Le cycle de gestion de projet CommonKADS**

CommonKADS propose un cycle de vie basé sur les méthodes de développement de logiciels [53]. Le processus d'élaboration d'un SBC avec KADS est un processus itératif, commençant généralement par une phase de prototypage rapide. Une analyse complète des données du système précède la conception et l'implémentation du SBC. Les résultats de la phase d'analyse (contraintes externes, modèle conceptuel des objets du domaine et les méthodes de résolution) servent d'entrée à la phase de conception où ils sont transformés en spécification de l'architecture fonctionnelle du SBC.

Les différents modèles proposés permettent non seulement la capture et la représentation en machine d'un savoir-faire, mais aussi la compréhension du problème réel, de l'organisation et de ses besoins. Ce sont donc aussi des documents de référence pouvant être directement intégrés à la mémoire d'entreprise.

### **3. La méthodologie MKSM**

MKSM(Methodology for Knowledge System Management) est une méthode d'analyse d'un patrimoine de connaissances d'une organisation [28]. Elle a été développée au sein de la Direction de l'Informatique Scientifique et Technique (DIST) de CEA (Commissariat à l'énergie Atomique) en 1993. La méthodologie continue d'être améliorée à l'université de technologie de Troyes et porte désormais le nom de MASK (Method for Analysing and structuring knowledge). MKSM vise à expliciter tout ou partie d'un patrimoine de connaissance afin de le rendre intelligible et compréhensible aux acteurs concernés, afin qu'ils puissent mettre en place un système de gestion de connaissances.

La démarche MKSM propose à la fois des modèles permettant l'analyse et la modélisation des connaissances d'une organisation et un cycle de développement de projet [68].

### 3.1 Les modèles MKSM

Les modèles de connaissances proposés par la méthode MKSM se fondent sur deux hypothèses principales :

- *L'hypothèse systémique* : Le patrimoine de connaissances d'une organisation forme un système complexe à part entière. Ce système de connaissances est, au sens de la théorie du système général, un objet actif et stable et évoluant dans un environnement, et par rapport à quelques finalités. Ce système peut donc être étudié selon trois points de vues indissociables : sa structure, sa fonction et son évolution.

- *L'hypothèse sémiotique* : toute connaissance peut être observée selon trois niveaux indissociables : le niveau syntaxique (données, représentation de l'information), le niveau sémantique (signification de l'information) et le niveau pragmatique (correspond au contexte, à l'environnement dans lequel cette information prend un sens). Cette vision peut être résumée par la formule : « Un système de connaissances est vu comme de l'information qui prend une certaine signification dans un contexte donné » .

La démarche de MKSM procède en construisant des modèles successifs. Chaque modèle correspond à une combinaison des deux approches systémique et sémiotique. L'ensemble de ces points de vues forment ce qui est appelé le microscope de la connaissance. A chaque point de vue correspond un modèle graphique de représentation. Les modèles sont instanciés lors d'une phase de recueil de connaissances auprès d'experts, de spécialistes ou de documents de référence.

L'ensemble des modèles instanciés et organisés de façon structurée forme ce que l'on appelle un livre de connaissances qui est le produit final de tout projet MKSM et correspond au cœur de la mémoire d'entreprise. Parmi les modèles proposés par MKSM, citons :

- *Le modèle du système de référence* : ce modèle permet d'appréhender, d'un point de vue systémique, l'activité ciblée en identifiant les flux intrants et extrants (les flux entre le système et son environnement définissant la finalité du système), les agents cognitifs (il peut s'agir de personnes physiques de matériels ou de sous systèmes spécifiques), les flux d'informations et les flux de connaissances ( définis entre les agents et le patrimoine de connaissances). Le système global est composé de quatre sous-systèmes : le système opérant, le système de décision, le système d'information et le système de connaissances.

1. Le système opérant transforme le flux intrant en flux extrant : c'est lui qui apporte de la valeur ajoutée.
2. Le système de décision pilote les activités du système opérant.
3. Le système d'information est décrit par des agents cognitifs au sens outils de gestion d'information
4. Le système de connaissances comprend l'ensemble des savoirs et savoir-faire utilisés ou créés par les différents agents cognitifs du système global.

- *Le modèle de domaine* : Il doit permettre la description de l'aspect structurel du domaine de la connaissance. Il s'agit de décrire, sous forme de processus, les phénomènes fondamentaux (généralement physiques ou biologiques) qui doivent être maîtrisés dans le cadre de l'activité des experts.

- *Le modèle d'activité* : permet d'analyser et de représenter l'activité d'un point de vue fonctionnel et hiérarchique. Il décrit, à l'aide d'un formalisme de type SADT (Structured-Analysis-Design-Technique) les différentes activités qui composent le processus métier.

- *Le modèle de concepts* : Le premier modèle de MKSM dédié à la représentation sémantique est le modèle des concepts qui repose sur le langage de représentation MOISE, plus précisément sur la partie dédié aux connaissances statiques. Ce modèle permet de représenter et d'organiser les différents concepts (catégories d'objets) manipulés par les agents cognitifs du système de référence. Différents types de relations peuvent être définies entre les concepts : la spécialisation, l'agrégation et la valuation.

- *Le modèle de tâches* : Il permet de décrire le savoir-faire que l'expert exploite pour résoudre un problème relatif à son travail. Ce modèle repose comme le précédent sur le langage de représentation MOISE, et plus particulièrement sur la partie dédiée aux connaissances dynamiques. Il permet de décrire sous une forme hiérarchique, la décomposition des tâches en sous tâches plus spécifiques. Par ailleurs, des structures de contrôle permettent de définir l'ordonnancement des sous tâches qui peut être de type séquentiel, parallèle, conditionnel ou itératif.

### **3.2. Le cycle de vie d'un projet MKSM**

La méthodologie MKSM se concentre sur l'analyse et la modélisation des systèmes de connaissances, elle laisse de côté les problèmes de réalisation et d'implantation. La conduite de projet MKSM comporte trois phases : le cadrage, le cycle de modélisation et le schéma d'orientation.

L'objet de la phase de cadrage est de déterminer le domaine d'application, le niveau d'approche (stratégique, opérationnel ...) ou le sujet d'étude de projet. Cette phase permet ensuite de définir les modèles qui seront utiles au projet, elle permet également d'identifier et de mettre en place les acteurs du projet : le comité de pilotage qui suit et oriente, le comité de projet qui réalise, le comité technique qui conseille.

Le cycle de modélisation consiste à modéliser la connaissance au travers des différents types de modèles qui ont été présentés. La modélisation s'effectue par l'interview d'experts ou l'analyse de documentation (avec la présence de personnes du domaine). Cette phase s'achève par la constitution d'un livre de connaissances regroupant l'ensemble des modèles produits. Ce livre fournit un point de départ indispensable à tout projet opérationnel de traitement de ces connaissances.

Le schéma d'orientation doit permettre d'articuler la modélisation MKSM avec la partie opérationnelle du projet. Le schéma d'orientation possède trois niveaux distincts :

- Un niveau stratégique permettant de définir des objectifs de la gestion du patrimoine de connaissances.

- Un niveau tactique établissant les actions à mener (en spécifiant leurs objectifs spécifiques, les sources de connaissances, les résultats attendus, leurs conséquences, leur environnement) afin d'atteindre les objectifs.
- Un niveau d'étude des risques des différentes actions à mener classant les risques selon six axes : les enjeux externes, les enjeux internes, les facilités d'accès aux connaissances, la criticité des connaissances, l'intérêt des résultats possibles et l'intégration à l'environnement.

#### **4. La méthodologie REX**

La méthodologie REX (acronyme de retour d'expérience) [12] est une méthode de gestion de connaissances et plus particulièrement de l'expérience mise au point par CEA (Commissariat d'Énergie Atomique). À l'origine son propos était de sauvegarder les connaissances acquises lors de la conception de réacteurs nucléaires. De cet objectif sont nées une méthode et un outil logiciel qui constituent aujourd'hui une solution utilisable par toute entreprise qui considère l'expérience acquise lors de ses activités comme un capital à conserver et à exploiter. Cette méthode et les outils associés permettent de recueillir l'expérience à partir de documents ou auprès de spécialistes puis de l'organiser afin de la mettre à disposition des équipes actuelles et futures.

REX se fonde sur trois étapes [68] :

- L'analyse des besoins et l'identification des sources de connaissances de l'organisation. Durant cette étape il s'agit de spécifier et de dimensionner le système de gestion des connaissances qui sera mis en place et permettre d'identifier les spécialistes du domaine.
- La construction d'éléments de connaissances à partir des documents, base de données, ou interviews.
- La mise en place puis l'exploitation du système de gestion des connaissances créé.

#### 4.1 Les éléments de connaissances

Le principe de base de la méthodologie REX consiste à composer des éléments de connaissances (EC) relatifs à une activité et qui, une fois restitués, pourront être valorisés par les utilisateurs. L'ensemble de ces éléments d'expérience sont stockés dans ce qui est appelé une mémoire d'expérience. Concrètement un élément d'expérience est un texte reflétant un savoir ou un savoir-faire. Trois types d'éléments de connaissances ont été définis :

- L'ECD (Elément de Connaissance documentaire) qui est produit à partir du fond documentaire et correspond au résumé d'un document .
- L'EEX (Elément d'Expérience) qui renvoie à l'expérience acquise par une personne de l'entreprise et qui est formalisée au cours d'un entretien.
- L'ESF (élément de savoir-faire) qui renvoie au savoir-faire acquis par une personne en participant à une activité particulière

Les EC sont construits principalement à l'issue d'interviews de spécialistes de l'organisation, de l'analyse des documents et de l'interrogation de base de données existantes. Un EC est typiquement constitué des éléments suivants : une entête, une description ou corps, une liste de références. L'entête comporte lui-même : un titre, un origine (le nom des experts interrogés), l'auteur, la date d'émission, une description du domaine et du contexte de validité. Le corps quant à lui est composé de trois parties : une description neutre d'un fait, une opinion propre et des commentaires, des recommandations.

#### 4.2 La mémoire d'expérience

Les éléments de connaissances constitués sont organisés dans une mémoire d'expériences de façon à être facilement réutilisables. Celle-ci comporte en outre un réseau terminologique et un modèle descriptif permettant de présenter le vocabulaire et les points de vue utilisés dans l'entreprise.

Le modèle descriptif permet de représenter les différents points de vue identifiés dans l'entreprise. Chaque point de vue est représenté par un réseau sémantique constitué d'objets ou de concepts liés entre eux par des liens de type : ensemble/élément, général/spécifique, proximité/évolution.

Le réseau terminologique quant à lui décrit le vocabulaire du domaine considéré. Il est également représenté par un réseau sémantique constitué d'objets (pouvant être des mots ou des phrases nominales) reliés entre eux par des relations sorte-de et concerne son principal objectif qui est de permettre des requêtes proches de la langue naturelle.

### **4.3 Le système de gestion des connaissances**

La méthodologie REX propose un outil informatique permettant à la fois de mémoriser et d'accéder aux éléments d'expérience. L'interface permet de consulter la mémoire d'expérience au moyen de requêtes en langage naturel. Le système retourne ensuite un ensemble d'objets (définis dans le réseau terminologique) et les EC qui leurs sont rattachés. REX est donc une méthodologie complète permettant l'analyse, la capitalisation et l'opérationnalisation des connaissances relatives à l'expérience .

## **5. La méthodologie Cygma**

CYGMA (Cycle de vie et Gestion des métiers et des Applications) [54] a été définie par la société KADE-TECH.. CYGMA prévoit 6 catégories de connaissances industrielles : connaissances singulières, terminologiques, structurelles, comportementales, stratégiques et opératoires. La méthode permet, en se basant sur ces catégories de définir des référentiels métiers appelés « Bréviaire de connaissances de filière métier » et de bases de connaissances, exploitables par des algorithmes de raisonnement déductif. Ces bases de connaissances sont appelées AMI (Application Métiers Industrielles ou Assistants métier de l'Ingénieur).

La méthode CYGMA préconise des entretiens avec les experts et une étude de la documentation de l'entreprise afin de définir un « bréviaire de connaissances ». Ce bréviaire sera ensuite validé avec les experts. Les connaissances dans ce bréviaire sont structurées en quatre documents : le glossaire métier, le livret sémantique, le cahier de règles et le manuel opératoire.



## 5.1 Le glossaire métier

le glossaire métier contient :

- *Des connaissances singulières* ; recueil de cas particuliers, apportant des éléments de définition des limites du domaine. Dans chaque cas, l'artefact ainsi que son développement sont décrits
- *Des connaissances terminologiques* ; sous forme de listes alphabétiques de termes utilisés dans le domaine métier. Chaque élément du vocabulaire est décrit dans le glossaire par sa définition, sa traduction, sa source et des références

## 5.2 Le livret sémantique

Dans le livret sémantique sont définis les connaissances structurelles. ces connaissances sont décrites sous forme de :

- *Connaissances ontologiques* ; une organisation hiérarchiques en classes d'objets des connaissances terminologiques. Ces classes sont organisées ainsi, grâce à des opérateurs logiques comme ET et OU, des valeurs booléennes VRAI ou FAUX et des listes énumérées de classes.
- *Connaissances factuelles*, un ensemble d'instances (« Base de Faits ») des classes d'objets . Une classe factuelle peut être définie explicitement avec une valeur ou implicitement . dans ce dernier cas , la valeur sera générée ultérieurement.
- *connaissances faits initiaux*, un sous ensemble des connaissances factuelles explicites, définissant le problème à résoudre.
- *Connaissances buts initiaux*, un sous ensemble de connaissances factuelles implicites, décrivant la solution du problème à résoudre.

Le livret sémantique est décrit sous forme d'un arbre mettant en évidence les opérateurs logiques qui existent entre les classes d'objets.

### 5.3 Le cahier de règles

Le cahier de règles comportant des connaissances comportementales qui sont définies par :

- Connaissances d'intégrité, ensemble de contraintes associées à une ou plusieurs propriétés d'une classe d'objets. Ce type de contraintes peut aussi mettre en relation plusieurs classes d'objets.
- Connaissances existentielles, ensemble de règles détectant l'existence d'un objet métier.
- Connaissances synthétique, ensemble de connaissances (définies sous forme de règles de production permettant d'écrire des faits synthétiques).

Le cahier des règles est défini d'une façon textuelle dans un document en langue naturelle. Chaque règle est définie suivant une fiche dont l'entête met en avant : le projet, la référence, la date de l'avant dernière et de la dernière modification, le type de la règle et le titre, et dont le corps permet de décrire l'historique (raison de la dernière modification), les sources (auteurs de la règle, document, norme, service), la genèse de la règle, son objectif, sa description et les remarques ainsi qu'un schéma permettant de l'illustrer.

### 5.4 Le manuel opératoire

Le manuel opératoire, rassemble : les connaissances stratégiques ou méta connaissances, qui permettent l'emploi optimisé des connaissances structurelles et comportementales et les connaissances opératoires, qui sont représentées sous forme d'enchaînement d'activités décrivant le processus de résolution. Le manuel opératoire comporte trois parties :

- La carte décrivant l'enchaînement des phases du processus de résolution. Les diagrammes SADT sont utilisés pour représenter ces phases.
- Le parcours décrivant graphiquement les multiples chemins possibles d'enchaînements des étapes.

- Les étapes , mettant à jour les agents (opérateurs, règles, outils d'assistance, etc), les actions mises en oeuvre et les moyens utilisés (succession de choix, référence aux règles, nom d'outils,etc).

## **6. Comparaison des différentes méthodes de capitalisation des connaissances**

Selon la comparaison des méthodes de capitalisation des connaissances dans [54] et [68] on a aboutit à une synthèse de ces comparaisons selon les critères les plus importants, illustrée dans le tableau 01 .

**Tableau 01 : Tableau De Comparaison Des Méthodes De Capitalisation Des Connaissances**

<b>Méthodologie</b>	<b>Connaissances et problèmes traités</b>	<b>Types de mémoire et modes de représentation</b>	<b>Niveau de formalisation</b>	<b>Outils</b>	<b>Méthodes de recueil et sources de connaissances</b>	<b>Complexité de mise en œuvre</b>
<b>CommonKads</b>	Objectif de création de SBC : surtout connaissances expertes mais aussi sur l'organisation et sur le futur SBC	Mémoire d'activité : modèles de connaissances.	Fort, les modèles KADS permettent de traiter les problèmes de représentation et de mise en œuvre des connaissances : un SBC peut découler directement de la modélisation .	KADS-Work-bech, KADS-tools, Open-KADS, Cokace	Entretiens avec experts.	Importante, complexité des modèles de représentation (formalisation importante), difficultés de mise en place d'une solution opérationnelle.
<b>MKSM</b>	Objectif de création d'une mémoire d'entreprise (système organisationnel, activités, savoirs et savoir-faire des experts)	Mémoire d'activité : modèles de connaissances .	Moyen les connaissances sont toutes modélisées sous forme d'un livre de connaissances) mais les modèles ne permettent pas une opérationnalisation immédiate.	L'outil MKSM	Entretiens avec experts et analyse des documents.	Moyenne, nécessite la collaboration de nombreux acteurs de l'entreprise, tâche de modélisation longue mais souple.
<b>REX</b>	Objectif de gestion de retour d'expérience .	Mémoire individuelle d'expérience.	Faible seulement une ontologie de domaine et une simple structuration des éléments de connaissances sous forme de fiches.	L'outil REX	Entretiens avec experts et analyse des documents.	Moyenne l'effort de capitalisation est limité car peu de formalisation. Une solution opérationnelle est intégrée.
<b>Cygma</b>	Objectif de gestion des connaissances d'un métier.	Mémoire métiers : référentiels métiers.	Fort modélisation de différentes sortes de connaissances : structurelles comportementales, opératoire, ontologique, terminologique.		Entretiens avec experts et analyse des documents.	Moyenne, une phase de formalisation longue, mais l'opérationnalisation est directe.

## **7. Conclusion**

A partir de la synthèse des comparaisons des différentes méthodes de gestion de connaissances, nous avons conclu que la méthodologie CommonKADS s'adapte bien à nos objectifs. Ainsi nous nous basons sur le modèle générique de modélisation de la tâche de diagnostic pour la conceptualisation du système. Le modèle obtenu est une représentation du système à base de connaissances qui constitue le noyau du système de gestion de connaissances visé.

# Le problème de diagnostic de pannes



## Chapitre 3

*L'*objectif de ce chapitre est de présenter la problématique de l'étude avec une méthode de résolution du problème.

## 1. Introduction

Le domaine de diagnostic et de maintenance des équipements dans les stations de compression de gaz nécessite un savoir-faire technique spécifique et une expertise, cette dernière est acquise sur le terrain à travers les études et les expériences par quelques spécialistes. Cependant, ces spécialistes ne sont pas disponibles à tout moment et dans chaque station.

Recueillir, préserver et exploiter les connaissances de diagnostic de pannes dans ce domaine est donc un besoin important ressenti comme une nécessité ou au moins une aide importante dans l'organisation qui centralise en effet une expertise et un retour d'expérience importants sur certains équipements. Ce besoin nous a conduit à fixer deux objectifs principaux :

- 1- Formaliser une démarche générale pour capitaliser les connaissances expertes de diagnostic et de maintenance et pour concevoir des aides au diagnostic exploitant des connaissances expertes.
- 2- Développer un environnement de gestion de connaissances de diagnostic, servant de support à cette démarche et offrant les services suivants :
  - a)- Recueil et gestion de connaissances de diagnostic et de maintenance de matériels.
  - b)- Consultation et analyse de connaissances.
  - c)- Exploitation des connaissances capitalisées.

Cet environnement doit fournir une assistance automatique pour le diagnostic de matériel.

## 2. Présentation du système réel

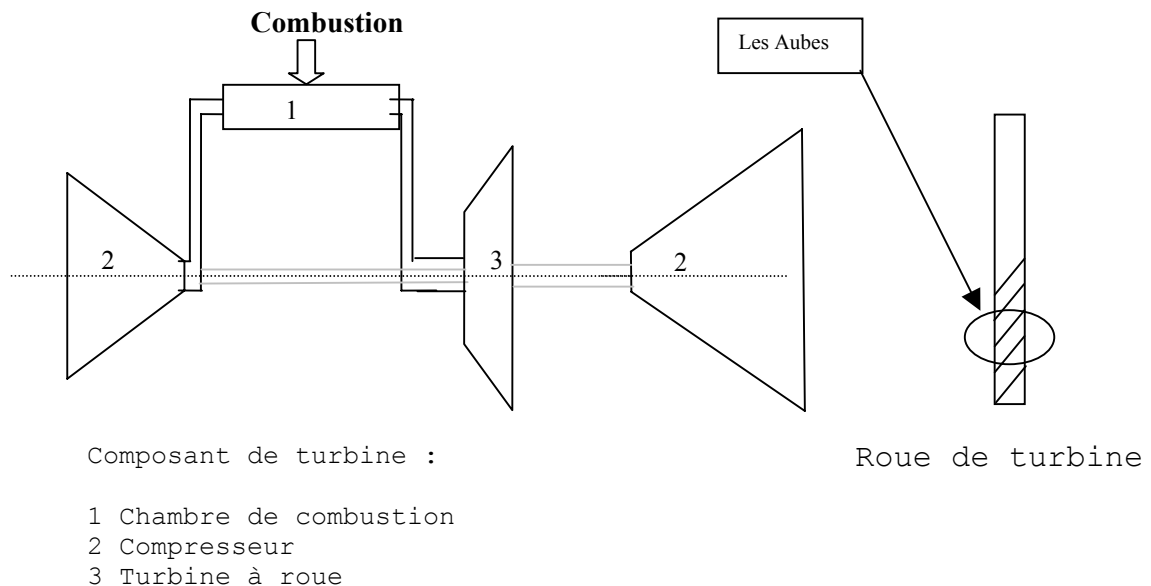
Notre procédé est une station de compression de gaz parmi d'autres stations similaires. Plusieurs composants tel que : des turbines, compresseur, ballons des Torches etc. sont contenus. Nous nous intéressons ici au contrôle de la turbine de gaz..

## 2.1. La turbine

C'est un moteur rotatif qui convertit l'énergie d'un courant d'eau, de vapeur ou de gaz en énergie mécanique. Plus généralement, c'est un organe permettant la détente d'un fluide en recueillant son énergie sous forme mécanique.

L'élément de base d'une turbine est une roue ou un rotor à ailettes, à hélice, à lames, à aubes ou à augets disposés sur sa circonférence, de façon que le fluide en mouvement exerce une force tangentielle qui fait tourner la roue et lui confère de l'énergie. Cette énergie mécanique est ensuite transmise par un arbre qui fait tourner un moteur, un compresseur, un générateur ou une hélice.

La figure suivante montre une turbine avec ces différents composants.



**Fig 3.1 :** La structure des composants d'une turbine

## 2.2 Le compresseur

C'est une machine entraînée par la turbine à gaz, il est de type *Centrifuge* (i.e. transforme l'énergie cinétique en pression).

Le compresseur rotatif est utilisé pour obtenir des pressions faibles et moyennes. Il est en général constitué d'une roue mobile à aubes ou d'un rotor qui tourne dans un

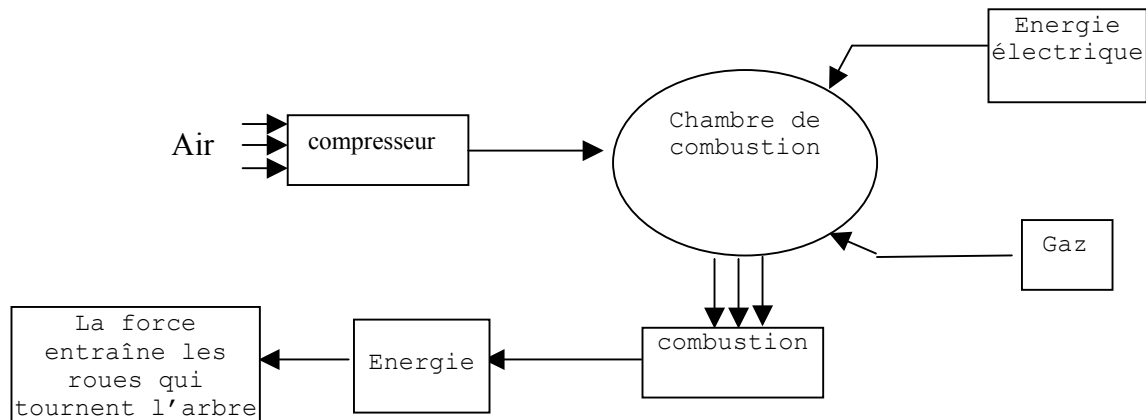


boîtier circulaire hermétique. L'air est aspiré au centre de la roue et est accéléré par la force centrifuge des aubes.

### 2.3. Chambre de combustion

L'air atmosphérique, aspiré par le compresseur axial, est comprimé puis refoulé dans la chambre de combustion où est introduit le combustible, le mélange désiré (air comprimé et gaz sous pression) est obtenu. Une étincelle fournie par une bougie provoque la combustion. La chaleur produite par la chambre de combustion et l'énergie dégagée par les produits de la combustion sont dirigées vers la turbine.

Le schéma suivant figure (3.2) explique ce principe :



**Fig 3.2 : Le principe de fonctionnement de la turbine.**

## 3. Le Problème de diagnostic

Le diagnostic est une tâche de la vie quotidienne. Il revêt une importance capitale pour les systèmes industriels mais aussi pour tous les systèmes dynamiques. D'une manière générale, la tâche de diagnostic consiste, à partir des observations faites sur un système et des connaissances sur le fonctionnement nominal ou symptomatique de celui-ci, à vérifier l'existence ou non de dysfonctionnement. Il permet également de détecter, d'identifier, de localiser et parfois de réparer les comportements et les composants à l'origine des anomalies.

Le diagnostic automatique est interdisciplinaire, à la fois dans les techniques utilisées et dans les applications. Historiquement les premiers travaux remontent au

début des années 70 [61] avec notamment de nombreuses applications en médecine, en électronique, en chimie, etc. Dans ces applications, le problème consiste à identifier un ensemble d'hypothèses sur les sources de fautes ou sur les composants fonctionnant en mode dégradé ou en panne. Le but de diagnostic est de couvrir le maximum de fautes.

Il existe différentes approches pour le diagnostic, parmi lesquelles : l'approche à base d'expertise ou d'heuristiques et l'approche à base de modèles. L'approche à base d'heuristique repose sur des connaissances de surface, c'est essentiellement des connaissances d'experts humains et des résultats d'études statistiques. L'approche à base de modèles s'appuie sur un modèle explicite du système à diagnostiquer, celui-ci permet de faire des prédictions sur le comportement du système.

Le diagnostic utilisant le raisonnement sur les connaissances de surface est intéressant dans les domaines où les systèmes changent rarement ou lentement de comportement et de configuration. C'est la méthode qui est la plus utilisée actuellement pour la réalisation de système de diagnostic industriel fonctionnant concrètement sur des applications réelles.

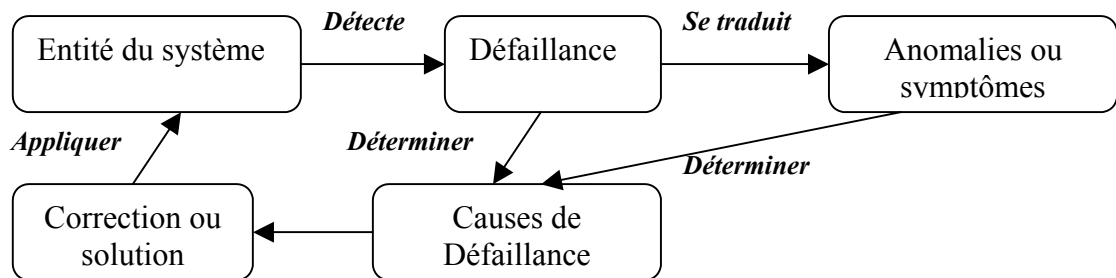
Les approches à base de modèles permettent de décrire les connaissances structurelles du système ainsi que ses modes de fonctionnement nominal, sous forme d'un modèle explicite. Le principe peut être vu comme la comparaison entre le comportement prédit par le modèle du système et les observations réelles faites sur celui-ci dans son environnement. Tout écart entre les deux fonctionnements implique la présence de dysfonctionnements que le processus de diagnostic doit traiter.

#### **4. Les Concepts associés**

Les principaux concepts auxquels nous faisons référence sont généralement inspirés de la terminologie adoptée dans le diagnostic industriel [61]

- **Diagnostic**<sup>(1)</sup> : Le diagnostic correspond à la détermination de la (des ) cause(s) d'un trouble fonctionnel , à partir de symptômes ou d'observations.
- **Défaillance** : Une défaillance correspond à une cessation de l'aptitude d'une entité à accomplir une fonction donnée, une entité étant un élément ou un composant du système étudié.
- **Symptôme ou anomalie** : Un symptôme est un phénomène qui révèle un trouble fonctionnel ou défaillance. Les symptômes ou anomalies correspondent à l'observation d'une différence entre l'état de fonctionnement normal du système et l'état observé
- **Cause** : Les causes constituent les sources des difficultés ou des défaillances. Ce sont celles qui font que les choses ne se déroulent pas comme attendues .
- **Correction** : Ensemble d'actions permettent de rétablir une entité dans son état de bon fonctionnement.

La figure suivante (3.3) reprend les concepts précédemment définis :



**Fig 3.3** Concepts d'un système de diagnostic [60]

<sup>(1)</sup> Le terme diagnostic vient du grec, il est composé de dia qui signifie par et de gnosis qui signifie connaissance , cette définition montre l'importance de l'A.C dans le déroulement du diagnostic.

## 5. Le diagnostic de pannes : méthode de résolution du problème

Les tâches génériques de la bibliothèque CommonKADS sont classées en trois grandes classes :

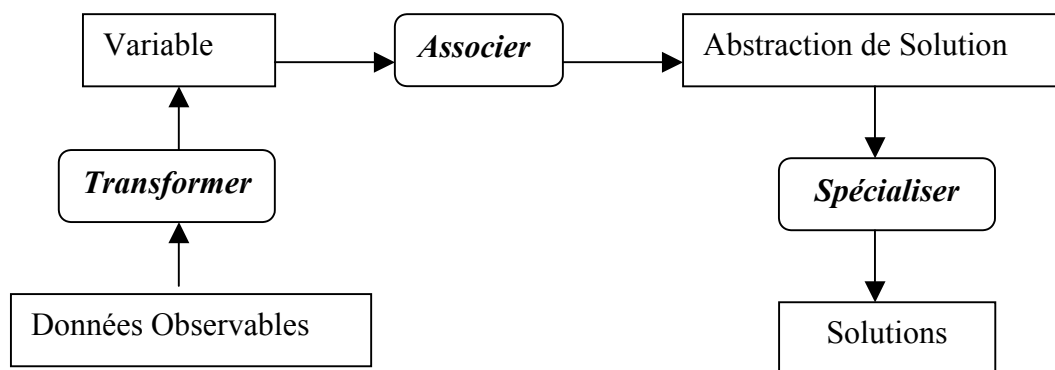
- 1- Les tâches d'analyse.
- 2- Les tâches de modification de systèmes
- 3- Les tâches de synthèse

Pour les tâches d'analyse, la structure interne du système analysé est connue et la solution consiste en l'identification d'une propriété courante du système ou la « prédiction » d'un état de ce dernier. Pour la deuxième catégorie il s'agit des actions modifiant l'état courant du système, mais ne lui ajoutant pas de nouvelles fonctionnalités. La dernière classe consiste à synthétiser une structure, étant donnée un ensemble de contraintes et de spécifications .

Notre problème de diagnostic est un des tâches les mieux connues des tâches d'analyse. Comme le diagnostic consiste à trouver une « faute » dans un système, une « faute » étant la « négation » d'une fonction ou d'un composant correct. Comme l'univers des « négations » est bien plus grand que la description d'un système correct, la recherche des fautes est souvent basée sur l'expérience (priorité des anomalies arrivant fréquemment) ce type de connaissances est de nature heuristique et c'est sur ce modèle de diagnostic c'est à dire diagnostic par classification heuristique qu'on s'est basé pour notre SBC , et comme on dispose d'un modèle de décomposition structurelle du système ainsi qu'un modèle causal de pannes est réalisé, il est possible d'utiliser une méthode de diagnostic systématique qui repose sur une localisation structurelle ou une localisation causale de pannes . Les associations heuristiques peuvent d'abord être utilisées pour focaliser les causes d'anomalie les plus probables puis ensuite la localisation pour compléter, justifier, valider ou apporter des preuves supplémentaires . Chacune de ces trois approches pourrait prétendre isolément résoudre le problème, pour notre système on cherche au contraire à les faire coopérer, la stratégie de coopération commence par utiliser les connaissances heuristiques disponibles , si cette approche heuristique ne suffit pas à résoudre

complètement le problème, le système met alors en œuvre ses connaissances causales pour tenter de construire une explication des phénomènes. Enfin en dernier ressort le système raisonnera sur la structure pour tenter d'identifier la cause du problème..

Le diagnostic selon un modèle heuristiques est donné par la figure (3.6). Il s'appuie sur la description de situations élémentaires auxquelles sont associés des éléments de solution.



**Fig 3.4** *Modèle de diagnostic basé sur des heuristiques* [53]

Pour le diagnostic heuristique, il s'agit de faire une abstraction des données observables pour obtenir les valeurs de paramètres du système en fonctionnement puis faire associer la solution qui correspond au cas de panne.

Pour le diagnostic systématique, la structure d'inférence commune à la localisation structurelle et à la localisation causale est donnée par la figure 3.7.

La structure d'inférence représentée par cette figure est commune à la localisation structurelle et à la localisation causale .

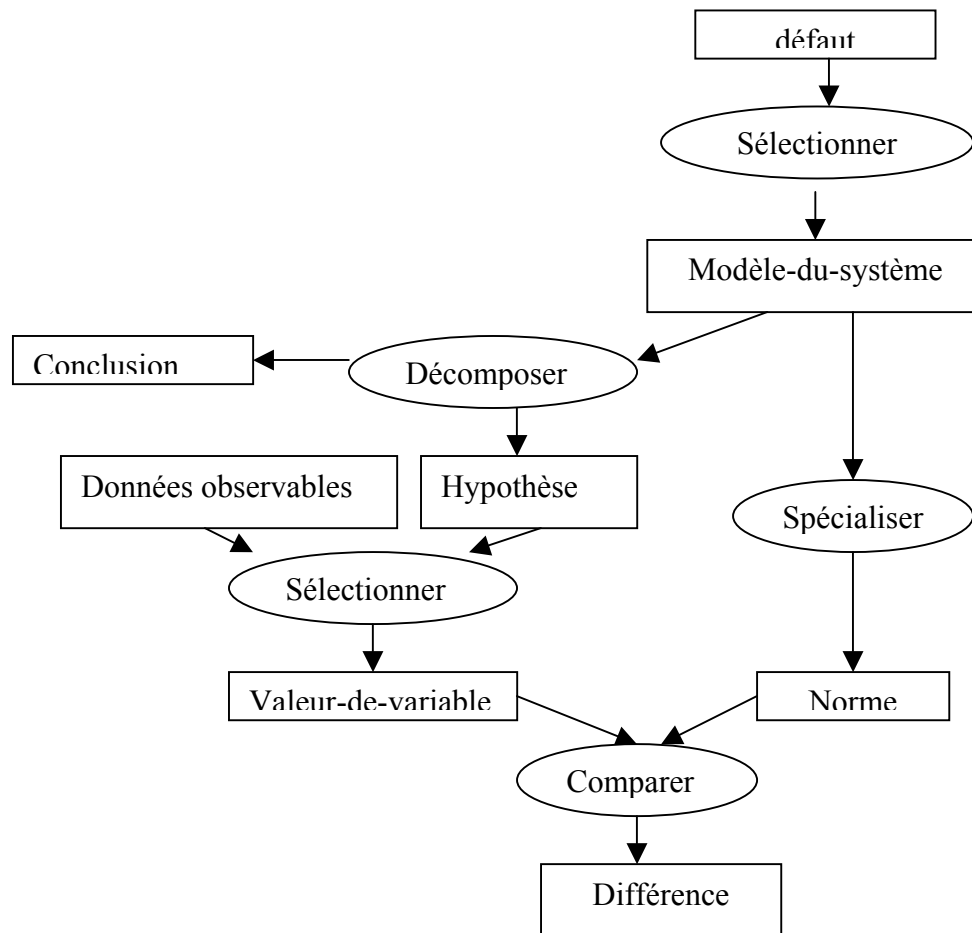
#### - Localisation structurelle

Cette tâche est possible si l'on dispose d'un modèle du système structuré par des liens type « consiste-en » ou « partie-de », si les composants sont testables indépendamment du reste du système, si l'on dispose des spécifications détaillées du comportement du système ou des valeurs de sortie.

#### - Localisation causale

Cette tâche est utilisable s'il est possible d'obtenir des données sur une part significative des états impliqués dans le modèle et s'il est possible d'expliquer la

fonction d'un système par des relations causales (ex : relation mécanique ou électrique dans un moteur). Dans la réparation de machine électronique, les deux types de localisations alternent souvent car un seul point de vue est généralement insuffisant pour éliminer des hypothèses.



**Fig 3.5** Structure des inférences de la tâche de diagnostic systématique[53]

Selon ce modèle, suite à un défaut la première inférence consiste à sélectionner le modèle du système qui suppose contenir au moins un composant défectueux. Ce modèle est structuré par des liens « partie de », puis il s'agit de décomposer le système de manière itérative et lui faire associer des hypothèses de défaillance pour les parties susceptibles d'être à l'origine des anomalies. Par la suite pour chaque hypothèse, il faut sélectionner une valeur de variable parmi toutes les sorties observables et la comparer avec la valeur attendue (norme), si une différence est

trouvée alors cet état est fautif et la panne est ainsi localisée, sinon l'hypothèse suivante sera vérifiée.

## **6. Conclusion**

Etant donnée que la problématique de l'étude est bien identifiée à travers les concepts et les principes de base associés, ainsi que la méthode de travail qui a été sélectionnée parmi un ensemble de méthodes puisqu'elle s'adapte le plus à nos besoin, nous allons procéder à la modélisation du système visé puis son implémentation.

**Partie II**  
**Conception & Réalisation**



# Analyse & conception



## Chapitre 4

*D*ans ce chapitre nous présentons la démarche générale de capitalisation des connaissances, ensuite nous nous intéressons à la manière dont les connaissances peuvent être modélisées. Cela nous conduira en particulier à présenter en détail les différents modèles utilisés pour représenter le système.

## 1. Démarche adoptée

En suivant le cycle de vie des systèmes de gestion de connaissances. La première phase consiste à délimiter le champ de l'étude en identifiant les besoins des utilisateurs qui se résument dans les points suivants :

- Disposer d'une solution technique efficace qui assure la capitalisation des connaissances du diagnostic, en prenant en compte la diversité de ces connaissances c'est à dire (sur des documents, des experts, des Bases de données ,.....etc)
- Permettre l'accès à ce capital de connaissances aux employés, pour les assister dans leurs tâches de diagnostic (aide au diagnostic).

Après avoir cerner le domaine on procède à la construction du système de gestion de connaissances. Une phase préliminaire d'acquisition de connaissances est indispensable. Elle concerne principalement :

- Extraction des connaissances à partir des bases de données qui constituent une partie importante des connaissances du domaine.
- Analyse des documents et extraction des connaissances manuellement ou de manière semi automatique des documents ce qui donne une partie des connaissances du domaine et également des connaissances comportementales
- Compléter la partie des connaissances comportementales ou de raisonnements à l'aide des interviews avec les experts.

Une première formalisation des connaissances acquises des différentes sources consiste à :

- 1- Une description des connaissances terminologiques du domaine (termes, définitions, traduction et sources).
- 2- Une classification des connaissances terminologiques sous formes d'objets et de classes . avec des relations du genre *partie-de* , *est-un* et autres et quelques contraintes sur les propriétés des objets .
- 3- Formalisation du savoir faire sous forme de règles ainsi qu'une description de l'enchaînement des activités relatives à la tâche de diagnostic.

Cette phase permet d'aboutir à :

- une ontologie de domaine
- une base de règles et une méthode de résolution du problème

## 2. Acquisition des connaissances

### 2.1. Comprendre l'activité

Plusieurs techniques existent pour le recueil des connaissances, héritées principalement du travail effectué en AC. Nous avons essentiellement utilisé trois de ces techniques : les entretiens semi-structurés, l'observation et l'analyse de documents.

Tout d'abord l'analyse des documents théoriques nous a permis d'avoir une vision formelle de l'activité de diagnostic étudiée, trois types de documents ont été étudiés :

- Une partie des documents consiste à des explications textuelles des composants et instruments installés dans la station de compression de gaz ainsi que des schémas descriptifs des circuits de la station étudiée.
- La deuxième partie décrit l'ensemble des alarmes signalant les pannes du système ainsi que des schémas de propagation de pannes. ce qui permet de bien localiser la source du dysfonctionnement.
- La troisième partie identifie les causes concernant les turbines par exemple nous avons identifié les causes de dysfonctionnement du premier degré c'est-à-dire le bruit, les appoints d'huile et d'eau indiquant une consommation anormale, la détection de gaz à l'intérieur des réservoirs, les températures, etc.
  - *Détection des bruits* : Un bruit anormal pendant le fonctionnement peut provenir d'un frottement de pièces mécaniques, fuites de gaz ou d'air. Les bruits métalliques spécifiques (grincement) dans les aubes et les étanchéités sont les signes. Ces frottements peuvent être causés par une déformation des pièces.
  - Appoints d'huile ou d'eau. Une consommation excessive d'huile de lubrification est significative d'évolution de réglage ou d'usure de labyrinthe d'étanchéité ou d'une mauvaise circulation de barrage

- Contrôle de présence de gaz dans les réservoirs d'huile .Une présence d'air ou de gaz à l'intérieur des réservoirs de lubrification est un signe de mauvaise étanchéité au niveau du circuit d'air de refoulement du compresseur axial turbine ou d'une défaillance au niveau du système étanchéité du compresseur de charge (machine entraînée).
- La température : Le fonctionnement des systèmes de combustion est évalué par l'irrégularité de la température des gaz brûlés .
- La quatrième partie concerne « le journal » , document de saisie et de maintenance des procédures. L'analyse du contenu de ces journaux nous a permis de capturer le savoir-faire des experts où leur méthodes de diagnostic de pannes selon une approche heuristique .Le journal est rempli sur site suite à un cas de pannes et permet de synthétiser les causes probables de la panne , les solutions apportées et les recommandations à faire.

La lecture ,l'exploitation et la compréhension du procédé n'ont été possibles qu'avec la collaboration des experts du domaine qui détiennent une masse plus importante des connaissances , plusieurs interviews ont été réalisés de manière orientée .

Enfin l'observation des spécialistes du domaine pendant leur activité nous a permis d'avoir un point de vue sur la pratique du travail et de se rendre compte de l'activité en temps réel.

Cette première phase d'analyse a permis d'identifier les concepts de base manipulés dans le domaine, de déterminer quelques règles de travail et d'en tirer les besoins dans ce domaine pour la capitalisation des connaissances.

### **3. Modélisation**

Dans le but de capitalisation de connaissances on doit passer à une formalisation des connaissances recueillies pour pouvoir les stocker, il s'agit soit d'organiser ou de réorganiser des connaissances explicites, soit de formaliser des connaissances implicites qui ont été localisées à l'étape précédente.

Une première classification des connaissances recueillies nous a permis de distinguer divers types de connaissances, à une extrémité se trouvent les connaissances du domaine dites statiques à l'autre extrémité se trouvent les connaissances de contrôle sur les premières, dans le but de résoudre les problèmes posés.

Dans notre cas on a appliqué plusieurs principes de formalisation. On a fixé comme but la construction d'une base de connaissance pour permettre la mise en place d'un SBC pour l'aide au diagnostic de pannes pour cela on a procédé initialement à l'élaboration du modèle conceptuel du SBC qui est une description à un niveau abstrait, indépendant de l'implantation, des objets du domaine d'expertise et leurs relations (le modèle du domaine) ainsi que la méthode de diagnostic à mettre en œuvre (le modèle de raisonnement) et les liens entre ces connaissances.

Notons qu'il existe une partie importante de ces connaissances manipulées et non formalisables. Il est important de prendre en compte des autres formes de connaissances qu'on a pu découvrir qui sont les schémas descriptives des différents systèmes de la station de compression. Ainsi que les documents eux mêmes qui demeurent toujours une référence très importante.

Une des raisons importantes pour la modélisation des connaissances est de pouvoir présenter et rendre les connaissances techniques, disponibles au delà de l'ensemble courant des experts.

Par conséquent, on veut réaliser ce qui suit :

- Acquérir les connaissances détenues par les experts et les mettre dans une forme lisible et opérable (explicitement les connaissances tacites)
- Partager les connaissances entre différents utilisateurs.
- Les réutiliser dans les applications à base de connaissances
- Les maintenir et les faire évoluer.

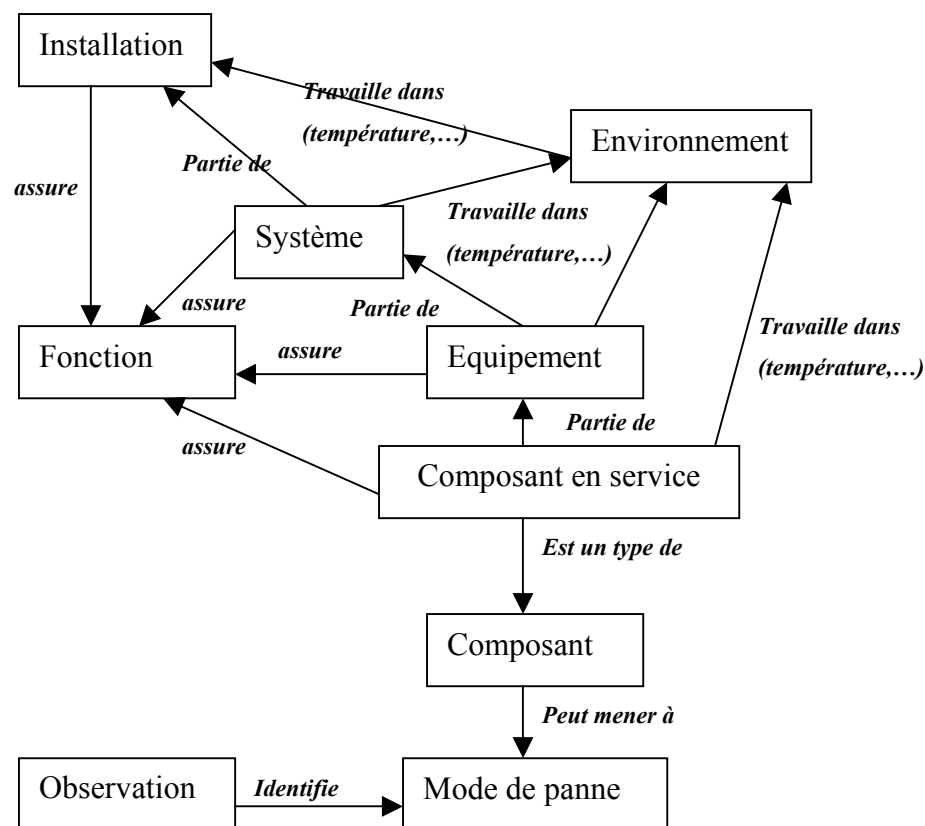
Les grandes catégories de connaissances à intégrer dans le système ayant été identifiées, nous présentons dans ce qui suit comment ces connaissances ont été modélisées. Plus exactement nous présentons les modèles obtenus pour chacune de ces catégories.

## 4. Approche de modélisation

L'approche de modélisation adoptée repose sur les principes suivants :

- Séparation entre modélisation/expression des connaissances (description du domaine) et l'opérationnalisation des connaissances (knowledge processing) les modèles permettent l'expression du domaine de connaissances et les services qui utilisent les connaissances capitalisées pour produire les résultats spécifiques (diagnostic, décision, etc)
- Modélisation formelle basée sur une description entité/relation du domaine pour permettre l'expression des connaissances.
- Décomposition du problème basée sur une description multiples des modèles comme moyen pour réduire la complexité du problème, pour faciliter la compréhension des modèles, leur réutilisation et leur validation .

### 4.1 Modèle de décomposition structurelle et fonctionnelle des composants

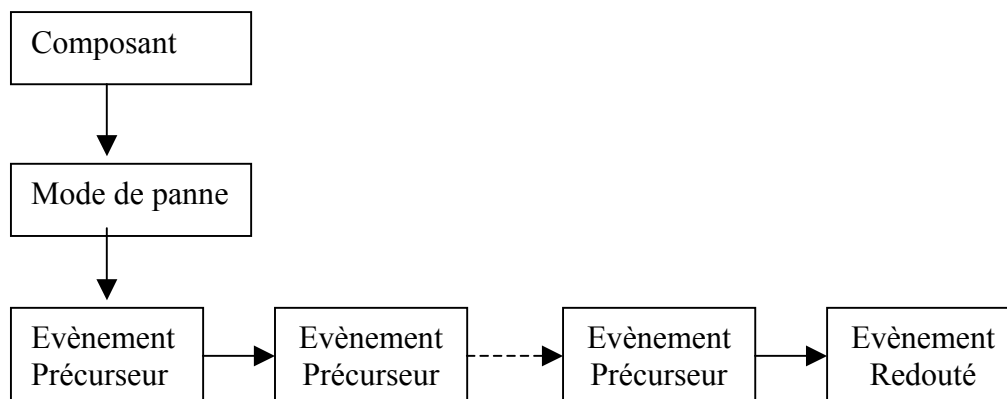


**Fig 4.1** Une vue de décomposition structurelle et fonctionnelle du système

Ce modèle représente des connaissances structurelles, elles décrivent la structure fonctionnelle du matériel .A chaque entité fonctionnelle sont attachés les phénomènes la concernant . ce modèle est une représentation formelle des concepts de base manipulés du procédé industriel visé, tout d'abord il s'agit de considérer différentes stations chacune correspond dans notre modèle à une *installation* .

Dans chaque installation un ensemble de circuits peuvent être distingués appelé *système* selon notre modèle exemple : système huile d'étanchéité , système huile de graissage , système huile de lubrification, etc. Chaque système à son tour peut être décomposé en *équipements* et composants élémentaires : exemple : pompes, filtres , soupapes, réservoirs ,etc. Ainsi à chaque installation , système, équipement ou composant on fait associer sa *fonction* où son rôle dans le procédé et également un ensemble de paramètres qui caractérise son environnement de travail .A chaque composant les modes de pannes relatives sont également représentés dans ce modèle.

#### 4.2. Le modèle causal de propagation de pannes



**Fig 4.2** *Aperçu général du modèle de propagation de pannes [43]*

Les connaissances représentées par ce modèle sont des connaissances causales qui décrivent des relations de cause à effet entre phénomène du domaine par exemple un évènement précurseur c'est un évènement qui précède et prédit un évènement redouté ce dernier est un évènement possible c'est à dire l'évènement qui suppose être l'origine de la panne .En pratique ces évènements correspondent au

déclenchement des alarmes qui permettent de localiser l'origine de la panne tout en tenant compte de leur mode de propagation, une alarme peut être déclenchée suite à l'activation d'autres alarmes ce qui traduit bien la succession d'un ensemble d'évènements précurseurs pour aboutir ou identifier l'évènement qui peut être une cause possible de la panne.

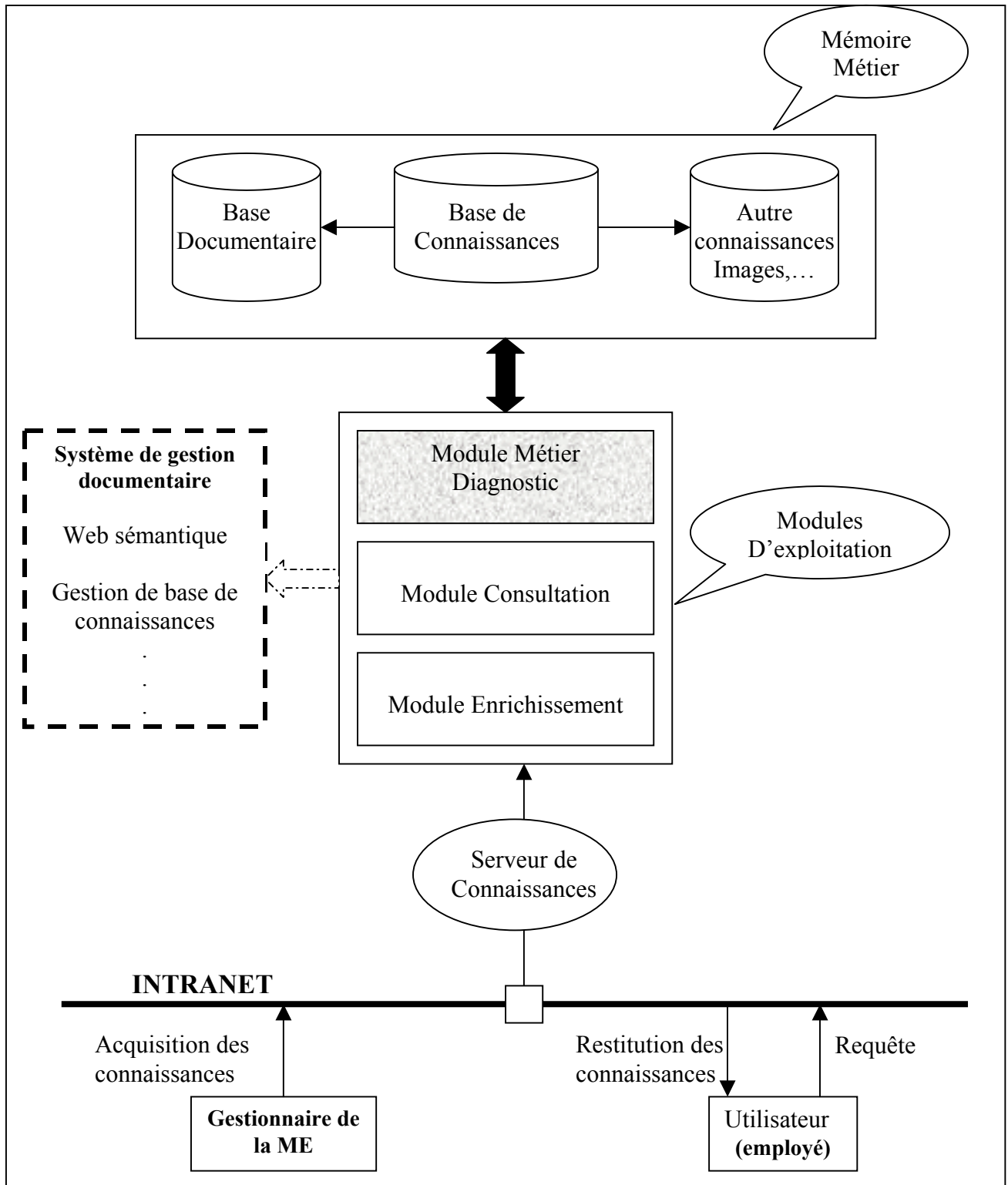
### **4.3 Le modèle de tâche**

la procédure de diagnostic appliquée consiste à procéder dans un premier temps à un diagnostic heuristique se basant sur la méthode de raisonnement de l'expert, ensuite il s'agit de sélectionner le modèle du système qui suppose contenir au moins un composant défectueux. Ce modèle est structuré par des liens « partie de », en appliquant une localisation structurelle de la panne sur la base du modèle de décomposition structurelle du système. Puis il s'agit de décomposer le système de manière itérative et lui faire associer des hypothèses de défaillance pour les parties susceptibles d'être à l'origine des anomalies. Par la suite pour chaque hypothèse, il faut sélectionner une valeur de variable parmi toutes les sorties observables et la comparer avec la valeur attendue (norme), si une différence est trouvée alors cet état est fautif et la panne est ainsi localisée, sinon l'hypothèse suivante sera vérifiée. La localisation causale est utilisée à la fin en se basant sur le modèle de propagation de pannes.



### 5. Architecture du système

Pour la valorisation des connaissances capitalisées il s'agit de permettre leur accès , diffusion et exploitation, pour cela on propose l'architecture suivante pour le système développé :



**Fig 4.3** Architecture Générale du système de gestion des connaissances

Le système développé est un système de capitalisation de connaissances adapté pour l'aide au diagnostic de pannes dans le système complexe de compression de gaz. Le système est composé de trois modules organisés autour d'une architecture particulière basée sur un serveur avec une interface web. le système est développé de façon entièrement modulaire , ce qui permet la réutilisabilité de certaines parties du développement. Les fonctionnalités de ce système sont les suivantes :

- **Le module métier diagnostic** : il permet de réaliser le diagnostic sur le système suite à une requête de l'utilisateur qui permet d'introduire l'état du système à travers l'activation dans une liste de l'ensemble des alarmes déclenchées .le système et à partir des inférences effectuées sur la BC pose une série de questions auxquelles l'utilisateur doit répondre selon les évènements observés , ce qui l'amène à une proposition de diagnostic final et un mode opératoire à suivre pour résoudre le problème. Pour argumenter la solution et pour assister réellement l'utilisateur dans son problème , à chaque objet ou tâche peut être associé un (ou plusieurs) documents (s) électroniques et qui seront proposés à l'utilisateur au besoin .
- **Le module consultation** : Il permet de comprendre comment fonctionne le système , par une présentation graphique avec des commentaires et des explications , permet également de consulter des cas de pannes et leur résolution ., en d'autre terme ce module est dédié à la formation des utilisateurs novices du système
- **Le module enrichissement** ce module est réservé aux experts pour permettre la mise à jour de la base de connaissances de la base documentaires , d'ajouter des liens des commentaires ,...etc ce qui assure l'évolutivité de la mémoire d'entreprise par un retour d'expérience suite à l'utilisation de ce système

## 6. Conclusion

Le passage de cette structure conceptuelle à une implémentation informatique est obligatoire pour tirer profit du système de gestion de connaissances développé. Pour cela le système s'organise sur la base d'un réseau intranet ; l'utilisateur final ainsi que le gestionnaire communique avec le système via des interfaces.

# Environnement de développement & Application



## Chapitre 5

*D*ans ce chapitre nous présentons les outils utilisés pour le développement de l'application ainsi que les principes de base pour l'implémentation du système développé.

## 1. Introduction

Ce chapitre sera consacré à l'expérimentation des modèles que nous avons développés au cours du chapitre précédent. Tout d'abord, nous commençons par représenter les outils et environnements utilisés, protégé2000 et Jess et nous présentons par la suite la méthodologie adoptée pour proposer des solutions informatiques.

## 2. Environnement de développement

### 2.1. Protégé 2000

L'origine de protégé 2000 est une petite application désignée pour le domaine médical. Cette application a évolué pour devenir un ensemble d'outils dont le but est plus général pour la construction des SBC (protégé , protégé II , protégé/win , protégé 2000 ), chaque génération de protégé a été testé avec des problèmes réels et utilisé pour la construction des SBC .La version la plus récente « protégé 2000 » incorpore le modèle de connaissances (OKBC) , s'exécute sur différentes plate-forme , et supporte une interface utilisateur facilement extensible et personnalisable et a été utilisé par plusieurs recherches [37] spécialement dans le domaine médical .

#### 2.1.1 Protégé 2000 et OKBC

Le partage et la réutilisation de connaissances est devenu un des buts primordial des communautés de recherches des SBC . L'interopérabilité entre systèmes de représentation des connaissances est une étape cruciale pour la réalisation de ce but . OKBC "The Open Knowledge-base Connectivity" facilite cette interopérabilité en fournissant un modèle de connaissances à base de frame et une riche et expressive API « Application-Programming Interface » pour interagir avec les serveurs des bases de connaissances .Actuellement un ensemble de systèmes de représentation de connaissances compatible –OKBC est disponible incluant Ontolingua , Loom et protégé 2000 [58]. En adoptant le modèle de connaissances OKBC , protégé 2000 bénéficie d'une grande expressivité des connaissances une sémantique propre au modèle théorique ainsi qu'une possibilité de réutilisation

### 2.1.2 Le modèle de connaissances de Protégé 2000

Le modèle de connaissances de protégé 2000 est à base de frames .Les frames sont les principaux blocs de construction d'une base de connaissance .Une ontologie sous protégé 2000 consiste en des classes, slots , facettes, et des axiomes .Les classes sont des concepts dans le domaine de discours . Les slots décrivent des propriétés ou des attributs de classes. Les facettes décrivent les propriétés des slots .Les axiomes spécifient des contraintes additionnelles . une base de connaissance sous protégé 2000 inclut l'ontologie et des instances individuelles de classes avec des valeurs spécifiques pour les slots.

### 2.1.3 Les Classes et Les Instances

les classes dans protégé 2000 constituent une hiérarchie taxonomique .si une classe A est une sous-classe d'une classe B chaque instance de A est aussi une instance de B. Par exemple, une classe qui représente vanne de sûreté est une sous-classe de la classe vanne, donc toute instance de la classe vanne de sûreté est également instance de la classe vanne. Protégé 2000 visualise la relation de sous-classes dans un arbre. Protégé supporte également l'héritage multiple: une classe peut avoir plus d'une super classe., la racine de la hiérarchie de classes dans protégé-2000 (et dans OKBC) est la classe :THING. Dans protégé 2000, les classes elles-mêmes peuvent être instances de classes, une méta classe est une classe dont les instances sont eux-mêmes des classes.

### 2.1.4 Les slots

les slots dans protégé 2000 décrivent les propriétés des classes et des instances, tel que le désignation d'un instrument , ou le débit d'une pompe, un slot lui-même est un frame .Dans protégé, comme dans OKBC les slots sont définies indépendamment de toute classe quand un slot est attaché à un frame dans l'ontologie, il décrit les propriétés de ce frame particulier par exemple, nous pouvons définir un slot désignation et l'attacher aux deux classes instrument et alarme pour représenter respectivement la désignation d'un instrument et la désignation d'une alarme .Quand un slot est attaché à un frame il peut avoir une valeur .Par exemple, le slot

désignation pour une instance spécifique d'alarme peut avoir la chaîne « L63HLD-ALM ».

Un slot peut être attaché à un frame de deux façons : comme un slot modèle (Template Slot) ou comme un slot propre (Own Slot). Un slot propre attaché à un frame décrit les propriétés d'un objet représenté par ce frame (un individu ou une classe). Les slots propres attachés à une classe ne sont pas hérités par ses sous-classes ou propagés à ses instances. Les slots modèles peuvent être attachés uniquement aux frames de classes, un slot modèle attaché à une classe est hérité par ses sous-classes. De plus, un slot modèle sur une classe devient un slot propre sur les instances de cette classe.

Par exemple, un slot qui contient la désignation d'une alarme spécifique –le slot désignation est attaché au frame qui représente cette instance individuelle de la classe alarme-- est un slot propre attaché à ce frame. toutes les autres slots sur le frame instance (ex : etat) sont aussi des slots propres attachés à cet instance.

les classes peuvent avoir aussi des slots propres. Par exemple, la documentation pour une classe est un slot propre attaché à cette classe depuis qu'il décrit la classe elle-même plutôt que ses instances. ce slot et ses valeurs ne sont pas héritées par les sous-classes. en effet la documentation (définition) assignée à la classe vanne n'est pas héritée par sa sous classe vanne de sûreté.

les slots modèles décrivent des propriétés qu'une instance d'une classe aura héritées comme slots modèles aux sous-classes, et ils deviennent des slots propres pour les instances.

pour résumer, les slots propres décrivent une propriété d'un frame (classe ou instance) lui-même plutôt que les propriétés d'instances de ce frame. les slots modèles décrivent des propriétés d'instances d'une classe.

### **2.1.5 Les facettes**

Une des manières pour spécifier les contraintes sur les valeurs de slot permises est à travers les facettes. Les contraintes spécifiées en utilisant les facettes incluent les cardinalités d'un slot (combien de valeurs le slot peut avoir), les restrictions sur le type de la valeur du slot (par exemple, nombre entier, chaîne de caractères, instance

d'une classe), valeur minimale et maximale pour un slot numérique, et ainsi de suite. Les facettes définissent aussi des restrictions sur un attachement du slot à un frame de classe.

### **2.1.6 Les formulaires d'acquisition de connaissances**

Protégé 2000 permet la saisie de données structurées en utilisant des formulaires d'acquisition de connaissances pour acquérir les informations des instances. Quand l'utilisateur définit une classe et lui attache des slots modèles protégé automatiquement génère un formulaire pour acquérir les instances de cette classe. Les slots de la classe, leur cardinalités et le type de valeur détermine la présentation par défaut et le contenu du formulaire qui peut être aussi personnalisé par l'utilisateur.

### **2.1.7 Interface**

les utilisateurs peuvent personnaliser la fenêtre standard que protégé 2000 génère automatiquement pour chaque classe afin de mieux convenir à ses exigences spécifiques .la personnalisation inclut le changement de la disposition des composants de la fenêtre en déplaçant les informations "importantes " au sommet de la fenêtre, en changeant des étiquettes pour les champs de la fenêtre et choisissant les différentes façons pour afficher et acquérir des valeurs du slot .par exemple nous pouvons utiliser un champ du texte (avec la validation appropriée ) .L'architecture à base de composants de protégé 2000 permet aux développeurs d'écrire acquérir et afficher des valeurs de slots à leurs propres composants de domaine spécifiques.

Protégé 2000 est un outil personnalisable , l'interface utilisateur consiste en plusieurs onglets permettant d'intégrer : la modélisation de l'ontologie ,La création des formulaires pour l'acquisition des connaissances ainsi que l'exécution de l'application. Au lancement de Protégé 2000, la fenêtre s'ouvre et les onglets standards deviennent visibles. Un nouveau projet s'ouvre toujours sur la vue des Classes. Les classes internes de Protégé THING et SYSTEM-CLASS sont tout ce qui est visible.



L'ontologie développée avec protégé 2000 peut également être éditée sous forme de pages HTML. Protégé est récemment adapté pour supporter la création et l'édition des ontologies du schéma RDF (Ressource Description Framework) et l'acquisition des instances RDF.

## 2.2. Jess

Jess (Java Expert system Shell) est entièrement écrit en langage Java de Sun. Jess supporte le développement des systèmes experts à base de règles qui peuvent être couplés au code écrit dans le langage puissant et portable Java. Ainsi Jess est une bibliothèque du programmeur. La bibliothèque est elle-même écrite en Java, elle sert comme un interpréteur d'un autre langage défini par « expert system shell » CLIPS, qui est à son tour spécialisé de LISP.

Jess peut être utilisé de deux manières. Premièrement, il peut être un moteur d'inférence ce dernier est un type spécial de programme qui applique très efficacement les règles aux données sous forme de base de connaissances. Les règles représentent des connaissances heuristiques des experts. Jess est également un langage de programmation général, et il peut directement accéder aux différentes classes de la librairie Java.

Jess dispose d'une ligne de commande interactive (`jess >`), une version graphique de cette même ligne de commande est également disponible, c'est la console jess.



Nous tapons la commande dans un champ texte en bas de la fenêtre et la sortie apparaît dans la fenêtre en haut .Jess s'exécute aussi comme applet java.

Sous Jess les identificateurs connus dans les autres langages sont appelés des atomes, jess utilise différent format de nombres, des chaînes et les listes qui sont des unités syntaxiques fondamentales de jess. Comme dans LISP, tout code sous jess (structure de contrôle, assignements, appels de procédure) est sous forme d'appel de fonction. Les variables sont des atomes commençant par (?). La variable avec le caractère (\$) au début réfère à une multivariable, spécifiant un type spécial de liste appelé « multifield ». Nous pouvons définir avec jess nos propre fonctions en utilisant le constructeur « deffunctions ».

La base de connaissance sous jess est défini par un ensemble de faits (facts), pour son développement il existe différentes commandes : deffacts, deftemplate, definstance, assert, modify,.. etc. L'exploitation de la base est assurée par des règles et des requêtes permettant la recherche de relations entre les faits de la base, à travers les commandes defrules et defqueries. Les règles sous jess sont traité par défaut comme l'instruction « if...then »c'est à dire en appliquant un raisonnement en chaînage avant. Pour utiliser un chaînage arrière il faut activer ce mode de raisonnement via la fonction « do-backward-chaining ». Jess est un langage extensible, il suffit d'ajouter de nouvelles fonctions au langage en écrivant une classe implémentant l'interface jess.userfonction.

### 2.3. Jess Tab

JessTab est un couplage entre protégé-2000 et Jess . Il fournit une fenêtre de Jess dans protégé. Jesstab et un ensemble d'extensions de Jess pour la transformation de la base de connaissance de protégé en des faits de jess ainsi que la manipulation de la base de connaissance de protégé . Ainsi JessTab peut être utilisé pour créer des programmes Jess qui prennent avantages de la base de connaissances protégé .Par exemple des règles sous Jess peuvent s'associer aux instances de protégé. L'intégration Protégé-Jess permet de développer la base de connaissances graphiquement dans protégé et exécuter la méthode de résolution du problème dans Jess. JessTab fournit des éditeurs pour les constructeurs Jess, comme les règles et les fonctions. Nous pouvons utiliser ces éditeurs pour afficher les définitions de Jess et

même les redéfinir. En plus, Jesstab peut stocker ces définitions avec la base de connaissances de protégé. Alternativement, nous pouvons stocker le programme Jess comme fichier séparé chargé au démarrage avec la base de connaissance.

Une utilisation typique de JessTab est de développer initialement une petite version de la base de connaissances dans protégé puis développer le code Jess qui opère sur cette base. Une fois le code Jess est performant nous pouvons étendre la base de connaissance.

### **3. Réalisation**

Etant donné l'importance du système, nous nous sommes focalisé sur l'implémentation de la base de connaissances avec le module métier qui l'exploite. Nous avons développé dans un premier temps un prototype du système portant sur un ensemble restreint de cas de pannes afin de nous permettre de valider les modèles développés, pour cela nous nous sommes focalisé sur l'implémentation de la BC (ontologie du domaine sous protégé 2000. Une fois faite on a utilisé Jess-Tab qui nous a permis d'utiliser le moteur d'inférence de Jess pour exploiter cette base, et appliquer ainsi la méthode de diagnostic appropriée. Le programme écrit sous Jess combine l'exploitation de règles et de requêtes pour répondre au problème de façon générale.

En réalité toute les manipulation de la base sont invisibles à l'utilisateur, les interfaces Java sont plus conviviales et permettent de manipuler les connaissances plus facilement par des clics souris, par des liens hypertextes, etc

#### **3.1 Présentation de l'application**

Dans le prototype développé on a représenté environ 450 concepts de différents types : des classes, des slots, et des instances. La base de connaissance est extensible du point de vue modélisation et instanciation. On note l'intégration d'un aspect sémantique au niveau de la base à travers le champs documentation où on peut spécifier une description textuelle des concepts. Ainsi, des annotations peuvent toujours être ajoutées aux concepts de la base et spécifiquement celles relatives aux

observations extraites des expériences de travail. On envisage l'exploitation ultérieure de cet aspect sémantique pour d'autres besoins .

Concernant la méthode de résolution de problème utilisée , on a développé sous Jess un programme exploitant la base de protégé et permettant de résoudre le problème de pannes qui survient suite à une très basse pression au niveau du système huile de commande du compresseur. Ce problème provoque le déclenchement de l'alarme L636HLD-ALM .En utilisant un ensemble de requêtes et de règles Jess et en inférant sur la base de connaissances , après avoir introduire quelques valeurs observables demandées (exemple : pression entrée et sortie filtre) le système essaye de localiser le composant responsable du dysfonctionnement et donne des consignes pour régler le problème .

L'évolution de ce système est possible par l'extension de la base de règles , permettant ainsi de traiter divers cas de pannes .

### **3.2 Description de l'ontologie**

Nous présentons une description textuelle de l'ontologie du domaine en annexe.

L'ontologie peut également être éditée sous format HTML .Nous donnons deux exemples de classes de l'ontologie .

**Project:** **compresseur**

**Class filtre**

Concrete Class Extends

[instrument](#)

**Direct Instances:**

None

**Direct Subclasses:**

[filtre d'aspiration pompe de lubrification](#), [filtre huile de lubrification double](#),  
[filtre d'aspiration huile de secours](#), [filtre huile de secours \(simple\)](#)

## Template Slots

Slot name	Documentation	Type	Allowed Values/Classes	Cardinality	Default
<i>constructeur</i>		String		0:1	
<i>quantité</i>		String		0:1	
<i>désignation</i>		String		0:1	
<i>pressostat HP</i>		Instance	<a href="#">pressostat HP différentielle</a>	0:1	
<i>Dint</i>		String		0:1	
<i>N°spécification</i>		String		0:1	
<i>Type</i>		String		0:1	
<i>indicateur press différent</i>		Instance	<a href="#">indicateur de pression différentielle gaz référence</a>	0:1	
<i>Plan installation</i>		String		0:1	
<i>Dext</i>		String		0:1	
<i>Hauteur</i>		String		0:1	
<i>code</i>		String		0:1	
<i>transmetteur filtre</i>		Instance	<a href="#">transmetteur de pression différentielle</a>	0:1	

**Project:** **compresseur****Class pompe de gavage****Concrete Class Extends**[pompe](#)**Direct Instances:**

None

**Direct Subclasses:**

None

**Template Slots**

Slot name	Documentation	Type	Allowed Values/Classes	Cardinality	Default
<i>debit</i>		String		0:1	
<i>pressostat hp</i>		Instance	<a href="#">pressostat HP pompe de gavage</a>	0:1	
<i>désignation</i>		String		0:1	
<i>pression aspiration</i>		String		0:1	
<i>puissance</i>		String		0:1	
<i>vanne PG</i>		Instance	<a href="#">vanne de sûreté sur pompe de gavage</a>	0:1	
<i>code</i>		String		0:1	
<i>Type</i>		String		0:1	
<i>N°spécification</i>		String		0:1	
<i>manometre</i>		Instance	<a href="#">manomètre de refoulement pompe de l'huile d'étanchéité BP</a>	0:1	
<i>pression refoulement</i>		String		0:1	
<i>vitesse</i>		String		0:1	
<i>constructeur</i>		String		0:1	
<i>Plan installation</i>		String		0:1	

### 3.3 Description des instances de la base de connaissances

Les instances représentent la base de connaissances effective, Nous avons extrait quelques exemples d'instances contenues dans la base développée, représenté en annexe.

### 3.4 Exemples Extraits de l'application

La fenêtre suivante représente une vue générale de l'ontologie développée, l'arbre de la partie gauche représente la hiérarchie de classes la fenêtre à droite montre un exemple de définition de la classe instrument.

Les onglets représentant différentes vues de la base de connaissance et des informations de configurations

La définition des slots propres

La définition des slots modèles

La hiérarchie de classes

Name	Type	Cardinality	Other Facets
S Type	String	single	
S code	String	single	
S constructeur	String	single	
S désignation	String	single	
S Plan installation	String	single	
S N*spécification	String	single	

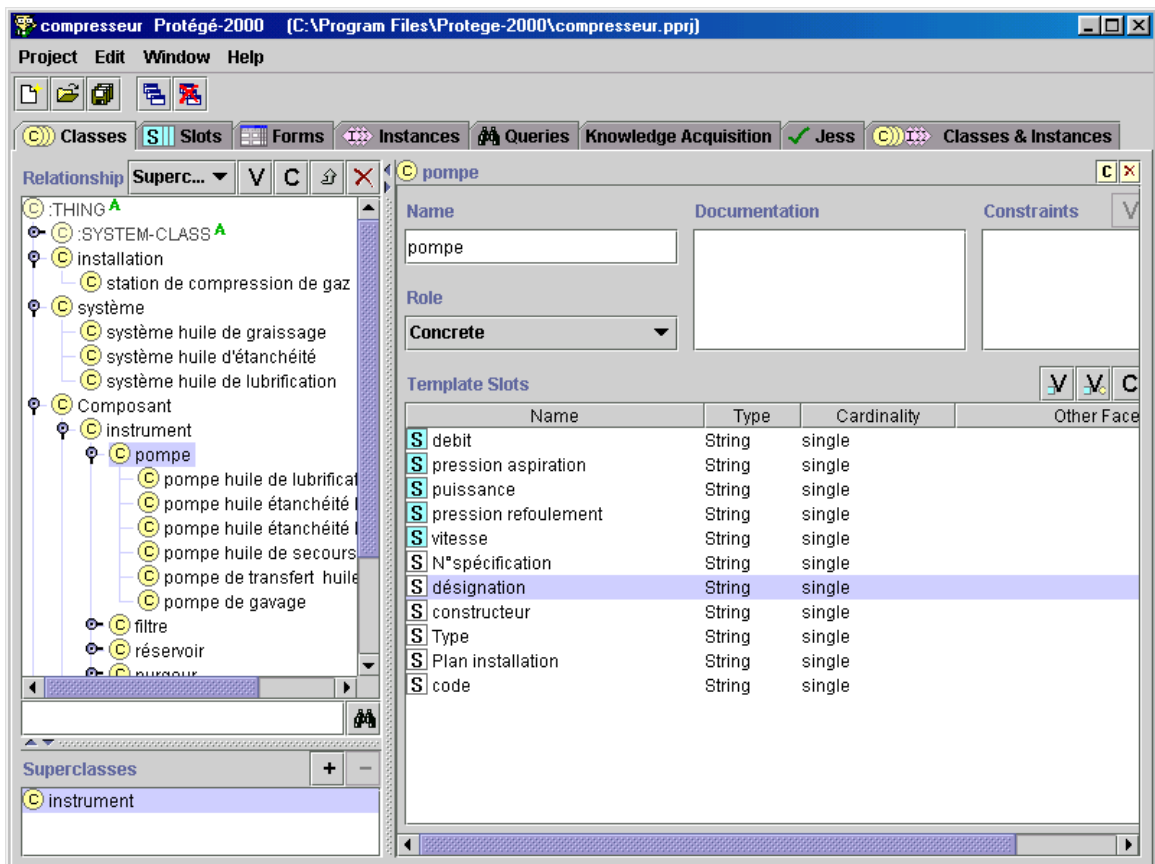
On note que l'héritage multiple est possible sous protégé 2000. C'est à dire une même classe peut avoir plusieurs super classes (ou classe parent).

Ainsi une classe est définie par ses slots propres tels que le nom de la classe, la documentation de la classe contenant une description de la classe, le rôle de la classe (une classe concrète ou abstraite) ainsi que les contraintes associées à la classe. La définition de la classe est complétée par ses slots modèles qui seront

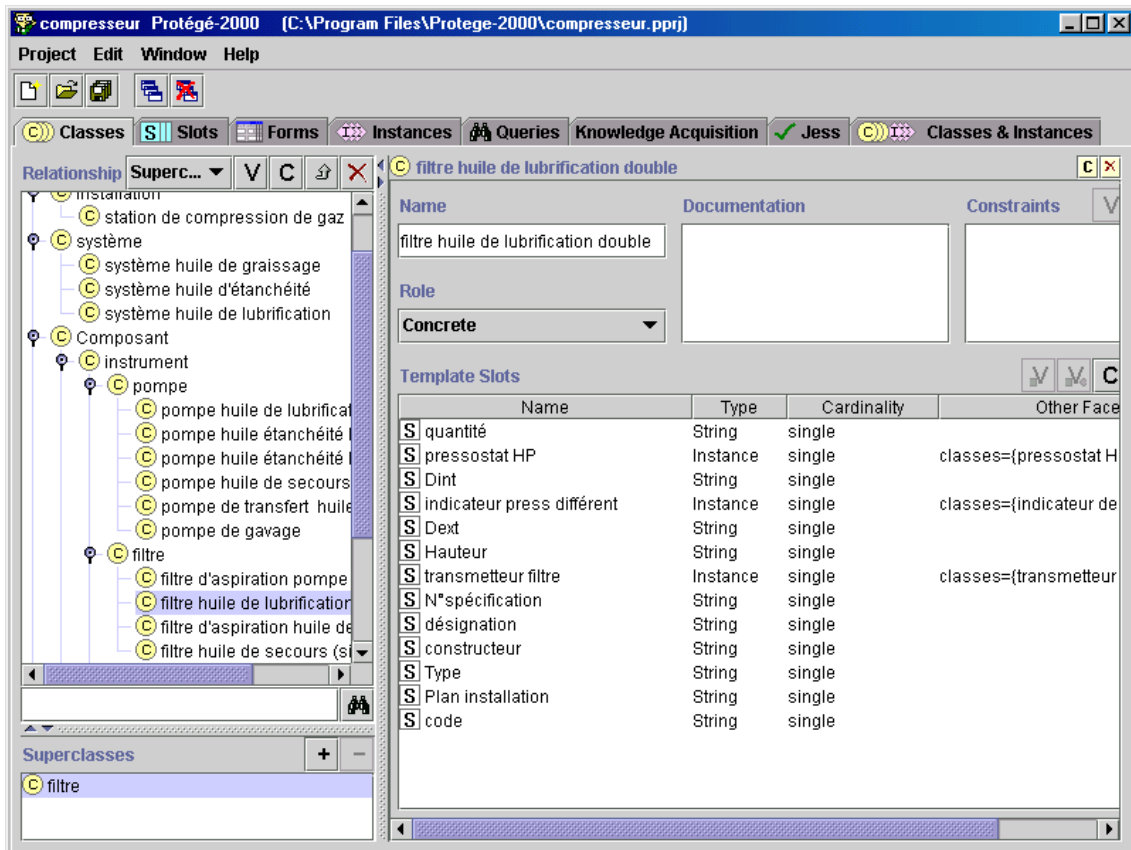
transférés à ses instances et hérités par ses sous classes permettant ainsi leur description.

Les fenêtres suivantes représentent des classes de la base de connaissances.

Description de la classe pompe avec ses sous classes ainsi que les propriétés (slots) associés à cette classe.



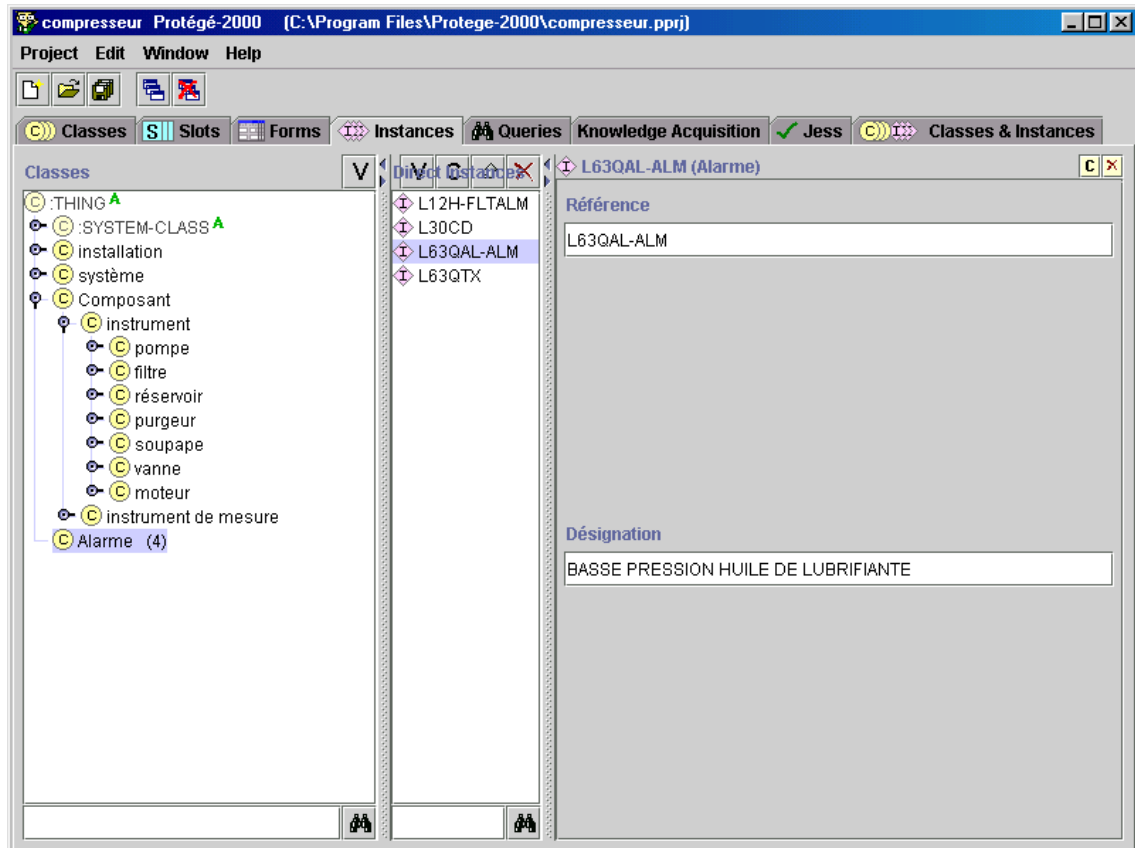
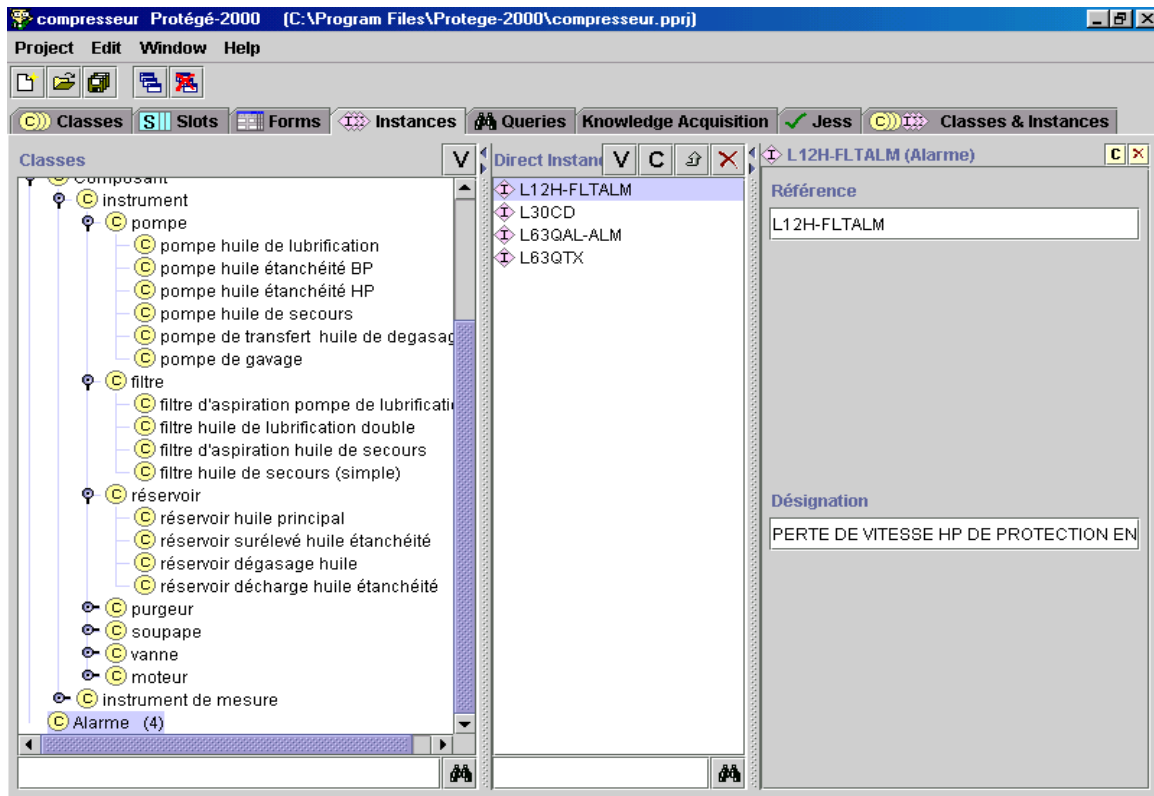
Description de la classe filtre huile de lubrification, une sous classe de la classe filtre.



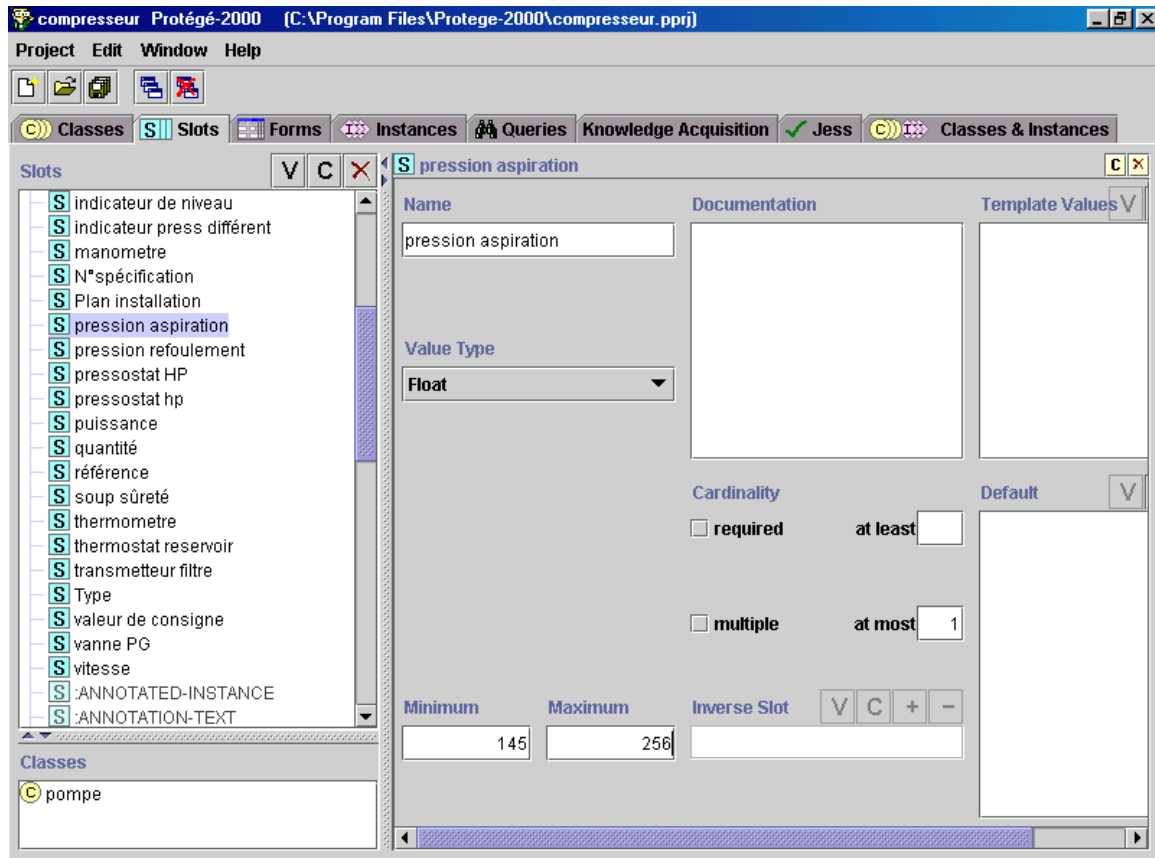
Les fenêtré suivante affiche des instances de la classe Alarme , représentant l'alarme L12H-FLTALM, et l'alarme L63QAL-ALM respectivement. Les champs affichent les valeurs des slots pour ces instances.

Sous l'onglet Instance trois parties peuvent être visualisées. La partie gauche représente la hiérarchie de classes , le nombre d'instances est affiché avec le nom de la classe( exemple Alarme (4)). La deuxième partie affiche la liste des instances de la classe sélectionnée .La troisième partie affiche les champs correspondants aux slots de la classe sélectionnée ces champs permettent l'édition et la saisie des valeurs de slots .



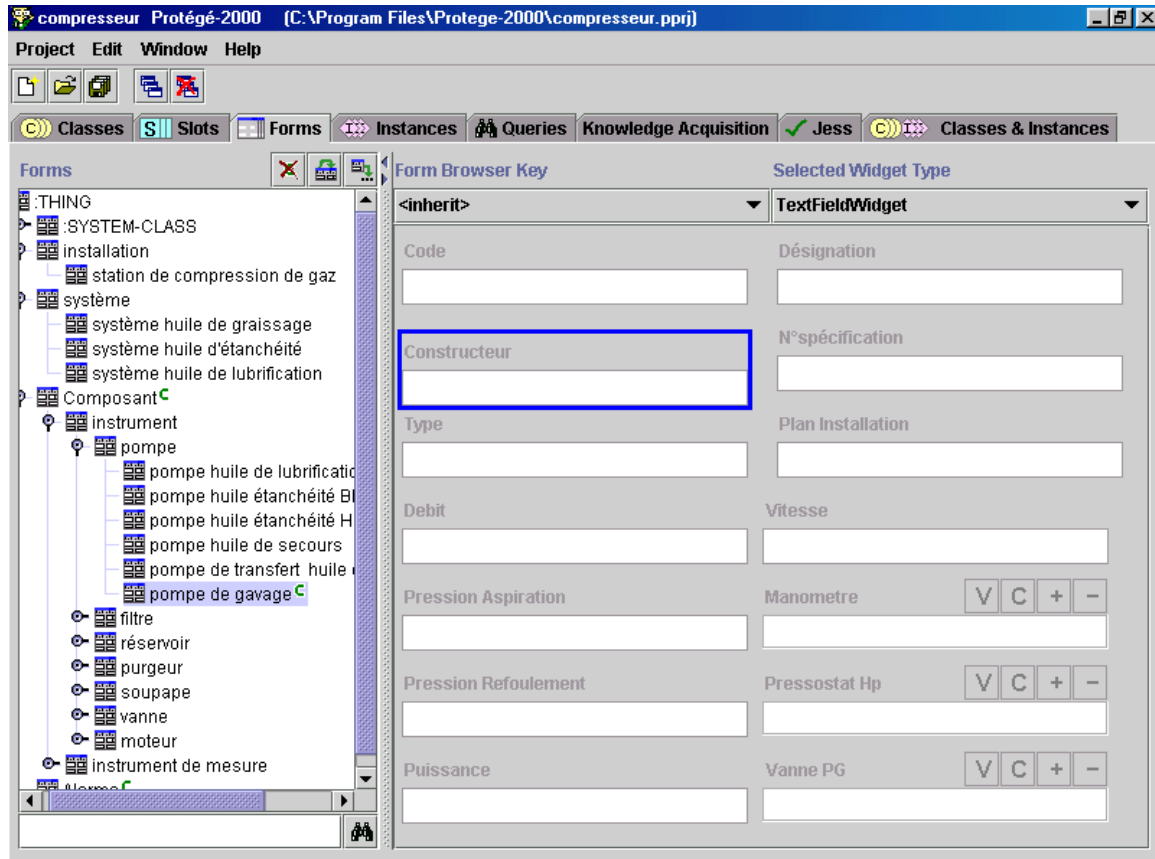


L'écran suivant affiche une fenêtre permettant la définition d'un slot avec les contraintes pouvant être associées à ce slot .

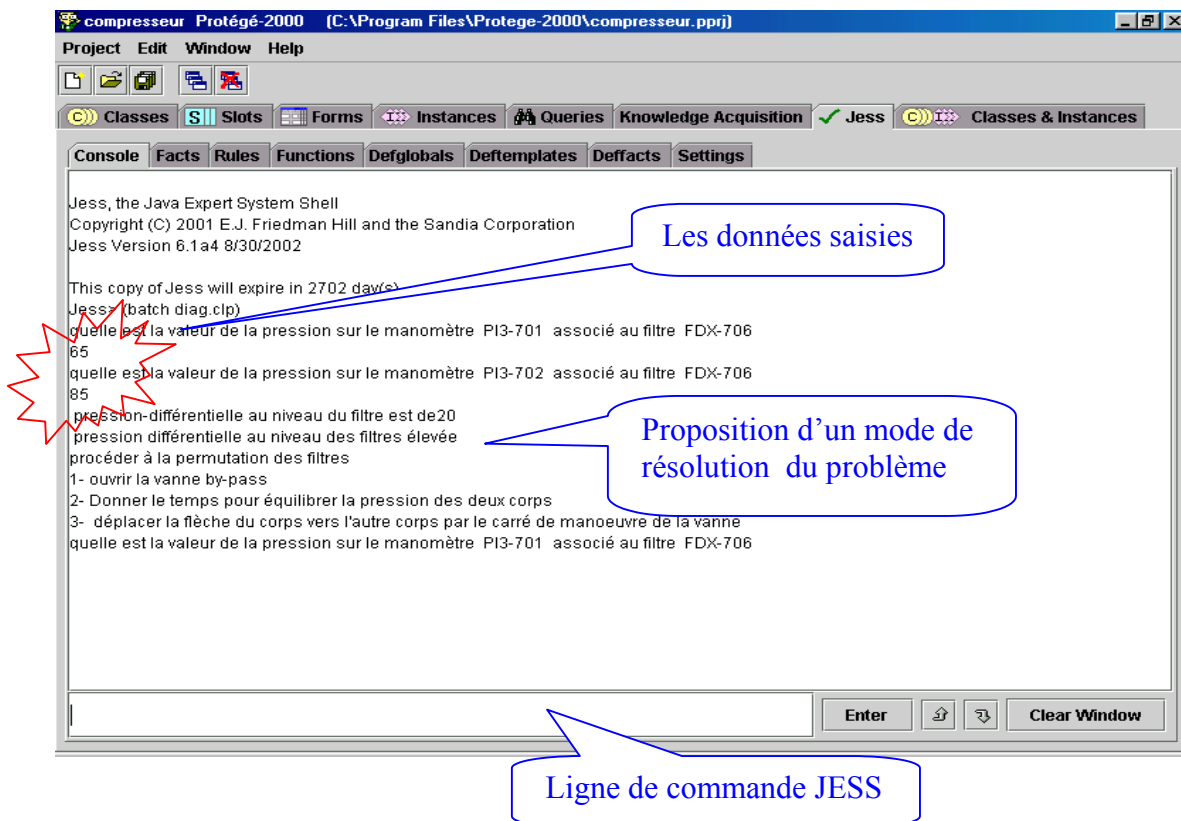


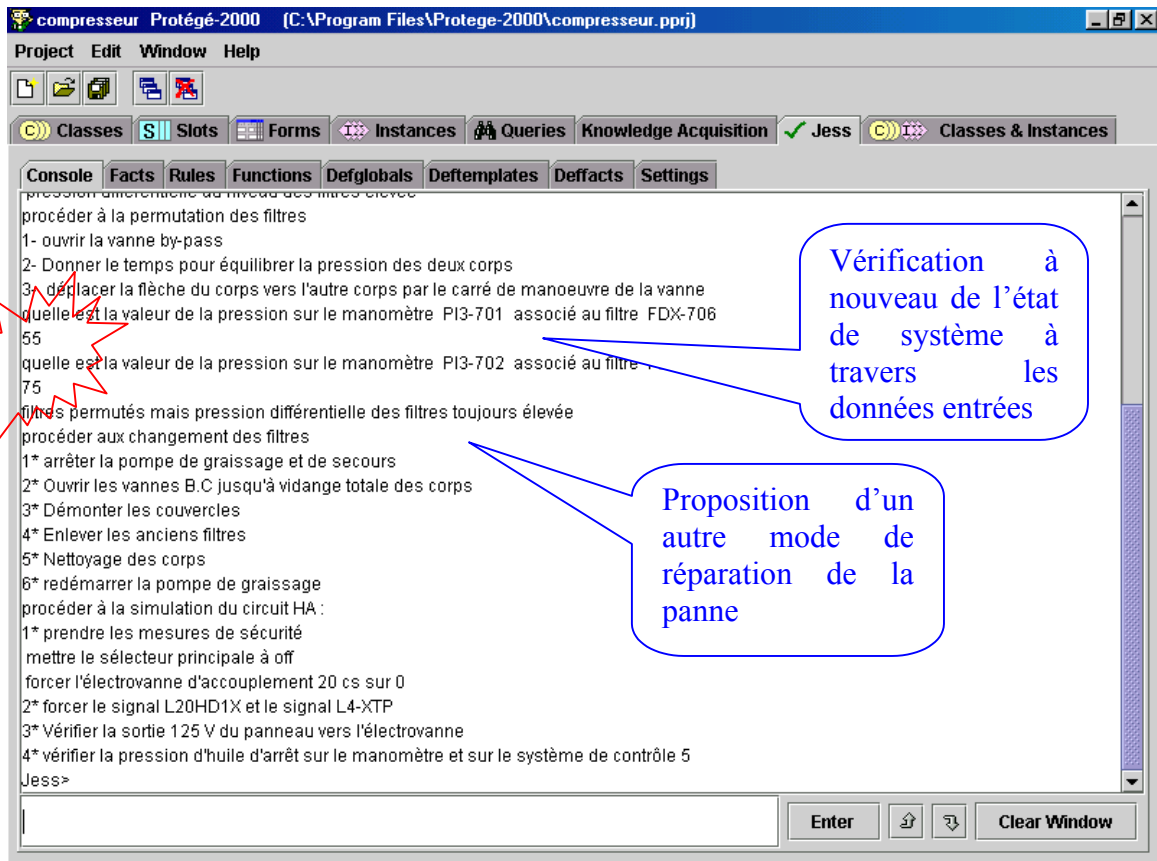
Le slot peut être à valeur unique ou multiple et il admet différents types : entier , réel, booléen , classe , instance ..etc . Une valeur par défaut peut être affectée au slot et une marge de valeur peut également être spécifiée. La documentation permet de décrire le slot et des notes peuvent être attribuées à chaque slot .

La fenêtre suivante affiche un formulaire pour l'acquisition des connaissances de la classe pompe de gavage ce formulaire est personnalisables et son format définitif sera utilisé pour l'édition et la création des instances de cette même classe .



Les écrans suivants représentent la fenêtre Jess accessible via Jess tab , permettant d'exécuter le programme exploitant la BC (ontologie) développée avec protégé 2000 ; une série de questions seront posées la réponse de l'utilisateur permet d'identifier la situation du système ce qui conduit à la détermination du cas de pannes suite aux inférences effectuées sur la base de règle formalisée en Jess . et donc de donner quelques consignes pour régler le problème.





# Conclusion & Perspectives

## **1. Conclusion**

Dans ce travail, nous avons étudié la problématique de gestion des connaissances, le domaine qui nous intéresse est celui des stations de compression de gaz en focalisant en premier lieu sur les turbines à gaz. Ces systèmes de gestion possèdent des spécificités intrinsèques liées au caractère potentiellement hétérogène des sources de connaissances. Ces spécificités nécessitent la mise en œuvre d'une démarche méthodologique de capitalisation des connaissances qui exploite des méthodes de l'ingénierie des connaissances basées sur des modèles conceptuels, c'est l'objectif que nous avons poursuivi pour la construction du système de gestion de connaissances pour la maintenance industrielle.

A partir de l'état de l'art du premier chapitre nous avons dégagé les points essentiels de la gestion des connaissances : Capitalisation des connaissances, mémoire d'entreprise et ontologie. par la suite et après un aperçu général sur les SBC nous avons abordé la phase de conception, nous avons défini les différents modèles qui permettent de spécifier le système de gestion de connaissances étudié : le modèle de décomposition structurelle et fonctionnelle du système, modèle de propagation de pannes, modèle de tâches heuristique et systématique. Les modèles de tâches s'appuient sur la méthode CommonKADS .

Enfin nous avons validé en partie les modèles par application d'une expérimentation sur le site et nous avons présenté un prototype dans lequel on s'est focalisé sur son noyau (le SBC : BC et méthode de résolution de problème ). L'utilisation de quelques exemples de cas de pannes a permis la réalisation des tests unitaires et globaux du prototype.

Le prototype établi ne répond pas à la totalité des attentes il reste à être amélioré notamment au niveau des interfaces.

Pour conclure, ce travail a permis d'ouvrir un sujet complexe consistant à l'utilisation d'une base de connaissances et d'une mémoire d'entreprise en milieu industriel, la problématique de maintenance axée sur la connaissance et la résolution de problème est fortement liée à l'IA d'où il faut exploiter les techniques et outils offerts par cette discipline pour servir au mieux le domaine de diagnostic.

## **2. Perspectives**

A terme, notre travail débouche sur les perspectives de recherches particulièrement intéressantes que nous proposons à priori à l'exploration :

- Nous envisageons d'améliorer le système de gestion documentaire en intégrant d'autres supports de connaissances (rapports de gestion, connaissances livresques, documents fournisseurs, guide, etc) en structurant ces documents sous une forme électronique (en XML par exemple) puis en mettant en place des outils techniques et interfaces permettant de gérer et interroger ces bases documentaire, Ce système sera considéré comme un complément du système de gestion de connaissances.
  
- Intégration d'un processus de retour d'expériences ce processus qui consiste à une analyse systématique de toute expérience issue de chaque activité de diagnostic. Ce processus est vue comme un mode de maintenance et d'enrichissement de la mémoire d'entreprise.



# Références

## RÉFÉRENCES

- [1] E.Aimeur (2000) , Evolution des systèmes à base de connaissances , 3<sup>ième</sup> séminaire national en informatique de Biskra ; 8,9&10 Mai 2000.
- [2] P. Allot et C.Coppens (2000), Utilisation de Connaissances Métiers dans les Outils d'Aide à la Conception ; Mise en oeuvre chez PSA , In Revue des arts et métiers 2000.
- [3] A. M. Alquier , M. H. Tignal (2000), Management des Connaissances et Management Par Projet , In J. Charlet , M. Zacklad , G. Kassel D. Bourigault , 2000  
Ingénierie des Connaissances , évolutions récentes et nouveaux défis , Edition Eyrolles.
- [4] F. Anguel (2003) , Contribution Pour l'Aide Au Diagnostic De Pannes Dans Un Système Industriel , In Proceeding ISPS'03 (Young Forum), 5,6&7Mai 2003.
- [5] F. Anguel (2003) , Une Approche de Construction d'un Système de Gestion de Connaissances Pour l'Aide au Diagnostic de Pannes Dans un Système Industriel  
In Proceeding JSTA'2003 , 24-25 Mai 2003.
- [6] S. Antipolis (2000), Acquisition des connaissances pour l'assistance à la conception entre agents Projet Acacia Rapport d'activité INRIA 2000.
- [7] S.Antipolis (2001) Action AID Gestion des connaissances pour l'aide à la conception coopérative .Rapport d'activité INRIA 2001.
- [8] N. Aussenac (1991) ,Dossier qui fait quoi ? acquisition des connaissances Bulletin de l'Affia N°5 mars 1991.
- [9] N. Aussenac J. Charlet (2000), Cours d'Acquisition de Connaissances
- [10] J. F Bally (1997), Capitalisation collective du savoir faire. L'expérience du projet DIADEME à EDF . In ALBI 97 , congrès Génie Industriel .
- [11] J. P. Barthès (1997), Capitalisation des connaissances et intelligence artificielle , in actes des journées franco finlandaise de Tampere 9-10 juin 1997 .
- [12] J.P. Barthès R. Dieng G. Kassel (1999), Dossier Mémoire d'entreprise Bulletin de l'Affia N°36 janvier 1999.
- [13] T. F. Barthelme V. Béatrice (2001),Analyse comparée de méthodes de gestion des connaissances pour une approche managériale .In Xième conférence de l'association internationale de management stratégique 13,14 &15Juin 2001 .
- [14] R.Benferhat (2002), Un modèle de projet et une mémoire de projet pour l'aide à l'élaboration de cahiers de charges au sein de Sonatrach Trc In actes de la conférence MCSEIA'02 6-8 Mai 2002 Annaba , Algérie pp 397-403
- [15] C.Brewster 2001 Knowledge acquisition for knowledge management : position paper :IJCAI 2001.
- [16] I.Boughzala (2001), Démarche méthodologique de conception de systèmes d'information coopératifs inter agents pour la gestion des connaissances , thèse de doctorat U. Paris VI 20 décembre 2001.
- [17] A Carvalho J Charlet (1999), Systèmes d'information et systèmes de gestion des connaissances premier pas vers une méthode d'architecture commune in actes de la conférence IC'99 juin 1999 pp 129-139.

- [18] J. Charlet , M. Zacklad , G. Kassel D. Bourigault (2000), Ingénierie des Connaissances : Recherches et Perspectives In J. Charlet , M. Zacklad , G. Kassel D. Bourigault , 2000 Ingénierie des Connaissances , évolutions récentes et nouveaux défis , Eyrolle
- [19] J Charlet (2002), Ontologies tutoriel RFIA 7 janvier 2002
- [20] E. Chouraqui , H. Farereny , D. Rayser , H. Prade (1985) Modélisation du Raisonnement et de la Connaissance , Technique et Sciences Informatiques vol.4 N°4/1985 p391-398 .
- [21] A.Claire B.Brisset (1999), Gestion des connaissances d'une mémoire corporative construite autour d'ontologie. In actes de la conférence IC'99 Juin 1999 , p181-188 .
- [22] C.Coppens (2002), Les outils de KBE : une approche chez PSA Peugeot Citroen In IDMME 2002 Clermont-Ferrand , France , May 14-16 2002 .
- [23] J. M David (1995), Les Systèmes Expert de Seconde Génération ou de l'Importance De la Modélisation dans la Construction de Systèmes à Base de Connaissances , Technique et Sciences Informatiques vol.14 N°4/1995 pp. 435-471.
- [24] S. Delaitre , S. Moisan et A. Mille (2001), Instrumentation d'un processus de retour d'expérience pour la gestion des risques .
- [25] R. Dieng O. Corby A. Giboin M. Ribière (1998), Methods and tools for corporate knowledge management Rapport de recherche INRIA N°3485 septembre 1998.
- [26] M. L. Doumbia , G. Roy V. Rajagopalan (1998), Méthodologie de développement d'un Système Expert D'Aide à la conception des Entraînements Electriques.
- [27] S.Duizabo N.Guillaume (1997), Les problématiques de gestion des connaissances dans les entreprises , DMSP cahier n°252 février 1997.
- [28] J. L Ermine , M. Chaillot , P. Bigeon , B. Charenton et D. Malacielles (1996), MKSM : Méthodologie Pour la Gestion des Connaissances Ingénierie des Systèmes d'Informations , APCET-Hermès ,1996 , Vol.4 , pp541-575 .
- [29] J. L. Ermine (2000), Enjeux et démarches de gestion des connaissances. , Net\_Actes 2000 : Textes des Communications .
- [30] J. L. Ermine M. Chaillot P. Bigeon B. Chaneton D. Malaveille (2000), MKSM Méthodes pour la gestion des connaissances
- [31] A. Faure G. Bisson (2000), Gérer les retours d'expérience pour maintenir une mémoire métier, Etude chez PSA Peugeot Citroen .
- [32] J. M.firestone M W.McElroy (2002), Generations of knowledge management juillet 2002.
- [33] C.Frydman , L.Torres, M, Le Goc (2000) Vérification et validation du modèle d'expertise du système SACHEM , USINOR .N° 1900759.
- [34] L. Giber (1999) ,D.Galereta Gestion des Connaissances dans le Contexte Spatial : Application Au Retour d'Expérience Sur la Sécurité des Serveurs In ESIA'99 .
- [35] A.J.Gadbout (1997), Que savons nous de la gestion des connaissances , in version préliminaire dans Kognos .the journal of knowledge Issues Summer 1997.
- [36] J. F. Gomes (2001), Knowledge infrastructure in new product development juin 2001.

- [37] W. E.Grosso H. Erikson M. .A.Musen (2000), Knowledge modelling at the millennium (the designed evolution of protégé 2000.
- [38] M. Grundstein , (1995) La Capitalisation des Connaissances de l'Entreprise , Système de production de Connaissances , In L'entreprise apprenante et les sciences de la complexité Aix en Provence .
- [39] M. Grundstein , (1998) Le Management des connaissances de L'Entreprise : Problématique , axe de progrès , approches opérationnelles , MG Conseil.
- [40] F. Gandon (2001), Experience in ontology engineering for a multi\_agents corporate memory system in actes de la conference IC'2001.
- [41] N. Guarino (1995), Formal ontology, Conceptual Analysis and knowledge representation in international journal of human and computer studies 43 (5/6) 1995 pp 625-646 .
- [42] N. Guarino (1997), Understanding, Building and using ontologies.
- [43] P.HaiK , S.Mahe and B.Ricard (2002) , Knowledge modelling as a support for knowledge acquisition ,sharing , operationalisation and maintenance EDF .
- [44] G. V. Heijst , R. V. D. Spek et E.Kruizinga , (1996) , Organizing Corporate Memories . In Knowledge Acquisition Workshop , Banff Canada .
- [45] F. R.Hickman (1989) Analysis for knowledge based systems a practical guide to the kads methodology .
- [46] G.Kassel (1999) , un nouveau cadre conceptuel pour les concepts de résolution de problèmes IC'99, Juin 1999
- [47] G.Kassel M.H.Adel ,C.Barry , P.Boulitreau C.Irastorza et S.Perpette (1999) Construction et exploitation d'une ontologie pour la gestion des connaissances d'une équipe de recherche , in actes de la conférence IC'99 Juin 1999 pp51-59 .
- [48] LAMIH (2000), Acquisition et gestion des connaissances groupe Prosper  
Projet Metacog version 22/03/2000
- [49] E. M. Laurent (1999), Quelques familles d'outils pour le knowledge management
- [50] M. Leclère B. Tixier F.Trichet (2000), Capitaliser et partager des connaissances rapport de recherche irin N° 00.11 Octobre 2000
- [51] M.Legoc (1999) Ontological models as shared model to validate a knowledge based Systems , USINOR
- [52] B. Leroux , P. et Laublet (1993) , An approach to knowledge acquisition combining alternate steps of constructive modelling and elicitation , In East West Artificial intelligence Conference EWAIC 93 , Moscou , Septembre 1993 .
- [53] P. Martin (1994), La méthodologie d'acquisition des connaissances KADS et les explications Rapport Inria N°2179 Janvier1994.
- [54] N. Matta , O. Corby , M. Ribière (1999), Méthodes de capitalisation de mémoire de projet rapport d'activité Inria 1999.
- [55] J.M.Mercatim, E.chouraqui et N.Bélanger (1999) , Etude d'un système d'aide au diagnostic des accidents de la circulation routière , in actes de la conference IC'99 , Juin 1999 pp 13-20

- [56] E.Mercier –Laurent (1999) , Quelques familles d’outils pour le knowledge management , EML Conseil knowledge Management 29 octobre 1999 [http:// pro.wanadoo .fr/eml-conseil](http://pro.wanadoo.fr/eml-conseil)
- [57] G.Morel et G.Kassel (2001) , Connaissances , Objet et Composants Complémentarité et intégration por le développement d’applications industrielles IC’2001
- [58] A.Musen , N. F. Noy , R. F. Mark (2000), The knowledge Model of protégé 2000 : combining interoperability and flexibility
- [59] S. Perpette (2000), Définition d’une méthode de construction d’ontologies : application à la gestion des connaissances d’une équipe de recherche.Rapport de recherche LRI d’Arniens univ de picardie Jules Verne
- [60] B.Ricard , P.Haik (1999), Distributed Knowledge based maintenance assistance for nuclear power plant components , EPRI International Maintenance Conference , Houston (USA) , Aout 2001.
- [61] B Ricard , P. Haik , j. lambert 2000 Un atelier pour la capitalisation et l’exploitation de connaissances de maintenance d’équipements industriels , EDF 2000
- [62] R. Sacile 1995 Utilisation de Commonkads pour la construction d’un mmodèle d’expertise pour le pronostic et la thérapie du cancer de sein Rapport de recherche Inria N°2737 1995
- [63] G.Schreiber , B.Wielinga , R.de Hoog (1994) CommaonKADS : A comprehensive Methodology for KBS development . IEEE EXPPERT Décembre 1994 pp 28-37
- [64] G. Schreiber , A. Anjewierden , M. De Alberdi , S. Tansley 1989 An Introduction To The KADS Methodology , A Report For Esprit Project .STC Technology Ltd
- [65] M. Sellami , A. Saadi , S. Bahaz ,F. Anguel M. Hadjeris (2003) , @MIND : Un Système Dynamique d’Intégration et de Consultation de Connaissances Accessibles sur le Web, 1<sup>ère</sup> conférence internationale SETIT, Sousse , Tunis 2003.
- [66] G. Simon (1997) Modèles et méthodes pour la conception des mémoires d’entreprises Le Système DOLMEN : une application en métallurgie –thèse de formation doctorale en informatique , Henri Poincaré Nancy 1 : Nancy 206p
- [67] A.C.Styliano , G.R Madey and R.D Smith (1992) , Selection criteria for expert system shells : a socio-technical framework , communications of the ACM octobre (1992) Vol35 n°10.
- [68] B. Tixier (2001) La problématique de la gestion des connaissances rapport de recherche sept 2001
- [69] Ressource Description Framework (RDF) shema specification 1.0 W3C 27 mars 2000
- [70] F. Anguel , M.Sellami (2003) , Un système de gestion de connaissances Pour l’aide au diagnostic de pannes dans un système industriel , In Proceeding of 1 st North African Region Seminar On Enginneering and Technology Education. Alger, 23-25 septembre 2003.

# Annexe

## Description de l'ontologie du domaine

Dans ce qui suit nous présentons une description textuelle de l'ontologie du domaine<sup>(1)</sup> :

```

;+ (version "1.7")

(defclass :CLIPS_TOP_LEVEL_SLOT_CLASS
  (is-a USER)
  (role concrete)
  (single-slot echelle
    (type STRING)
;+    (cardinality 0 1)
    (create-accessor read-write))
  (single-slot :STARTUP-FILE
    (type STRING)
;+    (cardinality 0 1)
    (create-accessor read-write))
  (single-slot quantité
    (type STRING)
;+    (cardinality 0 1)
    (create-accessor read-write))
  (single-slot :DEFINITION-NAME
    (type STRING)
;+    (cardinality 0 1)
    (create-accessor read-write))
  (single-slot thermostat^reservoir
    (type INSTANCE)
;+    (allowed-classes thermostat)
;+    (cardinality 0 1)
    (create-accessor read-write))
  (single-slot echelle^de^mesure
    (type STRING)
;+    (cardinality 0 1)
    (create-accessor read-write))
  (single-slot transmetteur^filtre
    (type INSTANCE)
;+    (allowed-classes transmetteur^de^pression^différentielle)
;+    (cardinality 0 1)
    (create-accessor read-write))
  (single-slot compresseur_00103
    (type STRING)
;+    (cardinality 0 1)
    (create-accessor read-write))
  (single-slot puissance
    (type STRING)

```

---

<sup>(1)</sup> Contenu dans le fichier (.ont) du projet protégé 2000.

```

;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot vitesse
  (type STRING)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot soup^sûreté
  (type INSTANCE)
;+      (allowed-classes soupape^de^sûreté)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot thermometre
  (type INSTANCE)
;+      (allowed-classes thermomètre^dans^la^casse^de^l'huile^de^dégasage)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot valeur^de^consigne
  (type STRING)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot constructeur
  (type STRING)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot Type
  (type STRING)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot :KB-SAVE
  (type SYMBOL)
  (allowed-values FALSE TRUE)
  (default FALSE)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot compresseur_00129
  (type STRING)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot indicateur^press^différent
  (type INSTANCE)
;+      (allowed-classes indicateur^de^pression^différentielle^gaz^référence)
;+      (cardinality 0 1)
        (create-accessor read-write))
(single-slot indicateur^de^niveau
  (type INSTANCE)

```



```

;+      (allowed-classes
indicateur^de^niveau^de^type^A^Rilexion^sur^purgeur^d'huile)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot Dext
        (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot pression^refoulement
        (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot :STARTUP-EXPRESSION
        (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot désignation
        (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot pressostat^hp
        (type INSTANCE)
;+      (allowed-classes pressostat^HP^pompe^de^gavage)
;+      (cardinality 0 1)
      (create-accessor read-write))

      (single-slot Dint
        (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot Hauteur
        (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot :DEFINITION
        (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot pressostat^HP
        (type INSTANCE)
;+      (allowed-classes pressostat^HP^différentielle)
;+      (cardinality 0 1)
      (create-accessor read-write))
      (single-slot code
        (type STRING)
;+      (cardinality 0 1)

```

```

        (create-accessor read-write))
(single-slot compresseur_00117
  (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
(single-slot pression^aspiration
  (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
(single-slot référence
  (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
(single-slot debit
  (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
(single-slot N°spécification
  (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
(single-slot vanne^PG
  (type INSTANCE)
;+ (allowed-classes vanne^de^sûreté^sur^pompe^de^gavage)
;+ (cardinality 0 1)
    (create-accessor read-write))
(single-slot manometre
  (type INSTANCE)
;+ (allowed-classes
manomètre^de^refoulement^pompe^de^l'huile^d'étanchéité^BP)
;+ (cardinality 0 1)
    (create-accessor read-write))
(single-slot Plan^installation
  (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass :JESS-ENGINE
  (is-a :SYSTEM-CLASS)
  (role concrete)
  (single-slot :STARTUP-FILE
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot :STARTUP-EXPRESSION
    (type STRING)

```

```
;+          (cardinality 0 1)
           (create-accessor read-write)))

(defclass :JESS-DEFINITION
  (is-a :SYSTEM-CLASS)
  (role abstract)
  (single-slot :DEFINITION
    (type STRING)
;+          (cardinality 0 1)
           (create-accessor read-write))
  (single-slot :KB-SAVE
    (type SYMBOL)
    (allowed-values FALSE TRUE)
    (default FALSE)
;+          (cardinality 0 1)
           (create-accessor read-write))
  (single-slot :DEFINITION-NAME
    (type STRING)
;+          (cardinality 0 1)
           (create-accessor read-write)))

(defclass :JESS-DEFGLOBAL
  (is-a :JESS-DEFINITION)
  (role concrete))

(defclass :JESS-DEFTEMPLATE
  (is-a :JESS-DEFINITION)
  (role concrete)
  (multislot :DOCUMENTATION
    (type STRING)
    (create-accessor read-write)))

(defclass :JESS-DEFFACTS
  (is-a :JESS-DEFINITION)
  (role concrete)
  (multislot :DOCUMENTATION
    (type STRING)
    (create-accessor read-write)))

(defclass :JESS-DEFMESSAGE-HANDLER
  (is-a :JESS-DEFINITION)
  (role concrete)
  (multislot :DOCUMENTATION
    (type STRING)
    (create-accessor read-write)))
```

```
(defclass :JESS-DEFMETHOD
  (is-a :JESS-DEFINITION)
  (role concrete)
  (multislot :DOCUMENTATION
    (type STRING)
    (create-accessor read-write)))

(defclass :RULE
  (is-a :SYSTEM-CLASS)
  (role abstract)
  (single-slot :DEFINITION
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot :DEFINITION-NAME
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass :JESS-RULE
  (is-a :JESS-DEFINITION :RULE)
  (role concrete)
  (multislot :DOCUMENTATION
    (type STRING)
    (create-accessor read-write)))

(defclass :FUNCTION
  (is-a :SYSTEM-CLASS)
  (role abstract)
  (single-slot :DEFINITION
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot :DEFINITION-NAME
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass :JESS-FUNCTION
  (is-a :FUNCTION :JESS-DEFINITION)
  (role concrete)
  (multislot :DOCUMENTATION
    (type STRING)
    (create-accessor read-write)))
```



```
                (create-accessor read-write))
(single-slot code
  (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass instrument
  (is-a Composant)
  (role concrete))

(defclass pompe
  (is-a instrument)
  (role concrete)
  (single-slot debit
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot pression^aspiration
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot puissance
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot pression^refoulement
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot vitesse
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass pompe^huile^de^lubrification
  (is-a pompe)
  (role concrete)
  (single-slot soup^sûreté
    (type INSTANCE)
;+ (allowed-classes soupape^de^sûreté)
;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass pompe^huile^étanchéité^BP
  (is-a pompe)
  (role concrete))
```

```

(defclass pompe^huile^étanchéité^HP
  (is-a pompe)
  (role concrete))

(defclass pompe^huile^de^secours
  (is-a pompe)
  (role concrete))

(defclass pompe^de^transfert^^huile^de^degasage
  (is-a pompe)
  (role concrete))

(defclass pompe^de^gavage
  (is-a pompe)
  (role concrete)
  (single-slot pressostat^hp
   (type INSTANCE)
   ;+ (allowed-classes pressostat^HP^pompe^de^gavage)
   ;+ (cardinality 0 1)
   (create-accessor read-write))
  (single-slot vanne^PG
   (type INSTANCE)
   ;+ (allowed-classes vanne^de^sûreté^sur^pompe^de^gavage)
   ;+ (cardinality 0 1)
   (create-accessor read-write))
  (single-slot manometre
   (type INSTANCE)
   ;+ (allowed-classes
manomètre^de^refoulement^pompe^de^l'huile^d'étanchéité^BP)
   ;+ (cardinality 0 1)
   (create-accessor read-write)))

(defclass filtre
  (is-a instrument)
  (role concrete)
  (single-slot quantité
   (type STRING)
   ;+ (cardinality 0 1)
   (create-accessor read-write))
  (single-slot pressostat^HP
   (type INSTANCE)
   ;+ (allowed-classes pressostat^HP^différentielle)
   ;+ (cardinality 0 1)
   (create-accessor read-write))
  (single-slot Dint

```

```

      (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
(single-slot indicateur^press^différent
      (type INSTANCE)
;+      (allowed-classes indicateur^de^pression^différentielle^gaz^référence)
;+      (cardinality 0 1)
      (create-accessor read-write))
(single-slot Dext
      (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
(single-slot Hauteur
      (type STRING)
;+      (cardinality 0 1)
      (create-accessor read-write))
(single-slot transmetteur^filtre
      (type INSTANCE)
;+      (allowed-classes transmetteur^de^pression^différentielle)
;+      (cardinality 0 1)
      (create-accessor read-write)))

(defclass filtre^d'aspiration^pompe^de^lubrification
  (is-a filtre)
  (role concrete))

(defclass filtre^huile^de^lubrification^double
  (is-a filtre)
  (role concrete))

(defclass filtre^d'aspiration^huile^de^secours
  (is-a filtre)
  (role concrete))

(defclass filtre^huile^de^secours^(simple)
  (is-a filtre)
  (role concrete))

(defclass réservoir
  (is-a instrument)
  (role concrete))

(defclass réservoir^huile^principal
  (is-a réservoir)
  (role concrete))

```



```

(defclass réservoir^surélevé^huile^étanchéité
  (is-a réservoir)
  (role concrete))

(defclass réservoir^dégasage^huile
  (is-a réservoir)
  (role concrete)
  (single-slot thermometre
    (type INSTANCE)
    ;+ (allowed-classes thermomètre^dans^la^casse^de^l'huile^de^dégasage)
    ;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot thermostat^reservoir
    (type INSTANCE)
    ;+ (allowed-classes thermostat)
    ;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot indicateur^de^niveau
    (type INSTANCE)
    ;+ (allowed-classes
    indicateur^de^niveau^de^type^A^Rilexion^sur^purgeur^d'huile)
    ;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass réservoir^décharge^huile^étanchéité
  (is-a réservoir)
  (role concrete))

(defclass purgeur
  (is-a instrument)
  (role concrete)
  (single-slot indicateur^de^niveau
    (type INSTANCE)
    ;+ (allowed-classes
    indicateur^de^niveau^de^type^A^Rilexion^sur^purgeur^d'huile)
    ;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass purgeur^automatique^huile^étanchéité
  (is-a purgeur)
  (role concrete))

(defclass purgeur^d'huile
  (is-a purgeur)
  (role concrete))

```

```

(defclass Purgeur^automatique^de^niveau
  (is-a purgeur)
  (role concrete))

(defclass soupape
  (is-a instrument)
  (role concrete))

(defclass soupape^de^sûreté
  (is-a soupape)
  (role concrete))

(defclass soupape^de^sûreté^pompe^de^gavage
  (is-a soupape^de^sûreté)
  (role concrete))

(defclass soupape^de^sûreté^ballon^huile^étanchéité
  (is-a soupape^de^sûreté)
  (role concrete))

(defclass soupape^à^trois^voies^de^temperature
  (is-a soupape)
  (role concrete))

(defclass vanne
  (is-a instrument)
  (role concrete))

(defclass vanne^de^réglage^pression^huile
  (is-a vanne)
  (role concrete)
  (single-slot valeur^de^consigne
   (type STRING)
   (cardinality 0 1)
   (create-accessor read-write))
  (single-slot echelle^de^mesure
   (type STRING)
   (cardinality 0 1)
   (create-accessor read-write)))

(defclass vanne^autorégulatrice^de^pression^différentielle^huile^gaz^BP
  (is-a vanne)
  (role concrete)
  (single-slot valeur^de^consigne
   (type STRING)
   (cardinality 0 1)

```

```
(create-accessor read-write))
(single-slot echelle^de^mesure
  (type STRING)
;+ (cardinality 0 1)
  (create-accessor read-write)))

(defclass vanne^autorégulatrice^de^pression^différentielle^HP
  (is-a vanne)
  (role concrete)
  (single-slot valeur^de^consigne
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot echelle^de^mesure
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass vanne^régulation^de^niveau^ballon^d'huile^étanchéité
  (is-a vanne)
  (role concrete)
  (single-slot echelle^de^mesure
    (type STRING)
;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass vanne^de^sûreté^sur^pompe^de^gavage
  (is-a vanne)
  (role concrete))

(defclass moteur
  (is-a instrument)
  (role concrete))

(defclass moteur^pour^pompes^huile^de^lubrification
  (is-a moteur)
  (role concrete))

(defclass moteur^pour^pompe^huile^de^secours
  (is-a moteur)
  (role concrete))

(defclass moteur^pour^pompe^huile^étanchéité^BP
  (is-a moteur)
  (role concrete))
```

```
(defclass moteur^pour^pompe^huile^étanchéité^HP
  (is-a moteur)
  (role concrete))

(defclass moteur^pour^pompe^de^transefert^huile^de^degasage
  (is-a moteur)
  (role concrete))

(defclass instrument^de^mesure
  (is-a Composant)
  (role concrete))

(defclass manostat
  (is-a instrument^de^mesure)
  (role concrete))

(defclass thermostat
  (is-a instrument^de^mesure)
  (role concrete))

(defclass thermomètre
  (is-a instrument^de^mesure)
  (role concrete)
  (single-slot echelle^de^mesure
    (type STRING)
    (cardinality 0 1)
    (create-accessor read-write)))
;+

(defclass thermomètre^dans^la^casse^de^l'huile^de^degasage
  (is-a thermomètre)
  (role concrete))

(defclass thermomètre^sur^le^huile^d'étanchéité^de^retour^de^bague^BP
  (is-a thermomètre)
  (role concrete))

(defclass thermomètre^sur^le^huile^de^retour^de^palier
  (is-a thermomètre)
  (role concrete))

(defclass manomètre
  (is-a instrument^de^mesure)
  (role concrete)
  (single-slot echelle^de^mesure
    (type STRING)
    (cardinality 0 1))
;+
```

```
(create-accessor read-write)))
```

```
(defclass manomètre^d'huile^de^lubrification^sur^collecteur
  (is-a manomètre)
  (role concrete))
```

```
(defclass manomètre^de^lubrification^sur^palier^de^comp
  (is-a manomètre)
  (role concrete))
```

```
(defclass manomètre^entrée^et^sortie^filtre
  (is-a manomètre)
  (role concrete))
```

```
(defclass manomètre^de^refoulement^pompe^de^l'huile^d'étanchéité^BP
  (is-a manomètre)
  (role concrete))
```

```
(defclass manomètre^refoulement^pompe^de^l'huile^d'étanchéité^HP
  (is-a manomètre)
  (role concrete))
```

```
(defclass pressostat
  (is-a instrument^de^mesure)
  (role concrete)
  (single-slot valeur^de^consigne
   (type STRING)
   ;+ (cardinality 0 1)
   (create-accessor read-write))
  (single-slot echelle^de^mesure
   (type STRING)
   ;+ (cardinality 0 1)
   (create-accessor read-write)))
```

```
(defclass pressostat^de^BP^d'huile^de^lubrification
  (is-a pressostat)
  (role concrete))
```

```
(defclass pressostat^de^minimum^pression^d'huile^de^lubrification^sur^collecteur
  (is-a pressostat)
  (role concrete))
```

```
(defclass pressostat^de^BP^d'huile^de^lubrification^sur^collecteur
  (is-a pressostat)
  (role concrete))
```

```

(defclass pressostat^HP^pompe^de^gavage
  (is-a pressostat)
  (role concrete))

(defclass pressostat^HP^différentielle
  (is-a pressostat)
  (role concrete))

(defclass indicateur
  (is-a instrument^de^mesure)
  (role concrete)
  (single-slot echelle^de^mesure
    (type STRING)
    (cardinality 0 1)
    (create-accessor read-write)))
;+

(defclass indicateur^de^pression^différentielle^gaz^référence
  (is-a indicateur)
  (role concrete))

(defclass indicateur^de^pression^en^"U"^pour^air^de^lavage
  (is-a indicateur)
  (role concrete))

(defclass indicateur^de^niveau^de^type^A^Rilexion^sur^purgeur^d'huile
  (is-a indicateur)
  (role concrete))

(defclass transmetteur
  (is-a instrument^de^mesure)
  (role concrete))

(defclass transmetteur^de^pression^d'huile^de^lubrification
  (is-a transmetteur)
  (role concrete)
  (single-slot echelle^de^mesure
    (type STRING)
    (cardinality 0 1)
    (create-accessor read-write)))
;+

(defclass transmetteur^de^vibration
  (is-a transmetteur)
  (role concrete)
  (single-slot valeur^de^consigne
    (type STRING)
    (cardinality 0 1)
    (create-accessor read-write)))
;+

```

```
(create-accessor read-write)))

(defclass transmetteur^niveau^ballon^d'huile^étanchéité
  (is-a transmetteur)
  (role concrete)
  (single-slot echelle^de^mesure
    (type STRING)
    (cardinality 0 1)
    (create-accessor read-write)))
;+

(defclass transmetteur^de^pression^différentielle
  (is-a transmetteur)
  (role concrete))

(defclass convertisseur
  (is-a instrument^de^mesure)
  (role concrete))

(defclass convertisseur^de^vibration
  (is-a convertisseur)
  (role concrete))

(defclass thermorésistance
  (is-a instrument^de^mesure)
  (role concrete)
  (single-slot echelle^de^mesure
    (type STRING)
    (cardinality 0 1)
    (create-accessor read-write)))
;+

(defclass thermoresistance^pour^palier^porteur
  (is-a thermorésistance)
  (role concrete))

(defclass thermoresistance^huile^graissage
  (is-a thermorésistance)
  (role concrete))

(defclass thermocouple
  (is-a instrument^de^mesure)
  (role concrete)
  (single-slot valeur^de^consigne
    (type STRING)
    (cardinality 0 1)
    (create-accessor read-write)))
;+
```

```
(defclass thermocouple^sur^palier^de^reducteur
  (is-a thermocouple)
  (role concrete))
```

```
(defclass thermocouple^sur^palier^de^compresseur
  (is-a thermocouple)
  (role concrete))
```

```
(defclass Alarme
  (is-a USER)
  (role concrete)
  (single-slot référence
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot désignation
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write)))
```



## Description des instances de la base de connaissances

Les instances représentent la base de connaissances effective , Nous avons extrait quelques exemples d'instances contenues dans la base développée<sup>(1)</sup> .

([compresseur\_00121] of  
 indicateur^de^niveau^de^type^A^Rilexion^sur^purgeur^d'huile  
 (constructeur "BONETTI")  
 (Type "-")  
 (code "LG 3.1X13")  
 (echelle^de^mesure "318 mm")  
 (Plan^installation "L")  
 (N°spécification "ITN 66020 rjo 039190156")  
 (désignation "Indicateur de niveau de type Ariflexion sur Lcv 3.1X15"))

([compresseur\_00122] of  
 indicateur^de^niveau^de^type^A^Rilexion^sur^purgeur^d'huile  
 (constructeur "BONETTI")  
 (Type "-")  
 (code "LG 3.1X14")  
 (echelle^de^mesure "318 mm")  
 (Plan^installation "L")  
 (N°spécification "ITN 66020 rjo 039190150")  
 (désignation "Indicateur de niveau de type Ariflexion sur LCV 3.1X16"))

([compresseur\_00123] of purgeur^d'huile  
 (constructeur "ARMSTRONG")  
 (Type "-")  
 (code "LCV 3.1X15")  
 (Plan^installation "L")  
 (indicateur^de^niveau [compresseur\_00121])  
 (N°spécification "ITN 64108 RSO 104700000")  
 (désignation "purgeur d'huile"))

([compresseur\_00124] of purgeur^d'huile  
 (constructeur "ARMSTRONG")  
 (Type "-")  
 (code "LCV 3.1X16")  
 (Plan^installation "L")  
 (indicateur^de^niveau [compresseur\_00122])  
 (N°spécification "ITN 64108 RSO 1047700020")  
 (désignation "purgeur d'huile"))

([compresseur\_00130] of Alarme  
 (référence "L63QTX")  
 (désignation "BASSE PRESSION HUILE SYNTHETIQUE  
 DECLENCHEMENT"))

---

<sup>(1)</sup> Contenu dans le fichier (.int) du projet protégé 2000.

([compresseur\_00131] of Alarme  
(référence "L63QAL-ALM")  
(désignation "BASSE PRESSION HUILE DE LUBRIFIANTE"))

([compresseur\_00132] of Alarme  
(référence "L30CD")  
(désignation "PERTURBATION REFROIDISSEMENT"))

([compresseur\_00133] of Alarme  
(référence "L12H-FLTALM")  
(désignation "PERTE DE VITESSE HP DE PROTECTION ENTREES"))